Universidad de Valladolid

Desarrollo y Evaluación de Riesgos de Privacidad en una App de Rastreo.

María Ruiz Molina

Tutores
Mercedes Martínez González
Amador Aparicio de la Fuente

Agradecimientos

En primer lugar, quiero agradecer a mis tutores Mercedes Martínez González y Amador Aparicio de la Fuente, el hecho de haberme elegido para realizar este proyecto, así como por su paciencia y tiempo dedicados a la corrección y orientación del mismo. También incluyo al profesor Julián Arroyo Álvarez por sus aportaciones a este trabajo.

En segundo lugar quiero dar las gracias a toda mi familia, en especial a mi madre, que ha estado siempre ahí, en momentos de risas y llantos; a mis tíos y a mis primos. Todos me han apoyado a lo largo de la carrera, y ayudado a tomar las mejores decisiones en los momentos más difíciles.

No me olvido de los que ya no están; mi padre, mi abuela, mi tía y mi tía abuela. Desde donde estén espero que se sientan orgullosos.

En tercer lugar quiero dar las gracias a la *Realeza Agastre*, mis amigos de la facultad y de fuera de ella, en especial a Dani, Willy, Christopher, Susana y Clara. Todos ellos han contribuido a que estos años hayan sido increíblemente buenos.

Y por supuesto y en especial, a Juan, que iniciamos este camino juntos aquel primer día de clase y lo hemos acabado a la vez, trabajando codo con codo, echando horas diarias, afrontando unidos los momentos complicados, y conociéndonos más y más. Gracias por haber hecho de este recorrido algo memorable.

Resumen

El trabajo aquí plasmado consiste en el diseño y desarrollo, junto a Juan Velázquez García, de una aplicación de rastreo de contactos, sobre la cual se elabora una evaluación de riesgos e impacto en la privacidad del usuario usando la herramienta PIA de la *Commission Nationale de l'Informatique et des Libertés* o CNIL (Francia), la cual implementa el Reglamento General de Protección de Datos.

Para ello, primeramente se realiza una profunda investigación sobre las aproximaciones para el desarrollo de aplicaciones de rastreo de contactos, la normativa entorno a ello y algunos casos existentes. También qué características poseen y los datos que obligatoriamente deben tratar.

Una vez estudiado el Estado del Arte, se elicitan aquellos requisitos necesarios para realizar una aplicación de rastreo de contactos.

Posteriormente, se realiza un estudio sobre el tipo de protocolos y paradigmas existentes en este área, se elige aquel que mejor se adapte al análisis de requisitos previamente realizado, y con ello se plantea y diseña la aplicación.

Durante todo este proceso se tiene en mente, siempre en primera instancia, la privacidad de los datos del usuario.

Tras ello, se ha desarrollado un prototipo de la aplicación empleando la técnica Pair Programming. Esto abarca tanto el desarrollo de la aplicación cliente como del servidor. También se logran los distintos tipos de comunicaciones existentes entre ellos, Bluetooth, TCP y UDP. Se comentan los distintos problemas que pudieran haber surgido durante el desarrollo de la aplicación y los cambios que se han tenido que realizar para poder ajustar el proyecto a la planificación sin grandes repercusiones.

Para demostrar el correcto funcionamiento de todo ello, se ha elaborado una lista de Casos de Prueba, clasificados en Caja Negra o Caja Blanca, Funcionales o No Funcionales, y aspecto concreto a tratar (seguridad, disponibilidad, interacción y comunicación). Se describe el procedimiento llevado a cabo para el éxito de cada uno de ellos, así como los distintos problemas que hayan podido surgir durante las pruebas y cómo se solventaron.

Una vez probado el correcto funcionamiento de la aplicación, se realiza una evaluación de riesgos e impacto en la privacidad. Para ello, se especifica primero el Estado del Arte, los distintos aspectos del Reglamento General de Protección de Datos a tratar en dicha evaluación, así como una presentación de la herramienta PIA, de la *Commission Nationale de l'Informatique et des Libertés* (CNIL), la cual será empleada para efectuar el análisis de riesgos y privacidad a la aplicación.

Detectadas las principales amenazas, se elabora una serie de salvaguardas con el fin de disminuir, para cada una, el riesgo de que esta se produzca.

Finalmente, se elabora una serie de conclusiones, así como una comparativa de este tipo de aplicaciones, sus vulnerabilidades y ventajas, frente a otras propuestas que han ido surgiendo a lo largo del año 2021 en la Unión Europea, territorio bajo la normativa del RGPD.

Abstract

This project aims to study some privacy standards, specifically, the proposals from the administrative authority, Commission Nationale de l'Informatique et des Libertés (CNIL).

Likewise, the current state of affairs regarding mobile contact-tracing apps' privacy risks will be reviewed. A risk evaluation will be proposed as a conclusion from the aforementioned standards.

Therefore, a prototyped app based on the current Covid contact-tracing applications will be developed, together with Juan Velázquez García, and the previously-established proposal will be applied to it.

Finally, a series of conclusions from this study will be presented in further detail.

Índice

1. Introducción **12**
1.1. Contexto . . . 12
1.2. Motivación . . . 12
1.3. Planteamiento del Problema . . . 13
1.3.1. Objetivos . . . 13

2. Planificación y Metodología **14**
2.1. Metodología . . . 14
2.2. Planificación . . . 15
2.3. Camino crítico . . . 16
2.4. Presupuesto . . . 18
2.5. Seguimiento . . . 19

3. Análisis de la aplicación **21**
3.1. **Requisitos no funcionales** . . . 21
3.1.1. **Requisitos de seguridad** . . . 21
3.1.2. **Requisitos de privacidad** . . . 23
3.1.3. Requisitos que se aplican únicamente cuando la aplicación envía al servidor una lista de contactos . . . 26
3.1.4. Requisitos que se aplican únicamente cuando la aplicación envía al servidor una lista de sus propios identificadores . . . 26
3.1.5. Requisitos de usabilidad . . . 27
3.2. **Requisitos Funcionales** . . . 27
3.2.1. **De cara al usuario** . . . 28
3.2.2. **De cara a otros dispositivos** . . . 28
3.2.3. De cara a la propia aplicación . . . 28
3.3. Requisitos de Información . . . 28
3.4. Base de Datos - Modelo conceptual . . . 29
3.5. Elección de paradigma de protocolos . . . 30
3.6. Base de Datos - Modelo lógico . . . 31
3.7. Modelo de intercambio de datos . . . 33
3.8. Desarrollo de los casos de uso . . . 33
3.8.1. UC-01 Enviar diagnóstico . . . 35
3.8.2. UC-02 Cambiar idioma . . . 35
3.8.3. UC-03 Comunicar semillas asociadas a IDs infectados . . . 36
3.8.4. UC-04 Comprobar riesgo de contagio . . . 36

3.8.5. UC-05 Enviar diagnóstico falso . . . 37
3.8.6. UC-06 Recibir ID . . . 37
3.8.7. UC-07 Enviar ID . . . 38

4. Diseño de la aplicación **39**
4.1. Estado del Arte: Protocolos . . . 39
4.1.1. Decentralized Privacy-Preserving Proximity Tracing (DP-3T) . . . 39
4.1.2. (Google/Apple) Exposure Notification (GAEN) system . . . 40
4.1.3. Pan-European Privacy-Preserving Proximity Tracing (PEPP-PT/PEPP) . . . 40
4.1.4. BlueTrace . . . 41
4.1.5. Otros . . . 42
4.2. Elección del protocolo . . . 43
4.3. Imprevistos respecto al protocolo elegido y consecuencias . . . 43
4.4. Implementación de los requisitos acorde a nuestra versión de DP-3T . . . 48
4.4.1. Tecnologías utilizadas . . . 54
4.5. Implementación de los Requisitos de Usabilidad . . . 54
4.6. Diseño de la Interfaz e Implementación de los Requisitos . . . 55
4.6.1. Pantalla Principal . . . 56
4.6.2. Pantalla de Información . . . 59
4.6.3. Pantalla de Ajustes de Idioma . . . 59
4.7. Diagrama de paquetes . . . 60
4.8. Implementación de la aplicación . . . 65
4.8.1. Implementación final de la interfaz . . . 65
4.8.2. Conexiones de red TCP . . . 67
4.8.3. Conexiones de red UDP . . . 67
4.8.4. Decisión sobre Bluetooth . . . 67
4.8.5. Cifrado de los datos . . . 68
4.8.6. Adaptaciones realizadas cara al prototipo . . . 69

5. Casos de prueba **71**
5.1. Pruebas de caja negra . . . 71
5.1.1. Intercambiar un ID entre dos dispositivos vía BT en rango . . . 71
5.1.2. Intercambiar un ID entre dos dispositivos vía BT en el límite del rango . . . 72
5.1.3. Intercambiar un ID entre dos dispositivos vía BT fuera de rango . . . 73
5.1.4. Intercambiar un ID entre dos dispositivos vía BT y desconectarse justo en el momento del envío . . . 73

5.1.5. Intercambiar un ID entre dos dispositivos vía BT y desconectarse justo 1 segundo antes del momento del envío . . . 73
5.1.6. Intercambiar un ID entre dos dispositivos vía BT y desconectarse justo 1 segundo después del momento del envío . . . 74
5.1.7. Uso de la aplicación sin conexión a BT . . . 74
5.1.8. Envío de un código correcto al servidor. (Código correcto: el pedido por la aplicación, otorgado por la autoridad sanitaria.) . . . 75
5.1.9. Envío de un código incorrecto al servidor . . . 76
5.1.10. Envío de diagnóstico sin fecha . . . 77
5.1.11. Envío de diagnóstico sin inserción de código . . . 77
5.1.12. Envío de código con menos de 12 cifras . . . 78
5.1.13. Envío de código con más de 12 cifras . . . 79
5.1.14. Envío de código con caracteres no numéricos . . . 79
5.1.15. Envío con fecha anterior a 14 días . . . 80
5.1.16. Envío con fecha posterior a 14 días . . . 81
5.1.17. Multicast de IDs infectados desde el servidor a los clientes . . . 81
5.1.18. Uso de la aplicación sin conexión a Internet (y sin recepción de multicast) . . . 82
5.1.19. Recepción por multicast de un ID infectado que se encuentra en la base de datos local 84
5.1.20. Recepción por multicast de IDs infectados y ninguno se encuentra en la base de datos local . . . 85
5.1.21. Recepción por multicast de IDs infectados y ninguno se encuentra en la base de datos local, pero el ID es un número por encima de uno almacenado . . . 86
5.1.22. Recepción por multicast de IDs infectados y ninguno se encuentra en la base de datos local, pero el ID es un número por debajo de uno almacenado . . . 86
5.1.23. Recepción por multicast de IDs infectados y no se posee ningún ID en la base de datos local con los que comparar . . . 87
5.1.24. Envío del multicast pero sin recepción (clientes inactivos) . . . 87
5.1.25. Escucha, por parte del cliente, de multicast pero sin envío (servidor inactivo) . . . 87
5.1.26. Cambio de idioma . . . 88
5.2. Pruebas de caja blanca . . . 89
5.2.1. Escucha del canal de conexión en el momento de envío de un código con los IDs al servidor . . . 89
5.2.2. Escucha del canal de conexión en el momento de envío de un código incorrecto al servidor 90
5.2.3. Escucha del canal de conexión en el momento del envío del multicast . . . 90
5.2.4. Barrido de puertos de la máquina cliente . . . 91
5.2.5. Barrido de puertos de la máquina servidor . . . 92
5.2.6. Ataque DDoS . . . 94
5.2.7. Inyección de código desde la aplicación cliente . . . 96

6. Análisis de riesgos de seguridad y privacidad **97**
6.1. Estado del arte: Seguridad y Privacidad . . . 97
6.1.1. Reglamento General de Protección de Datos . . . 97
6.1.2. Delegado de Protección de Datos . . . 100
6.1.3. Seguridad de los datos . . . 100
6.1.4. Evaluación de Impacto de Protección de Datos . . . 100
6.1.5. Unión Europea y aplicaciones de rastreo de contactos . . . 100
6.1.6. Unión Europea y estado COVID . . . 102
6.1.7. Commission Nationale de l'Informatique et des Libertés . . . 105
6.1.8. Herramienta PIA . . . 108
6.2. **PIA:** Evaluación de Impacto de la Privacidad . . . 110
6.2.1. Creación del proyecto . . . 110
6.2.2. Contexto . . . 114
6.2.3. Principios Fundamentales . . . 116
6.2.4. Riesgos . . . 119
6.2.5. Validación . . . 123
6.3. PIA: Proceso de la Evaluación . . . 126
6.3.1. Contexto . . . 126
6.3.2. Principios Fundamentales . . . 126
6.3.3. Riesgos . . . 126
6.3.4. Validación . . . 127
6.3.5. Conclusiones de la evaluación . . . 149
6.4. Comparativa de resultados con otras alternativas europeas . . . 150

7. Conclusiones y Trabajo Futuro **152**

Bibliografía **169**

Índice de figuras

1. Planificación inicial . . . 15
2. Planificación final . . . 19
3. Modelo conceptual . . . 30
4. Modelo lógico Cliente . . . 31
5. Modelo lógico Servidor . . . 32
6. Modelo de intercambio de datos . . . 33
7. Diagrama de casos de uso . . . 34
8. Esquema de generación de identificadores efímeros . . . 39
9. Solicitud a cumplimentar para tener acceso a la API . . . 43
10. Documentación requerida . . . 44
11. Modelo de datos de la base de datos de los clientes . . . 51
12. Modelo de datos de la base de datos del servidor . . . 53
13. Menú de navegación . . . 55
14. Pantalla Principal . . . 56
15. Botón Comunica tu contagio . . . 57
16. Confirmación del envío . . . 57
17. Simulación de los distintos tipos de daltonismo . . . 58
18. Pantalla de información . . . 59
19. Pantalla de idiomas . . . 59
20. Diagrama de Paquetes Aplicación Cliente JAVA . . . 61
21. Diagrama de Paquetes Aplicación Cliente RES . . . 62
22. Diagrama de Paquetes Aplicación Servidor . . . 63
23. Pantalla de inicio. Colores de la aplicación . . . 65
24. Pantalla de inicio. Protanopia . . . 66
25. Pantalla de inicio. Deuteranopia . . . 66
26. Pantalla de inicio. Tritanopia . . . 67
27. Resultado de entropías . . . 68
28. Intercambio de ID entre dispositivos . . . 72
29. Solicitud para activar Bluetooth . . . 74
30. Aplicación con contagio . . . 76
31. Aviso sobre código incompleto . . . 77
32. Aviso sobre código incompleto . . . 78
33. Límite anterior de aceptación de fechas . . . 80
34. Límite posterior de aceptación de fechas . . . 81

35. Aviso de desconexión . . . 82
36. Aplicación ejecutándose correctamente . . . 83
37. Aplicación ejecutándose correctamente . . . 85
38. Aplicación en inglés . . . 88
39. Resultado del programa de sniffing . . . 89
40. Resultado del sniffing . . . 90
41. Puerto 4446. Recepción de multicast. UDP . . . 91
42. Puerto 4445. Envío de multicast. UDP. . . . 92
43. Puerto 3327. Acceso a base de datos. TCP. . . . 93
44. Puerto 3384. Envío de códigos. TCP. . . . 93
45. Resultado de ataque de denegación de servicio. . . . 95
46. Diagrama del modelo de Self-Sovereign Identity. . . . 104
47. Esquema de la CNIL. . . . 106
48. Creación de nuevo proyecto PIA. . . . 110
49. Creación de nuevo proyecto PIA con datos. . . . 111
50. Interfaz PIA. . . . 111
51. Pasos para el análisis PIA. . . . 112
52. Recomendaciones para el análisis PIA. . . . 113
53. Pantalla Overview de Context en PIA. . . . 114
54. Pantalla Data, Processes and Supporting Assets de Context en PIA. . . . 115
55. Pantalla Proportionality and Necessity de Fundamental Principles en PIA. . . . 117
56. Pantalla Controls to Protect the Personal Rights of Data Subjects de Fundamental Principles en PIA. . . . 118
57. Pantalla Planned or Existing Measures de Risks en PIA. . . . 120
58. Tipos de riesgos en PIA. . . . 120
59. Pantalla ejemplo de Risks en PIA. . . . 121
60. Pantalla Risks Overview de Risks en PIA. . . . 122
61. Imagen visual generada en PIA. . . . 123
62. Plan de riesgos generado en PIA. . . . 124
63. Pantalla de comentario en Validación en PIA. . . . 125
64. Build APK . . . 158

Índice de tablas

1. Costes por personal del proyecto 18
2. Caso de uso: Enviar diagnóstico 35
3. Caso de uso: Cambiar idioma 35
4. Caso de uso: Comunicar semillas asociadas a IDs infectados 36
5. Caso de uso: Comprobar riesgo de contagio 36
6. Caso de uso: Enviar diagnóstico falso 37
7. Caso de uso: Recibir ID 37
8. Caso de uso: Enviar ID 38

1. Introducción

1.1. Contexto

Debido a la transformación tecnológica tan acelerada que estamos viviendo, el crecimiento de datos y la cantidad de información generada son abrumadores. En 2020 se generaban diariamente más de un billón de megabytes de datos, y la cantidad de datos almacenados se veía duplicada cada 10 meses. El flujo de todos ellos, debido al gran negocio existente en torno a la compra-venta de datos personales, hace que cada usuario aparezca en torno a entre 800 y 1000 bases de datos. Esto supone un importante riesgo para la privacidad de las personas de las que proceden estos datos, pues llega un momento en el que desconocen quiénes poseen su información. [35]

Muchos de estos datos se generan prácticamente sin darnos cuenta, como con el uso habitual de dispositivos móviles y personales para acceder a servicios.

Esta situación afecta a todas las áreas de la información y, recientemente debido a la situación de la pandemia, se están generando nuevas oleadas de datos.

En el año 2020 comenzó la pandemia de la COVID-19. Una de las primeras respuestas que hubo para tratar de frenar los contagios fue la creación de aplicaciones móviles de rastreo de contactos. Estas aplicaciones cumplen dos funciones. Almacenar el *estado COVID del usuario* y avisarle cuando haya estado en contacto con otro usuario contagiado. De esta forma se buscaba frenar la expansión del virus, pues si un usuario recibe la notificación de *Riesgo de contagio* puede hacer cuarentena a la espera de síntomas o un análisis PCR (Polymerase Chain Reaction), de este modo evitando propagar más el virus.

Todos estos datos generados son de carácter sensible, y por lo tanto deben cumplir un tratamiento donde la privacidad de los mismos sea central.

Es por ello que es obligatorio que las aplicaciones de rastreo de contactos, así como los protocolos por los que se rigen, cumplan con el RGPD y ante todo busquen y antepongan la privacidad de sus usuarios.

1.2. Motivación

Debido a que este proyecto trata un tema tan nuevo, han incluso surgido nuevas motivaciones a lo largo de su desarrollo.

Inicialmente, la motivación de este proyecto se debe a la escasa profundización y concienciación sobre este tema, que aún se encuentra en una fase incipiente, pero que afecta a una enorme cantidad de la población. La elaboración de un buen Estado del Arte sobre los distintos protocolos existentes, así como un análisis de riesgos y privacidad para estudiar cómo se tratan los datos de los usuarios en este tipo de aplicaciones, son inicialmente grandes razones de peso para llevar a cabo este proyecto. Además, el uso de aplicaciones de rastreo ha sido algo especialmente incentivado por las autoridades estatales y sanitarias.

No solo eso, debido a la rápida evolución que ha sufrido el ámbito aquí presentado a lo largo del año en el que se ha elaborado el proyecto, se genera una motivación más durante el desarrollo del mismo.

Esta segunda motivación viene propiciada por la necesidad de almacenamiento y control del estado COVID de los usuarios. Es por ello que la Unión Europea comienza a valorar distintas propuestas que cumplan este fin, comparando las distintas aportaciones de cada una. Dentro de ellas entran tanto las aplicaciones de rastreo de contactos, como nuevas vertientes que hacen uso de sistemas descentralizados, como por ejemplo la blockchain.

Como anunció la Unión Europea, se espera para 2024 que exista algún tipo de almacenamiento de la identidad digital de cada ciudadano miembro de la misma. Esta identidad digital iría asociada a diversa información del individuo, y entre ella se encontraría el estado COVID.[60] [57]

1.3. Planteamiento del Problema

El supuesto presentado es una aplicación que trata con datos de carácter sensible, tales que los relacionados con la salud de las personas, y por ello merecen de un especial cuidado. Es por ello que este tipo de servicios deben respetar y clarificar a sus usuarios qué datos son recogidos y con qué fines van a ser tratados, prestando gran atención a su privacidad tal y como especifican las autoridades reguladoras de la protección de datos.

La aplicación que se ha desarrollado pues, se plantea desde dicho punto de vista, cumpliendo con una serie de estrictos requisitos de privacidad impuestos por el European Data Protection Board, así como otras entidades y aquellos que se han considerado oportunos para el correcto funcionamiento. Posteriormente para verificar el correcto cumplimiento con respecto al RGPD 7, se realizará un análisis de privacidad de la misma empleando la herramienta PIA, la cual implementa tales normativas.

El RGPD es el reglamento europeo que rige el tratamiento de datos personales, así como de su libre circulación. Es una normativa a nivel de la Unión Europea, por lo que toda entidad que establezca negocios con sus territorios y que trate datos personales, ha de cumplirla.

De la mano de esto se habrán realizado previamente los correspondientes casos de prueba para tanto comprobar el correcto funcionamiento de la aplicación, como detectar posibles brechas de datos que pudieran afectar a la privacidad de los usuarios.

1.3.1. Objetivos

El objetivo final de este proyecto es realizar una propuesta de evaluación de privacidad en aplicaciones de rastreo de contactos y aplicarla a un caso real.

Para ello, es necesario ir cumplimentando una serie de subobjetivos, manteniendo siempre como punto principal la privacidad. Estos son los siguientes:

- **Objetivo 1.** Conocer el Estado del Arte de los protocolos de rastreo de contactos, de la evaluación de riesgos de privacidad y determinar qué protocolos y directrices se deben tener en cuenta durante el diseño y desarrollo.
- **Objetivo 2.** Realizar el análisis, diseño y desarrollo de una aplicación prototipo de rastreo de contactos, y sus respectivos casos de prueba. Se empleará la técnica de la programación en pareja alternando roles.
- **Objetivo 3.** Elaborar y aplicar una propuesta de análisis de riesgos de seguridad y privacidad a partir de lo presentado por las soluciones de la CNIL y de lo visto en el estudio del Estado del Arte.
- **Objetivo 4.** Obtener una serie de conclusiones a partir del estudio realizado para determinar los puntos fuertes y débiles de la aplicación en cuestiones de seguridad y privacidad.

Finalmente, de las conclusiones obtenidas del estudio, se espera que las brechas detectadas en el tratamiento de los datos se tengan en cuenta en futuros proyectos de índole similar.

2. Planificación y Metodología

En este apartado se procede a explicar la planificación llevada a cabo para la elaboración del proyecto, así como la metodología de trabajo usada.

2.1. Metodología

Debido a que el desarrollo de la aplicación es llevado a cabo por dos alumnos, Juan Velázquez García y María Ruiz Molina, se ha optado por una metodología de desarrollo Extreme Programming [40], en concreto a la hora de la implementación de la aplicación, Pair Programming o programación en pareja.

Dicha metodología se basa en los siguientes puntos:

- **Constante comunicación cara a cara y feedback.** Si bien debido a que el desarrollo de la aplicación se ha realizado durante la pandemia del COVID-19, se ha hecho uso de aplicaciones de videoconferencias con el fin de realizar una comunicación lo más cercana posible. A los tutores se les ha ido informando de manera incremental de los cambios y de las nuevas implementaciones según se iban realizando.
- **Simplicidad.** A la hora de tomar la decisión que concierne al diseño de la aplicación, se optará por aquella más sencilla y que implemente los requisitos previamente elicitados.
- **Responsabilidad.** La calidad del software recae en los propios desarrolladores. En este marco se incluye el desarrollo de casos de prueba así como del posterior análisis de privacidad.
- **Coraje.** Capacidad para desechar trabajo ya realizado, si así se requiere, con el fin de investigar otras ideas que puedan ser más apropiadas, se adapten mejor a lo solicitado o solucionen algún posible imprevisto.

En concreto, la programación en pareja se basa en que el código sea escrito por parejas (en este caso una única) de desarrolladores.
Mientras uno escribe el código como tal, el otro observa, discute las ideas y hace comentarios o sugerencias mientras investiga sobre ello. Dichos roles se han de ir intercambiando con el fin de que el trabajo dedicado a cada papel sea igual para ambos miembros. [40]

2.2. Planificación

Con la metodología previamente explicada en mente, se lleva a cabo la siguiente planificación:

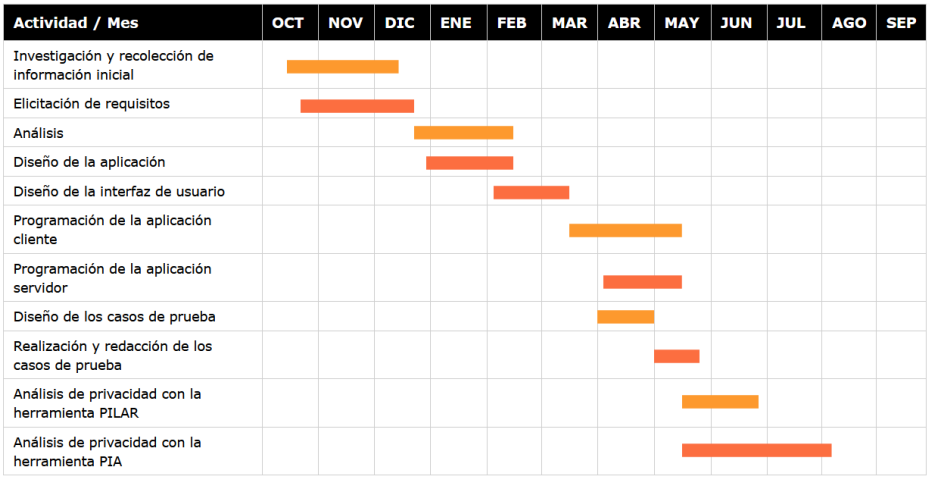


Figura 1: Planificación inicial

Como puede verse, el proyecto se organizó y distribuyó en distintas fases a lo largo de un total de 42 semanas, entre mediados de octubre, momento de inicio, hasta mediados de agosto. Al ser un trabajo con una parte en común y otra individual, los análisis de privacidad, se reflejan ambos en el diagrama de Gantt. Juan Velázquez García termina el proyecto y su parte el mismo día que la actividad de *Análisis con la herramienta PILAR* acaba.
María Ruiz Molina termina el proyecto y su parte el mismo día que la actividad de *Análisis con la herramienta PIA* acaba.

- **Investigación y recolección de Información Inicial.** Desde la semana 1 a la semana 8. Se subdivide en:
  - **Investigación de protocolos de rastreo de contactos.**
  - **Investigación de la aplicación Radar Covid.**
  - **Investigación sobre estándares de privacidad.**
  - **Investigación sobre el protocolo DP-3T.**
- **Elicitación de requisitos.** Desde la semana 2 a la semana 10.
- **Análisis.** Desde la semana 10 a la semana 15. Se subdivide en:
  - **Investigación sobre los tipos de paradigmas en aplicaciones de rastreo de contactos.**
  - **Casos de uso.**
  - **Actores.**
  - **Diagrama de casos de uso.**
  - **Modelo de intercambio datos.**
- **Diseño de la aplicación.** Desde la semana 10 a la semana 16. Se subdivide en:

  - **Investigación sobre la generación de identificadores efímeros**
  - **Diseño del protocolo.**
  - **Diseño de la base de datos del cliente.**
  - **Diseño de la base de datos del servidor.**
- **Diseño de la interfaz de usuario.** También se considera su programación. Desde la semana 15 a la semana 20.
- **Programación de la aplicación cliente.** Desde la semana 20 a la semana 28.
- **Programación de la aplicación servidor.** Desde la semana 24 a la semana 28.
- **Diseño de los casos de prueba.** Desde la semana 24 a la semana 28.
- **Realización y redacción de los casos de prueba.** Desde la semana 28 a la semana 31.
- **Análisis con la herramienta PILAR.** Desde la semana 30 a la semana 35, teniendo un margen de 3 semanas más hasta el momento de cierre de actas.
- **Análisis con la herramienta PIA.** Desde la semana 30 a la semana 42, teniendo un margen de 3 semanas más hasta el momento de cierre de actas.

2.3. Camino crítico

Dentro del proyecto, hay etapas que pueden provocar un desplazamiento temporal importante en las subsiguientes. Esto puede deberse a la necesidad que tienen unas de que otras hayan sido acabadas del todo.

En concreto nos encontramos con las siguientes etapas:

- **Elicitación de requisitos.** Esta primera fase es necesaria para poder empezar a elaborar las necesidades de la aplicación central del proyecto, la cual es necesaria para poder llevar a cabo el análisis de riesgos y privacidad. Debe ser previa al análisis, donde se tendrán en cuenta las necesidades aquí encontradas.
- **Análisis.** Al igual que la anterior, esta etapa es necesaria para poder comenzar a dar forma a lo que será la aplicación de rastreo de contactos.
- **Diseño de la aplicación.** Del mismo modo que las dos fases anteriores, es necesario que la tarea de diseño se realice previamente al desarrollo. De este modo, un retraso en esta parte afectaría a la programación de la aplicación y con ello a las tareas que dependen de un prototipo funcional.
- **Programación de la aplicación cliente.** Debido a que las etapas que la siguen necesitan probarla y analizarla, es necesario que la aplicación cliente funcione correctamente en todos los aspectos.
- **Programación de la aplicación servidor.** Del mismo modo, debido a que las tareas que la siguen necesitan probarla y analizarla, es necesario que el servidor esté terminado y funcionando para poder continuar.

La tarea de **Investigación y recolección de Información Inicial**, si bien una parte ha de llevarse a cabo al principio del proyecto para poder tener un mejor enfoque, parte de la investigación se puede llevar a cabo de manera paralela a la **Elicitación de Requisitos**, e incluso en momentos más avanzados del proyecto, pues al tratarse de un campo tan nuevo, es frecuente que surjan nuevas fuentes de información que ayuden a redondear aspectos concretos.

Por otro lado, el **Diseño de la interfaz de usuario** es un apartado que puede posponerse, pues no hace falta un diseño atractivo para poder comprobar el correcto funcionamiento de la aplicación, así como analizar cómo trata los datos que recolecta.

El **Diseño de los casos de prueba** puede elaborarse en parte de manera paralela al **Análisis con la herramienta PIA**. Si bien hay ciertas funcionalidades que deberán probarse antes de realizar el análisis para corroborar que se va a evaluar la aplicación tal y como se ha diseñado, la evaluación de otros casos de prueba no necesita ser previa al análisis de riesgos y privacidad. Estos pueden ser aquellos relacionados con la interacción o disponibilidad.

2.4. Presupuesto

Se procede a presentar un análisis del coste monetario del proyecto.

Para realizar el cálculo del tiempo empleado por ambos alumnos en el desarrollo del proyecto, se parte del total de créditos de un Trabajo de Fin de Grado de Ingeniería Informática. Esto equivale a **12 ECTS**, y dado que 1 ECT equivale a 25 horas de trabajo, se obtiene un total de 300 horas por proyecto. Dado que se engloban dos TFG, el total sería de 600 horas, muchas de ellas elaboradas realmente en paralelo en el momento inicial de creación de la aplicación (la parte común). Una vez esta serie de tareas comunes es terminada, las horas restantes se desarrollarían de manera individual por cada alumno.

Así, se va a calcular en concreto para este proyecto, teniendo en cuenta las horas llevadas a cabo en la parte común junto a las del compañero Juan Velázquez García, y las de la parte individual únicamente las respectivas a las de mi parte correspondiente.

Considerando en la planificación que la parte común llevará de la semana 1 a la 30 y la parte individual de la 30 a la 42, equivale a una proporción de 30 semanas sobre 42 para la parte común, esto es aproximadamente un 70 % del proyecto y un 30 % para la parte individual.

De este modo, de un total de 600 horas suma de ambos esfuerzos, 600 * 0.7 = 420 horas. Sin embargo esta primera parte, dado que fue iniciada apenas comenzó el curso escolar, el total de horas es superior, 450 para cada alumno y haciendo un total de 900 * 0.7 = 630 horas (315 para cada alumno).

Por otro lado, de 300 horas que lleva a un único alumno un TFG, 300 * 0.3 = 90 horas.

En total, el esfuerzo entre ambas partes será de 720 horas, teniendo en cuenta que las correspondientes a la primera parte se dividen entre dos alumnos.

Se ha utilizado la página de la Sección Sindical CGT Sopra Steria, https://cgtsoprasteria.org/blog/rangos-salariales-anuales-tic/, para obtener los salarios.

Por ello, se considera un salario promedio para programador junior (grupo E-I) de 14800 euros al año por 7.5 horas diarias, una para cada alumno, se obtiene, a 6.51 euros la hora[20], para la primera alumna 6.51 * 405 = 2636.55 euros y para el segundo alumno, 6.51 * 315 = 2050.65 euros.

Por otro lado, el trabajo de guía, corrección y resolución de problemas llevado a cabo por los tutores del proyecto, se calculará empleando un salario medio de programador senior (grupo D-I). Esta cifra equivale a 16531 euros al año, que se traduce a 7.27 euros la hora.[20] Entre el tiempo empleado en reuniones, lecturas y correcciones de la memoria, así como resolución de dudas, se calculan unas 60 horas en total de dedicación.

Esto hace un total de 60 * 7.27= 436.2 euros * 3 tutores en total = 1308.6 euros

Para una mejor visualización de los costes, estos pueden consultarse en la siguiente tabla:

Personal	Coste €/Hora	Horas	Total €
Alumna María Ruiz Molina	6.51	405	2636.55
Alumno Juan Velázquez García	6.51	315 *para este proyecto	2050.65
Tutora Mercedes Martínez González	7.27	60	436.20
Tutor Amador Aparicio de la Fuente	7.27	60	436.20
Tutor Julián Arroyo Álvarez	7.27	60	436.20

Tabla 1: Costes por personal del proyecto

En total, el presupuesto por personal del proyecto sería de 5995.80 euros.

2.5. Seguimiento

Debido a un imprevisto a la hora de hacer uso del protocolo de rastreo de contactos, hubo que ajustar los tiempos de la planificación.
En concreto, hubo que desarrollar un protocolo desde cero, por lo que los tiempos de programación de la aplicación tuvieron que aumentarse. Esto llevó a un desplazamiento de las actividades posteriores a esta y también a una disminución del tiempo disponible para realizarlas.

Tras esto, el diagrama de Gantt queda así:

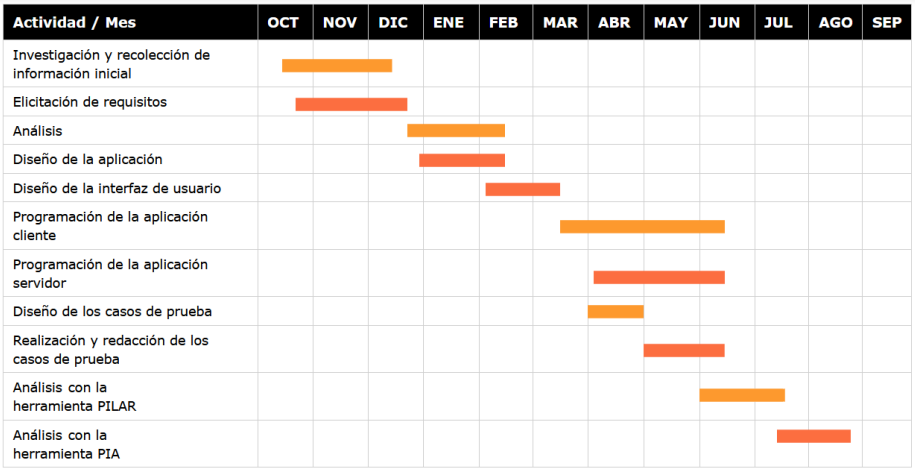


Figura 2: Planificación final

La planificación final queda distribuida de la siguiente manera:

- **Investigación y recolección de Información Inicial.** Desde la semana 1 a la semana 8. Se subdivide en:
  - **Investigación de protocolos de rastreo de contactos.**
  - **Investigación de la aplicación Radar Covid.**
  - **Investigación sobre estándares de privacidad.**
  - **Investigación sobre el protocolo DP-3T.**
- **Elicitación de requisitos.** Desde la semana 2 a la semana 10.
- **Análisis.** Desde la semana 10 a la semana 15. Se subdivide en:
  - **Investigación sobre los tipos de paradigmas en aplicaciones de rastreo de contactos.**
  - **Caso de uso.**
  - **Actores.**
  - **Diagrama de casos de uso.**
  - **Modelo de intercambio datos.**
- **Diseño de la aplicación.** Desde la semana 10 a la semana 16. Se subdivide en:

    - **Investigación sobre la generación de identificadores efímeros.**
    - **Diseño del protocolo.**
    - **Diseño de la base de datos del cliente.**
    - **Diseño de la base de datos del servidor.**
- **Diseño de la interfaz de usuario.** También se considera su programación. Desde la semana 15 a la semana 20.
- **Programación de la aplicación cliente.** Desde la semana 20 a la semana 31.
- **Programación de la aplicación servidor.** Desde la semana 24 a la semana 31.
- **Diseño de los casos de prueba.** Desde la semana 24 a la semana 28.
- **Realización y redacción de los casos de prueba.** Desde la semana 31 a la semana 34.
- **Análisis con la herramienta PILAR.** Desde la semana 33 a la semana 38.
- **Análisis con la herramienta PIA.** Desde la semana 33 a la semana 44.

Debido a este contratiempo, el camino crítico se vio afectado, y esto retrasó en parte a las tareas de **Programación de la aplicación cliente**, **Programación de la aplicación servidor**, **Diseño de los casos de prueba** y **Análisis con la herramienta PIA**.

Afortunadamente, se previó cierto margen de finalización, por lo que este problema pudo ser afrontado, apoyado además por la metodología empleada, *Pair Programming*, la cual admite que puedan ocurrir contratiempos como el sucedido.

3. Análisis de la aplicación

En este trabajo se ha llevado a cabo el desarrollo de una aplicación de rastreo de contactos, la cual interacciona con:

- **Usuarios.** Los cuales instalan una aplicación cliente en sus dispositivos móviles.
- **Autoridad Sanitaria.** La cual gestiona un servidor. Dado que aquí se ha desarrollado un prototipo, se simulan las funcionalidades básicas de un caso real.

Para comenzar el proyecto, primeramente se realizó una búsqueda de información sobre aplicaciones de rastreo de contactos existentes en el mercado, como son SwissCovid, COVIDSafe, Smittestopp o Radar COVID. El fin de dicho estudio ha sido comprender las necesidades y funcionamiento primordiales para poder deducir y aplicar los requisitos esenciales. En especial se toma en consideración aquellos relacionados con la privacidad del usuario debido al carácter médico y sensible de los datos.

Estudios como el realizado por Paul-Olivier Dehaye y Joel Reardon [26] y artículos como el escrito por Patrick Howell O'Neil [55], nos marcan que hay países, en este caso Suiza y Noruega, que consideran la privacidad como algo indispensable, hasta el punto de retirar del mercado sus aplicaciones de rastreo de contactos.

Debido al especial hincapié en dicha cuestión, estos se han tomado como guías esenciales para concluir en la fuerte necesidad de otorgar a la privacidad la merecida atención. A raíz de ello, se decidió tomar como referentes a diversos organismos legislativos que dictaminan requisitos de privacidad sobre este tipo de aplicaciones.

De esta investigación hemos obtenido como resultado la siguiente lista de requisitos para la aplicación. Estos se clasifican en funcionales, no funcionales, donde se incluyen los aspectos de privacidad.

3.1. Requisitos no funcionales

1. **Aplicación móvil.** Al ser una aplicación en la que se requiere recopilar con quién se tiene contacto, esta ha de ser portátil y por tanto debe instalarse en un teléfono móvil, ya que es un objeto que llevamos siempre encima.
2. **Anonimización de identificadores.** Con el fin de preservar la privacidad del usuario, este será identificado de manera anónima.
3. **El servidor debe avisar únicamente de los contagios activos.** Se considera contagio activo aquel cuya fecha es menor a la fecha de recepción en el servidor. Esto es con el fin de evitar almacenar identificadores asociados a individuos ya recuperados de la enfermedad.

3.1.1. Requisitos de seguridad

1. **Control de entrada de datos.** Se supervisarán las entradas de datos de la aplicación con el fin de evitar inyecciones de código.
2. «**Un mecanismo debe verificar el estado de los usuarios que notifican en la aplicación su condición de positivos en infección.** [...] Por ejemplo, facilitando un código de un solo uso vinculado con un laboratorio de pruebas o a un profesional de atención sanitaria. Si no se puede obtener confirmación de forma segura, no debe procederse al tratamiento de datos». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo Directrices 04/2020 sobre el uso de datos

de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-1, 2020) [13]

3. «**Los datos enviados al servidor central han de transmitirse a través de un canal seguro.** El uso de servicios de notificación prestados por proveedores de plataformas de sistema operativo debe evaluarse cuidadosamente y no debe dar lugar a la divulgación de ningún dato a terceros». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-2, 2020)

4. «**Las solicitudes no deben ser vulnerables a la manipulación por parte de un usuario malintencionado**». Para evitar falsos positivos. (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-3, 2020)

5. **Uso de técnicas criptográficas.** «Deben aplicarse las técnica criptográficas más avanzadas para asegurar los intercambios entre la aplicación y el servidor, y entre aplicaciones, y, como regla general, para proteger la información almacenada en las aplicaciones y en el servidor. Entre las técnicas que pueden utilizarse figuran, por ejemplo, las siguientes: cifrado simétrico y asimétrico, funciones hash, prueba privada de pertenencia (private membership test, PMT), intersección privada de conjuntos adoptadas 19 (private set intersection, PSI), filtros Bloom, recuperación de información privada, cifrado homomórfico, etc.» (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-4, 2020)

6. **El servidor central no debe conservar los identificadores de conexión a la red de ningún usuario.** «El servidor central no debe conservar los identificadores de conexión a la red (p. ej., las direcciones IP) de ningún usuario, incluidos los que han sido diagnosticados positivamente y que han transmitido su historial de contactos o sus propios identificadores». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-5, 2020)

7. **El servidor debe autenticar la aplicación y viceversa.** «Para evitar la suplantación o la creación de falsos usuarios, el servidor debe autenticar la aplicación». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-6, 2020)

8. «**La aplicación debe autenticar el servidor central**». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-7, 2020)

9. «**Las funcionalidades del servidor deben estar protegidas frente a ataques de repetición**». Para evitar suplantación de identidad y ataques de denegación de servicio. (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-8, 2020)

10. «**La información transmitida por el servidor central debe estar firmada para autenticar su origen e integridad**». Esto se logra mediante la criptografía asimétrica. (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-9, 2020)

11. **Administración de acceso al servidor.** «El acceso a todos los datos almacenados en el servidor central y que no estén a disposición del público debe circunscribirse a las personas autorizadas». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-10, 2020)

12. **Administración de permisos de la aplicación.** «El gestor de permisos del dispositivo en el nivel del sistema operativo solo debe solicitar los permisos necesarios para acceder a los módulos de comunicación y utilizarlos cuando resulte necesario, para almacenar los datos en el equipo terminal y para intercambiar información con el servidor central». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo SEC-11, 2020)

3.1.2. Requisitos de privacidad

1. **Minimalidad de los datos.**«Los intercambios de datos deben respetar la privacidad de los usuarios (y, en particular, el principio de minimización de datos)». Para respetar la privacidad de los datos, la aplicación se limitará a trabajar con datos que no permitan averiguar la identidad del usuario, tales que los identificadores efímeros o el intercambio de semillas generadoras con el servidor para que no viajen los identificadores por la red. (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-1, 2020)

   - **Datos que se almacenan.**
     - La información anonimizada que identifica al usuario.
     - La información de los contactos cercanos.
     - Datos referentes a permisos de la aplicación.
       - android.permission.INTERNET: permite comunicarse con el backend del servidor.
       - android.permission.ACCESS_NETWORK_STATE: permite a la aplicación saber si el dispositivo está conectado a internet.
       - android.permission.BLUETOOTH: usado para poder comunicarse entre dispositivos móviles. [42]
       - android.permission.REQUEST_IGNORE_BATTERY _OPTIMIZATIONS: permite que la aplicación se ejecute en cualquier momento en segundo plano, permitiendo que las sincronizaciones entre la aplicación y el servidor sucedan en los momentos oportunos.
   - **Datos que se envían**
     - **De cliente a servidor de la entidad sanitaria.**
       - Prueba de haber dado positivo en un diagnóstico.
       - De manera opcional la fecha de síntomas o de prueba diagnóstica positiva.
       - Información anonimizada que representa al usuario. De esta forma se respeta la privacidad del usuario, pues no se puede asociar dicho positivo a ningún identificador.
       - Tráfico de paquetes disuasorios para evitar que agentes externos identifiquen usuarios infectados, pues de otro modo la comunicación cliente dirección servidor solo se da en caso de positivo.
     - **De servidor a cliente.**
       - Información anonimizada de usuarios infectados. Esta la recibe la aplicación cliente y puede contrastar si ha estado en contacto o no con algún positivo. En caso de coincidir se avisa de posible contacto de riesgo. Estas se envían mediante broadcast a todos los clientes para evitar distinciones entre usuarios que pudieran afectar a su privacidad.
     - **Entre clientes.**
       - Información que identifique que han estado en contacto.

2. **Evitar que la aplicación identifique o rastree a los usuarios.** «La aplicación no puede permitir identificar directamente a los usuarios al utilizar la aplicación». (European Data Protection Board,

Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-2, 2020)

3. «**La aplicación no ha de permitir que se rastreen los movimientos de los usuarios**». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-3, 2020)

4. «**El uso de la aplicación no debe permitir que los usuarios obtengan información de otros usuarios (y, en particular, que sepan si son o no portadores del virus)**». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-4, 2020)

5. **La confianza en el servidor debe ser limitada.** «La gestión del servidor central debe seguir normas de gobernanza claramente definidas e incluir todas las medidas necesarias para garantizar su seguridad. La ubicación del servidor central debe permitir una supervisión eficaz por parte de la autoridad supervisora competente». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-5, 2020)

6. «**Ha de llevarse acabo una evaluación de impacto relativa a la protección de datos, que debería ponerse a disposición del público**». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-6, 2020)

7. «**La aplicación solo debe revelar al usuario si ha estado expuesto al virus y, en la medida de lo posible, sin facilitar información sobre otros usuarios, el número de veces y las fechas de la exposición**». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-7, 2020)

8. «**La información transmitida por la aplicación no debe permitir a los usuarios identificar a los usuarios portadores del virus ni conocer sus movimientos**». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-8, 2020)

9. **Preservar la identidad de los usuarios frente a las autoridades sanitarias.** «La información transmitida por la aplicación no debe permitir a las autoridades sanitarias identificar a los usuarios que pueden estar expuestos sin el consentimiento de estos». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-9, 2020)

10. «**Las solicitudes cursadas por la aplicación al servidor central no deben revelar ninguna información sobre el portador del virus.**» (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-10, 2020)

11. «**Las solicitudes cursadas por la aplicación al servidor central no deben revelar ninguna información innecesaria sobre el usuario, excepto, posiblemente —y solo cuando resulte necesario—, sus identificadores seudónimos y su lista de contactos**». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-11, 2020)

12. «**Impedir ataques de enlace**». Para evitar el robo de datos de los usuarios. (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-12, 2020)

13. «**Los usuarios han de poder ejercer sus derechos a través de la aplicación**». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-13, 2020)

14. «**La supresión de la aplicación debe entrañar la eliminación de todos los datos recogidos a nivel local**». Esto incluiría la lista de identificadores de los contactos, así como los identificadores propios y las semillas recibidas por el servidor al hacer broadcast. (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-14, 2020)

15. «**La aplicación solo puede recoger datos transmitidos por instancias de la aplicación o de aplicaciones interoperables equivalentes.** No pueden recogerse datos sobre otras aplicaciones ni otros dispositivos de comunicación de proximidad». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-15, 2020)

16. **Implementación de servidores proxy.** «Para evitar la reidentificación por parte del servidor central, deben implementarse servidores proxy. La finalidad de estos servidores no colusores es combinar los identificadores de varios usuarios (tanto los de los portadores del virus como los enviados por los solicitantes) antes de compartirlos con el servidor central, para evitar que este conozca los identificadores de los usuarios (como las direcciones IP)». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-16, 2020)

17. «**La aplicación y el servidor deben desarrollarse y configurarse cuidadosamente con el fin de que no recojan datos innecesarios.** (P. ej., no debe incluirse ningún identificador en los registros del servidor, etc.) y de evitar el uso de SDK de terceros que recojan datos para otros fines». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo PRIV-17, 2020)

18. **Salvaguardar la privacidad de la lista de contactos de cara al servidor.** El servidor no invade la lista de contactos de los clientes.

19. **Evitar la identificación de los usuarios a partir de la prueba de diagnóstico positivo.** Esto se evita haciendo que dicha prueba sea solamente asociada al identificador anónimo y desde el servidor se envíe la notificación a todos los clientes.

20. **Evitar el acceso al IMEI del dispositivo móvil.** Pues mediante este número de 15 dígitos, se identifica a un terminal cuando se conecta a una red móvil.

21. **Evitar el acceso a la dirección MAC de Bluetooth del móvil.** Pues este identificador único se emplea durante conexiones Bluetooth.

22. **Evitar el acceso a la dirección IPv4 y MAC de la tarjeta WiFi.** Pues pueden utilizarse para identificarnos al conectarnos a la red.

El desarrollo de las aplicaciones de rastreo de contactos puede realizarse desde dos enfoques. Dependiendo de la opción elegida en la parte de diseño, tras analizar el estado del arte y los diferentes protocolos existentes para su desarrollo, se aplicarán un grupo de los siguientes requisitos.

En el caso de que un usuario se declarase infectado, la aplicación envía a un servidor el historial de los contactos de proximidad que se han obtenido mediante escaneo, se aplicarán los siguientes requisitos:

3.1.3. Requisitos que se aplican únicamente cuando la aplicación envía al servidor una lista de contactos

1. «El servidor central debe recoger el historial de contactos de los usuarios declarados positivos [...] como resultado de una acción voluntaria por parte de estos». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo CON-1, 2020)
2. «El servidor central no debe mantener ni difundir una lista de los identificadores seudónimos de usuarios portadores del virus». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo CON-2, 2020)
3. «El historial de contactos almacenado en el servidor central debe eliminarse una vez se haya notificado a los usuarios su proximidad a una persona con diagnóstico positivo». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo CON-3, 2020)
4. «Excepto si un usuario detectado como positivo comparte su historial de contactos con el servidor central o si un usuario solicita al servidor que investigue su posible exposición su posible exposición al virus, ningún dato debe salir del equipo del usuario». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo CON-4, 2020)
5. «Cualquier identificador incluido en el historial local debe eliminarse a los X días de su recogida (corresponde a las autoridades sanitarias definir el valor X)». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo CON-5, 2020)
6. «Los historiales de contactos enviados por distintos usuarios no deben someterse a tratamiento adicional; por ejemplo, no debe examinarse su correlación cruzada para elaborar mapas globales de proximidad». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo CON-6, 2020)
7. «Los datos contenidos en los registros del servidor han de minimizarse y deben cumplir los requisitos de protección de datos». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo CON-7, 2020)

Por el contrario si en el caso de que un usuario se declarase infectado la aplicación envía a un servidor la lista de sus propios identificadores difundidos, se aplicarán los siguientes requisitos:

3.1.4. Requisitos que se aplican únicamente cuando la aplicación envía al servidor una lista de sus propios identificadores

1. «El servidor central debe recoger los identificadores de los usuarios declarados positivos [...] difundidos por la aplicación, como resultado de una acción voluntaria por parte de estos». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo ID-1, 2020)
2. «El servidor central no debe mantener ni difundir el historial de contactos de usuarios portadores del virus». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo ID-2, 2020)

3. «Los identificadores almacenados en el servidor central deben eliminarse una vez distribuidos a las demás aplicaciones». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo ID-3, 2020)
4. «Excepto si un usuario detectado como positivo comparte sus identificadores con el servidor central o si un usuario solicita al servidor que investigue su posible exposición al virus, ningún dato debe salir del equipo del usuario». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo ID-4, 2020)
5. «Los datos contenidos en los registros del servidor han de minimizarse y deben cumplir los requisitos de protección de datos». (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Anexo ID-5, 2020)

3.1.5. Requisitos de usabilidad

Uno de los aspectos más importantes de esta aplicación es que sea accesible para la gran mayoría de las personas debido a su estrecha relación con la salud. El hecho de que en uno de los puntos de la aplicación se requiera que los usuarios comuniquen una prueba de contagio y detecten los contactos que vayan teniendo (explicado más adelante en Requisitos Funcionales), convierte en una necesidad que una gran mayoría de la población haga uso de la aplicación. Esto, por tanto, obliga a que la interfaz de usuario tenga que estar pensada y diseñada a conciencia.

Los requisitos que debe cumplir el diseño de la interfaz son los siguientes:

1. **La interfaz debe ser sencilla.**
2. **La interfaz debe ser suficientemente intuitiva.** Esto significa que se hará uso de símbolos que se relacionen rápidamente por analogía y costumbre de uso a acciones concretas.
3. **La interfaz debe ser agradable a la vista.**
4. **La interfaz debe ser accesible para personas con daltonismo.** Se ha empleado un simulador para determinar una paleta de colores apropiada. [78]
5. **El texto debe ser claro y conciso.**
6. **El texto debe ser legible para personas dentro de un rango de edad de entre 11 y 75 años, y con discapacidades visuales tales como el daltonismo o presbicia.** En caso de no poderse agrandar más el tamaño de la letra, se hará uso de símbolos que evoquen analogías con el concepto escrito.
7. **La interfaz debe invitar a publicitar su uso.** La interfaz incorporará detalles atractivos e identificativos originales de la aplicación.
8. **La interfaz debe concienciar sobre los síntomas del estado de exposición del usuario.**
9. **Internacionalización.** La interfaz será accesible en diversos idiomas.

3.2. Requisitos Funcionales

Debido a la naturaleza del proyecto, donde habrá una interacción entre distintos despliegues de la aplicación, se ha realizado una distinción entre los requisitos que llegan al usuario y los que ven otros dispositivos.

3.2.1. De cara al usuario

1. **Visualización de mi riesgo de exposición.** El usuario debe recibir retroalimentación visual sobre si ha estado expuesto o no a otro usuario contagiado y en qué medida.
2. **Posibilidad de comunicación de contagio.** Tanto a las autoridades sanitarias como a contactos cercanos.
3. **Notificación y muestra del nivel de exposición a un contagio.** El usuario debe recibir un aviso en el caso de haber tenido un contacto cercano, así como información sobre el nivel de riesgo del mismo.
4. **La aplicación dará directrices y recomendaciones al usuario.** Dependiendo de su nivel de riesgo, la aplicación dará unas u otras recomendaciones acorde a la enfermedad.

3.2.2. De cara a otros dispositivos

1. **Aviso de foco de contagio.** Habrá de existir algún tipo de información con el fin de que se pueda identificar que dos dispositivos han estado cerca. Esto es para, en caso de que uno esté contagiado, se pueda avisar a sus contactos.
2. **Permitir la identificación de dispositivos.** Para llevar un seguimiento de los contactos cercanos.
3. **Medición del tiempo de exposición a un contacto.** A partir de cierto intervalo de exposición se considerará un contacto directo.
4. **Informar de un contagio.** Se deberá notificar en caso de contagio a los dispositivos que hayan estado en contacto al usuario.
5. **Cálculo de la distancia entre dispositivos.** La aplicación será capaz de determinar la distancia con otro dispositivo con el fin de determinar la existencia y nivel de riesgo.
6. **Envío de información de contagio al servidor de la autoridad sanitaria.** Una vez allí se ha de comprobar y verificar su validez.
7. **Comunicación de dispositivos contagiados.** El servidor deberá comunicar periódicamente a los clientes la lista de nuevos contagios.

3.2.3. De cara a la propia aplicación

1. **Uso de información anonimizada como identificación del usuario.** Se deberá generar tal que sea imposible relacionarlos con el usuario al que esté asociada.
2. **Cambio de estado de exposición.** Con el objetivo de proporcionar al usuario una aproximación del posible riesgo de contagio al que se ha visto expuesto.

3.3. Requisitos de Información

Información almacenada localmente en el dispositivo del usuario:

1. **Identificación anónima asociada a cada usuario.** Esta deberá preservar la privacidad de la información asociada al usuario a la vez que permitirá el seguimiento de su estado de salud.
2. **Información anonimizada sobre qué usuarios han estado en contacto.** Dado que no es necesario saber con quién se ha estado sino solamente si se ha estado en contacto con usuarios infectados.

3. **Información sobre el estado COVID del usuario.** Esto es si tiene o no el virus, o su ha podido estar en contacto con alguien contagiado.

Información almacenada en el servidor.

1. **Información anonimizada asociada a los usuarios infectados.** Con el fin de evitar el almacenamiento de la información sensible, en su lugar se almacenará en el servidor una identificación anonimizada de los usuarios.
2. **Fecha de recepción.** Se almacenará la fecha de recepción de cada comunicado al servidor con el fin de eliminar aquella información extinta pasados los 14 días que dura la infección.

3.4. Base de Datos - Modelo conceptual

Como se ha especificado antes, en esta aplicación es necesario el almacenamiento de cierta información tanto en los clientes como en el servidor. Por esta razón se debe estructurar una forma de almacenamiento de dichos datos, la cual permita organizarlos y tenerlos a disposición.

La existencia de una aplicación cliente y otra servidor obliga a diferenciar dos sistemas de almacenamiento, cada uno orientado a sus necesidades específicas.

Es por ello que se procede a modelar dos bases de datos, cada una dedicada a una parte del proyecto, es decir, cliente y servidor.

Modelo Conceptual

Los datos que se manejan en esta aplicación son los identificadores anonimizados. Estos consisten en un valor asociado de manera anónima a un individuo, pero que permitan conocer sus contactos cercanos para poder avisarlos en caso de foco de contagio; y su estado COVID. En concreto, han de poseer los siguientes atributos.

- **Valor del identificador.** Esta información identificará de manera anónima a los usuarios, pudiendo determinar sus contactos cercanos y estado.
- **Fecha de recepción.** Es la fecha de recepción del identificador en un dispositivo. Se emplea para llevar cuenta de los catorce días que está activo el virus en ese contacto.

Por lo tanto, un identificador sería tal que así:

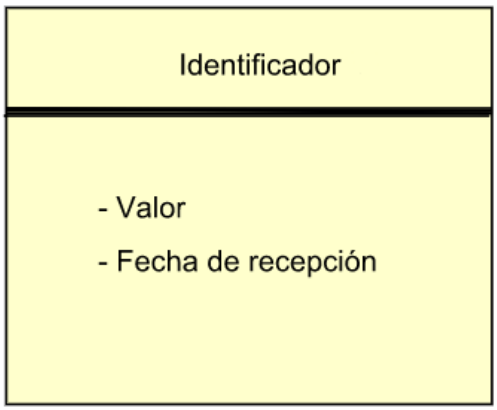


Figura 3: Modelo conceptual

3.5. Elección de paradigma de protocolos

Debido a la necesidad de acotar la manera de tratar los datos de los usuarios, así como el funcionamiento de la aplicación, es necesario recopilar información sobre los distintos paradigmas existentes para el desarrollo de aplicaciones de rastreo de contactos.

En este ámbito encontramos dos enfoques principales.

- **Intercambio de Identificadores.** Este enfoque se basa en el uso de identificadores efímeros. Los identificadores efímeros son códigos alfanuméricos que identifican a un usuario anónimamente durante un cierto periodo de tiempo, tras el cual se desechan por otros. Estos se generan haciendo uso de algún algoritmo criptográfico, el cual calcula los valores a partir de semillas generadoras (claves criptográficas, fechas...). Las aplicaciones cliente de los usuarios se intercambian estos identificadores con el fin de registrar los contactos.
- **Geolocalización.** Por otro lado, en este paradigma se emplean técnicas de geolocalización, tal que de haber un usuario contagiado se alerta a aquellos que hayan estado en las mismas zonas que él a la vez. Este enfoque es mucho más invasivo que el anterior para la privacidad de los usuarios, pues se hace uso de un dato sensible.

Los datos de geolocalización pueden revelar mucha información personal del usuario a partir de los lugares que visita, como son por ejemplo su dirección personal o la de su trabajo, pero también datos sobre su religión si visita algún edificio de culto, o incluso sobre su orientación sexual determinado por algunos lugares que pudiera visitar.

EL EDPB (European Data Protection Board) establece que los datos de geolocalización no deberían requerirse excepto si son de absoluta necesidad. En concreto dicta que «El seguimiento sistemático y masivo de la localización o los contactos de las personas físicas es una grave injerencia en su privacidad. Esta práctica solo puede legitimarse sobre la base de su adopción voluntaria por parte de los usuarios para cada uno de los fines respectivos, lo que implica, entre otras cosas, que las personas que decidan no utilizar esas aplicaciones, o no sepan hacerlo, no deben sufrir ninguna desventaja.» (European Data Protection Board, Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, Párrafo 24)

Debido a todo ello **se selecciona el paradigma que hace uso de identificadores**, y todo el desarrollo se orientará hacia este en lugar de al de geolocalización.

3.6. Base de Datos - Modelo lógico

Se procede a realizar el modelo lógico, acotando mejor los datos a almacenar una vez elegido el paradigma.

Modelo Lógico

Debido a las características de este proyecto se han considerado dos bases de datos. Una de ellas será la base de datos local, donde se almacenarán los identificadores tanto propios como los ajenos, es decir, los obtenidos vía Bluetooth.

Por otro lado, la base de datos del servidor almacenará aquellas claves y fechas generadoras asociadas a identificadores contagiados. Tan solo retendrá las que haya recibido en los últimos catorce días, tiempo durante el que el virus permanece activo.

La base de datos local tendrá diferenciados los identificadores propios de los ajenos. Esto es debido a que poseen distintos atributos, y de este modo, se evitará el guardado de múltiples campos con valor *NULL* o vacío. En concreto cada tabla contiene los siguientes atributos:

- **Identificadores propios.**
  - **Valor.** Es el número que define propiamente al identificador.
  - **Clave generadora.** Es uno de los parámetros necesarios para la generación del identificador.
  - **Fecha generadora.** Es uno de los parámetros necesarios para la generación del identificador.
- **Identificadores ajenos.**
  - **Valor.** Es el número que define propiamente al identificador. Se recibe en el intercambio vía Bluetooth.
  - **Fecha de recepción.** Es la fecha de recepción en el dispositivo del valor del identificador.

Así pues, queda de la siguiente forma:

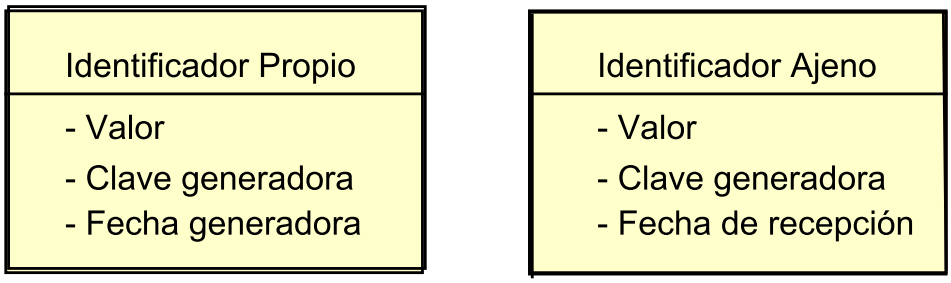


Figura 4: Modelo lógico Cliente

Por otro lado, la base de datos del servidor almacena aquella información asociada a los identificadores contagiados. Contendrá la clave y fecha generadoras de cada uno de ellos, así como el momento de recepción de cada uno. Esto último es con el fin de ir eliminando aquellos que posean una fecha extinta, es decir, hayan transcurrido catorce días.

- **Clave generadora.** Es uno de los parámetros necesarios para la generación del identificador.

- **Fecha generadora.** Es uno de los parámetros necesarios para la generación del identificador.
- **Fecha de recepción.** Es la fecha de recepción en el servidor de los datos del identificador contagiado.

De este modo, quedaría de la siguiente forma:

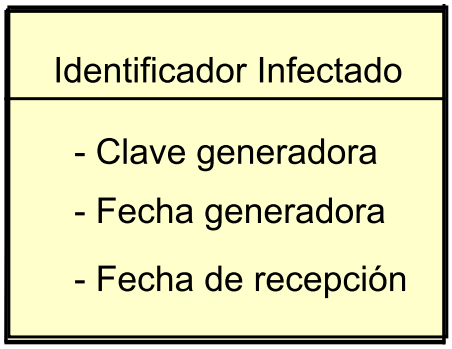


Figura 5: Modelo lógico Servidor

3.7. Modelo de intercambio de datos

Los datos serán intercambiados entre los siguientes involucrados:

- **Aplicación cliente.** Estos realizarán el intercambio de identificadores actuando entre ellos como cliente y servidor alternando los roles.
- **Servidor.** El servidor recibe de un cliente contagiado el código y las semillas asociadas a sus identificadores. Además, de manera aleatoria, todos los clientes envían datos análogos pero falsos con el fin de generar ruido. A la hora de interactuar el servidor con los clientes para enviar los datos, se realiza un broadcast o multicast con el fin de salvaguardar la privacidad al no hacer diferenciación entre clientes.

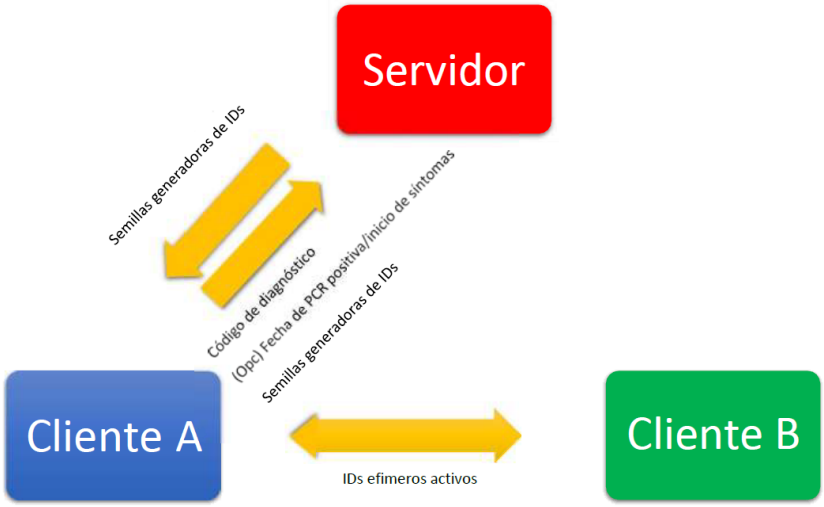


Figura 6: Modelo de intercambio de datos

3.8. Desarrollo de los casos de uso

Tras un análisis de los requisitos elicitados en este documento, hemos llegado a la conclusión de que nuestro sistema implica la actuación de cinco actores:

- **Usuario.** Sería el actor principal, es decir, a quien va pensada la funcionalidad de la aplicación.
- **Servidor.** Sirve como punto de contacto con la autoridad sanitaria pertinente. Su función principal es la gestión de los datos referentes a los identificadores efímeros y los diagnósticos, tanto su envío y recepción como el borrado de registros antiguos o no vigentes.
- **Timer.** Es un representación del paso del tiempo, el cual desencadena aquellos casos de uso que se llevan a cabo según un periodo temporal. En segundo plano, envía al servidor de manera periódica códigos falsos. Esto se hace con la finalidad de generar ruido, pues de otro modo podría detectarse quién está infectado ya que el envío de datos dirección cliente servidor solo se haría al comunicar un contagio.

- **Sensor Bluetooth servidor.** Se trata de un actor que actúa como parte de la comunicación Bluetooth, el cual es el encargado de recibir los identificadores enviados por el Sensor Bluetooth cliente de otros dispositivos implicados en el intercambio.
- **Sensor Bluetooth cliente.** Al igual que el anterior, se trata de un actor que actúa como parte de la comunicación Bluetooth. En cambio, la función de este es transmitir los identificadores a recibir por el Sensor Bluetooth servidor del resto de dispositivos implicados en el intercambio.

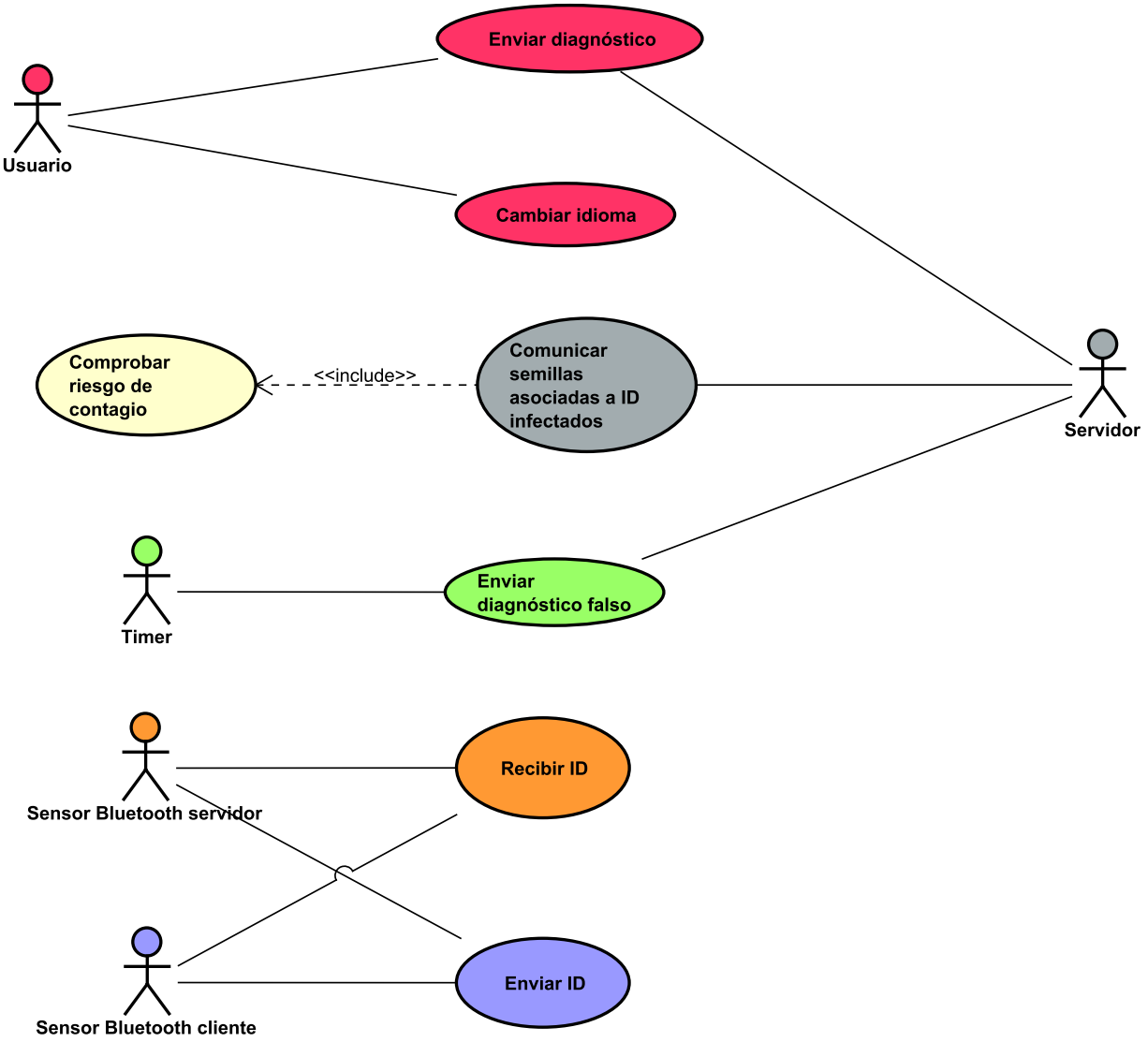


Figura 7: Diagrama de casos de uso

Como se puede apreciar en el diagrama anterior los casos de uso serían los siguientes:

3.8.1. UC-01 Enviar diagnóstico

ELEMENTO	VALOR
Caso de Uso	Enviar diagnóstico
Resumen	Se realiza este caso de uso cuando el actor Usuario quiere comunicar su contagio.
Actor	Usuario, Servidor
Precondición	Tener conexión a Internet.
Postcondición	Las semillas generadoras del actor Usuario aparecen como infectadas en el servidor. La aplicación (*usuario*) cambia su estado a infectado.
Secuencia Base	1- El actor Usuario introduce el código de diagnóstico proporcionado por la autoridad sanitaria. 2- El sistema pide confirmación al usuario. 3- El actor usuario verifica la acción. 4- El sistema envía el código al servidor. 5- El sistema le comunica al actor Usuario que el código es correcto y le informa de las medidas que debe tomar(cambiar pantalla inicial). 6- El sistema envía las semillas generadoras de los ID del actor Usuario al servidor.
Secuencia Alternativa	
Excepciones	3'- El actor usuario no verifica la acción y se vuelve al paso 1. 5'- El sistema comunica un error de conexión y el caso de uso queda sin efecto. 5"- El sistema comunica al actor Usuario que el código es incorrecto y el caso de uso queda sin efecto.
Sub Caso de Uso	

Tabla 2: Caso de uso: Enviar diagnóstico

3.8.2. UC-02 Cambiar idioma

ELEMENTO	VALOR
Caso de Uso	Cambiar idioma
Resumen	Se realiza este caso de uso cuando el actor Usuario quiere cambiar el idioma de la aplicación.
Actor	Usuario
Precondición	
Postcondición	El idioma de la aplicación ha cambiado al seleccionado por el actor Usuario.
Secuencia Base	1- El actor Usuario selecciona un idioma entre los disponibles. 2- El sistema le pide confirmación al actor Usuario. 3- El actor Usuario confirma su selección. 4- El sistema aplica los cambios sobre la aplicación.
Secuencia Alternativa	
Excepciones	3'- El actor Usuario no confirma la acción y el caso de uso queda sin efecto.
Sub Caso de Uso	

Tabla 3: Caso de uso: Cambiar idioma

3.8.3. UC-03 Comunicar semillas asociadas a IDs infectados

ELEMENTO	VALOR
Caso de Uso	Comunicar semillas asociadas a ID infectados
Resumen	Se realiza un broadcast del contenido de la base de datos del servidor de forma periódica con el fin de notificar posibles contagios a los clientes.
Actor	Servidor
Precondición	Debe haber pares clave - fecha marcados como infectados en la base de datos. Tener conexión a Internet
Postcondición	Los clientes obtienen los pares clave - fecha necesarios para generar los ID's infectados.
Secuencia Base	1- El actor Servidor envía los pares infectados a todos los clientes. 2- Se realiza el caso de uso *Comprobar riesgo de contagio*
Secuencia Alternativa	
Excepciones	
Sub Caso de Uso	Comprobar riesgo de contagio

Tabla 4: Caso de uso: Comunicar semillas asociadas a IDs infectados

3.8.4. UC-04 Comprobar riesgo de contagio

ELEMENTO	VALOR
Caso de Uso	Comprobar riesgo de contagio
Resumen	Este caso de uso se realiza para comprobar si ha habido algún contacto con algún usuario infectado.
Actor	
Precondición	Debes haber recibido pares clave - fecha infectados.
Postcondición	El sistema muestra el riesgo de contagio más alto entre los identificadores comprobados.
Secuencia Base	1- El sistema genera los ID's asociados a los pares recibidos. 2- El sistema compara los ID's generados con aquellos obtenidos mediante intercambio Bluetooth. 3- El sistema muestra el riesgo de contagio más alto de entre los ID's coincidentes.
Secuencia Alternativa	
Excepciones	3'- Ningún ID generado coincide el caso de uso queda sin efecto.
Sub Caso de Uso	

Tabla 5: Caso de uso: Comprobar riesgo de contagio

3.8.5. UC-05 Enviar diagnóstico falso

ELEMENTO	VALOR
Caso de Uso	Enviar diagnóstico falso
Resumen	Este caso de uso se realiza un número aleatorio de veces a lo largo del dia
Actor	Timer, Servidor
Precondición	Tener conexion a Internet.
Postcondición	Se genera tráfico falso que realiza la función de ruido en la red con la finalidad de prevenir ataques pasivos de escucha.
Secuencia Base	1- El actor Timer genera unas claves falsas y una fecha aleatoria dentro de los últimos 15 días. 2- El sistema genera un código de diagnóstico falso. 3- El sistema envía el código al servidor. 4- El sistema envía las claves y fechas generadoras de los ID del actor Usuario al servidor.
Secuencia Alternativa	
Excepciones	3'- El sistema comunica un error de conexión y el caso de uso queda sin efecto.
Sub Caso de Uso	

Tabla 6: Caso de uso: Enviar diagnóstico falso

3.8.6. UC-06 Recibir ID

ELEMENTO	VALOR
Caso de Uso	Recibir ID
Resumen	Este caso de uso se realiza cuando el actor Sensor Bluetooth servidor detecta una señal Bluetooth de otro dispositvo para recibir su ID.
Actor	Sensor Bluetooth servidor, Sensor Bluetooth cliente
Precondición	El dispositivo debe tener el Bluetooth y la geolocalización activados.
Postcondición	El ID recibido es almacenado en la base de datos del sistema.
Secuencia Base	1- El actor Sensor Bluetooth servidor detecta una señal Bluetooth (UUID IGUAL) de la aplicación. 2- El sistema inicia un contador de 15 min. 3- El actor Sensor Bluetooth servidor recibe del actor Sensor Bluetooth cliente el ID efímero correspondiente a ese periodo de tiempo. 4- El sistema almacena el ID recibido en la base de datos junto con la intensidad de señal recibida.
Secuencia Alternativa	
Excepciones	3' - Se deja de detectar señal Bluetooth antes de agotar los 15 min y el caso de uso queda sin efecto.
Sub Caso de Uso	

Tabla 7: Caso de uso: Recibir ID

3.8.7. UC-07 Enviar ID

ELEMENTO	VALOR
Caso de Uso	Enviar ID
Resumen	Este caso de uso se realiza cuando el actor Sensor Bluetooth cliente detecta una señal Bluetooth de otro dispositvo para enviar su propio ID.
Actor	Sensor Bluetooth cliente, Sensor Bluetooth servidor
Precondición	El dispositivo debe tener el Bluetooth y la geolocalización activados.
Postcondición	El ID efímero del periodo de tiempo correspondiente se envía al Sensor Bluetooth servidor.
Secuencia Base	1- El actor Sensor Bluetooth cliente detecta una señal Bluetooth (UUID IGUAL) de la aplicación. 2- El sistema inicia un contador de 15 min. 3- El actor Sensor Bluetooth cliente envía al actor Sensor Bluetooth servidor el ID efímero correspondiente a ese periodo de tiempo.
Secuencia Alternativa	
Excepciones	3' - Se deja de detectar señal Bluetooth antes de agotar los 15 minutos y el caso de uso queda sin efecto.
Sub Caso de Uso	

Tabla 8: Caso de uso: Enviar ID

4. Diseño de la aplicación

Partiendo de los requisitos anteriormente elicitados, se deben tomar ciertas decisiones importantes de diseño. Para ello, se acotan las soluciones que vamos a implementar para los problemas que han sido presentados anteriormente en los requisitos.

4.1. Estado del Arte: Protocolos

El protocolo de comunicación de exposición es una de las partes principales de la aplicación.

Se recogen a continuación los más relevantes. [44]

4.1.1. Decentralized Privacy-Preserving Proximity Tracing (DP-3T)

Se trata de un protocolo descentralizado de código abierto. [5] Este protocolo se basa en la generación de identificadores efímeros, los cuales se intercambian cuando dos clientes se encuentran a una distancia inferior a 2 metros y durante más de 15 minutos de exposición.

El proceso para generar estos identificadores, ilustrado en la Figura 8 es el siguiente:

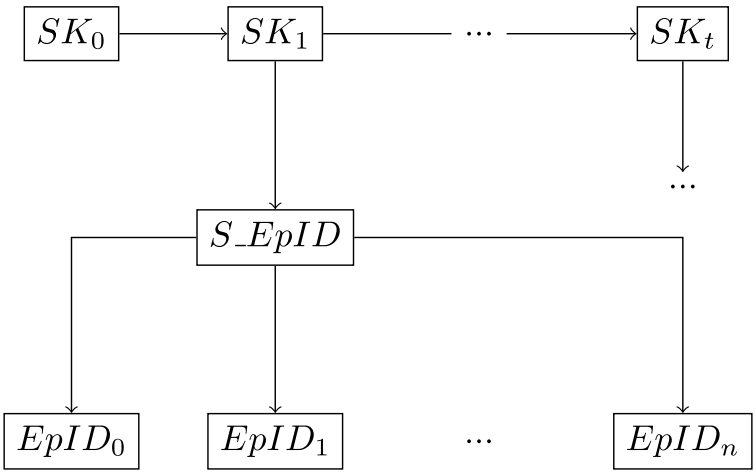


Figura 8: Esquema de generación de identificadores efímeros

1. **Generación de SK_t.** O Secret Key (clave secreta) número t. Inicialmente se genera una clave secreta SK_t correspondiente al día t actual. Esta clave se obtiene a partir del resumen hash SHA-256 de la clave SK_{t-1}. La generación de la primera SK_0 se hace mediante el algoritmo de curvas de Edward Ed25519.

2. **Generación del S_EphID.** O Secret Ephemeral IDentifier (identificador efímero secreto). A partir del SK_t del día se genera el S_EphID empleando una función:

$$S\_EphID(BK) = PRG(PRF(SK_t, BK))$$

   Donde PRG es un cifrado de flujo que produce n*16 bytes, siendo n el número de identificadores diarios. El número n se determina tal que n=(2460)/l, siendo l el tiempo de vida en minutos de un identificador efímero. PRF es una función pseudoaleatoria de la forma HMAC-SHA256 y BK es una variable global.

3. **Generación de $\mathbf{EphID_n}$** Tras ello el S_EphID se subdivide en n fragmentos de tamaño 16 bytes. Cada uno de estos fragmentos es un EphID cuyo orden de uso se determina de forma aleatoria.

En concreto, en DP-3T el tiempo de uso para su intercambio de un identificador efímero es de 15 minutos.

Cuando dos usuarios se encuentran, las aplicaciones móviles comienzan a actuar entre ellas como servidor-cliente, intercambiando los roles, para de esta manera enviarse mutuamente los identificadores.

Cuando se produce un contagio, el usuario envía un código de verificación que previamente la autoridad sanitaria le ha proporcionado. Este código se envía junto a las semillas generadoras. Esta información recibida por el servidor es enviada de forma periódica a los clientes. Así, las aplicaciones cliente pueden calcular los identificadores contagiados a partir de las semillas generadoras recibidas desde el servidor. Si alguno de estos identificadores generados coincide con uno de los almacenados significa que ha habido una exposición a contagio y el protocolo avisa al usuario de ello a través de la aplicación cliente.
Al enviar las semillas generadoras, se preserva la privacidad de los usuarios contagiados, pues los identificadores no son enviados nunca como tal. [24] [30] [49]
Este protocolo se creó para apoyar el protocolo GAEN, funcionando sobre él aunque con algunos cambios a nivel de tratamiento y a la hora de crear las semillas generadoras y los identificadores.

4.1.2. (Google/Apple) Exposure Notification (GAEN) system

Originalmente conocido como *Privacy-Preserving Contact Tracing Project*.
Este protocolo emplea un enfoque descentralizado. Fue creado con el fin de que existiera una comunicación entre dispositivosAndroid e iOS. Sin embargo no es compatible con los dispositivos *Huawei* posteriores a mayo de 2019. Está implementado a nivel de sistema operativo para de esta forma ser más eficiente al realizar todos los procesos en segundo plano. Funciona de manera muy similar a DP-3T, empleando identificadores efímeros (EphIDs), los cuales cambian cada 15-20 minutos (al resetearse la MAC Bluetooth del dispositivo). Estos son calculados mediante una clave AES y una marca de tiempo calculada a partir de Unix Epoch Time.

Cuando se produce un contagio, desde la aplicación cliente se suben al servidor las semillas generadoras. De esta forma, el servidor puede reenviar esa información a los demás usuarios y estos generar los identificadores correspondientes. Si alguno coincidiera con uno almacenado, el protocolo avisa a través de la aplicación cliente de que se ha estado expuesto a un posible contagio.[31]

La diferencia principal con DP-3T se encuentra en la generación de las claves secretas SK. En DP-3T estas claves se obtienen a partir de un resumen hash de la clave SK del día anterior. Sin embargo, en GAEN todas las claves secretas son generadas a partir del mismo inicializador.

Otra diferencia reside en el sello temporal empleado para generar esos identificadores. DP-3T utiliza una marca de tiempo más basta o un resumen hash de la misma. Por ello garantiza la privacidad mejor que GAEN, aunque a costa de ser más vulnerable ante los ataques de repetición, pues son más fáciles de llevar a cabo debido a un período de validez más largo (ya que las estampas temporales son menos precisas y por lo tanto hay más décimas de tiempo entre ellas). [28]

4.1.3. Pan-European Privacy-Preserving Proximity Tracing (PEPP-PT/PEPP)

Este protocolo es descartado debido a la necesidad de registro en el servidor, medida tomada para evitar multicuentas. Esto se hace mediante datos personales pseudónimos que se usan para generar el identificador PUID. Dicho identificador es necesario para que el servidor asocie dicho dispositivo y sea capaz de enviarle los datos pertinentes ante el registro de casos positivos. El PUID se emplea junto a una clave global, la cual cambia cada 60 minutos, para generar en el servidor los ID efímeros EBID. Estos se envían mediante

broadcast a los clientes de la aplicación. Dichas claves globales se eliminan a las 4 semanas.
El hecho de que el servidor sea capaz de obtener los PUID originales mediante la clave y los EBID, lo convierte en un sistema con gran capacidad para identificar a usuarios, de divulgación completa y correlación. Esto le convierte en un objetivo con un riesgo muy alto, pues es posible reidentificar a los usuarios mediante los PUID y las claves, ya que estas se quedan almacenadas durante bastante tiempo.

Por otro lado, los EBID permiten el rastreo en tiempo real de los usuarios, infectados o no. Esto es debido a que el servidor *backend* permite su conversión en identificadores permanentes, relacionando cualquier reporte o interacción del portador con dispositivos y sensores Bluetooth al PUID original.

Además, debido a que no existe una metodología de identificación y autenticación de los EBID, es posible generarlos de manera falsa, asociándolos a un usuario de manera externa. Esta amenaza puede concretarse en un ataque de un tercero que, asociando un EBID falso a un usuario, sea capaz de rastrearlo. El servidor nunca identificaría dicha anomalía al carecer de una firma digital o certificado que corrobore la autenticidad del EBID.
Esto desanonimiza a los usuarios sin necesidad de atacar al servidor al asignarles un EBID persistente externo, permitiendo su geolocalización constante mediantes sensores Bluetooth.

Los detalles de los contactos de un caso positivo son revisados de manera manual por la entidad sanitaria con el fin de evitar falsos positivos, haciendo el trabajo más lento que estando gestionado por un servidor como en las variantes de DP-3T. [56]

Algunos riesgos principales a raíz de estas características son:

- **Falsificación de un riesgo.**
  Dado que los usuarios infectados suben al servidor su lista de contactos, es posible realizar una inyección de un EBID en dicha lista para generar un falso positivo. Dado que la verificación de los encuentros no es posible, no hay manera de defenderse de dicho ataque.
  Este problema no se encuentra en protocolos descentralizados, pues los identificadores del registro de contactos no se suben al servidor en ningún momento. Además es necesario el consentimiento de la autoridad sanitaria para notificar de un positivo.
- **Riesgo de compromiso de datos en dispositivos desbloqueados.**
  Actualmente, el desarrollo de este protocolo es imposibilitado debido a que el día 10 de abril Apple y Google, acorde a la minimalización de datos, introdujeron una nueva api de Rastreo de Contactos, donde se imposibilita la transmisión de la lista de contactos vía red como exige el protocolo PEPP-PT/PEPP.

4.1.4. BlueTrace

El principal inconveniente de este protocolo, al igual que en PEPP-PT/PEPP es el uso del procesamiento de reportes centralizado.

Los protocolos que utilizan este tipo de procesamiento tienen como principal inconveniente el envío de los datos de contacto del usuario a las autoridades sanitarias.

Entre las responsabilidades de dichas autoridades se encuentran asignar los detalles del contacto a cada usuario, determinar si ha habido un contagio y finalmente advertir a los usuarios si este ha ocurrido. Esto implica una correlación directa entre el usuario y los contactos.

En cambio, los protocolos descentralizados delegan todas estas funciones en la red, aumentando así la eficiencia y la privacidad de los usuarios.

A diferencia de PEPP-PT/PEPP, BlueTrace genera los identificadores temporales (TempIDs) utilizando el identificador del usuario, el instante de tiempo en el que se crea el TempID, el tiempo de expiración del ID, un vector de inicialización (IV) y una clave privada proveniente la autoridad sanitaria.

Los tres primeros parámetros se transmiten encriptados pero el IV lo hace en texto plano. Si a esto le sumamos un ataque a los servidores de las autoridades sanitarias, se podrían llegar a desencriptar los TempID.

Otro inconveniente, en este caso propio de este protocolo, es que es necesario proveer el número de teléfono para poder iniciar las aplicaciones que lo utilicen y que las autoridades comuniquen a ese número de teléfono un posible contacto, lo que vulnera claramente la privacidad del usuario. [43]

4.1.5. Otros

A parte de los protocolos mencionados en la sección anterior, existen otras alternativas menos probadas, pero bastantes prometedoras. Estas son consideradas alternativas factibles al protocolo DP-3T, pues presentan un grado similar de privacidad. Estos protocolos son:

- **ConTra Corona.**

  Es un protocolo que pretende aprovechar las virtudes de los protocolos centralizados, pero delegando cierta funciones principales en otras organizaciones, para minimizar la confianza requerida en el servidor central.

  Para ello, esta solución se basa en el paradigma *Upload What You Observed*, que consiste en enviar al servidor todos los pseudónimos (los identificadores efímeros de este protocolo) que ha recogido en un periodo determinado. Esto marca una diferencia con respecto a DP-3T, que utiliza el paradigma *Upload What You Sent*, es decir enviar los pseudónimos que has usado durante el periodo de tiempo. [9]

- **EpiOne.**

  Se trata de un protocolo híbrido que emplea criptografía de clave pública. Permite identificar cuántos tokens almacenados por un usuario coinciden con los que posee el servidor pero sin la necesidad de que el usuario revele sus tokens.

  Esto se logra con cardinalidad de intersección de conjuntos privados o *PSI-CA* **??**. La PSI-CA de EpiOne permite que haya dos partes, cada una con un conjunto privado de tokens, conozcan el tamaño de la intersección entre sus conjuntos sin revelar más información. Con ello se puede ver si hay alguna coincidencia, pero no cuál. De esta forma se respeta la privacidad y a su vez se puede alertar en caso de contacto con alguien contagiado. [63]

- **Pronto-C2.**
  Este protocolo es totalmente descentralizado. Este utiliza el algoritmo de Diffie-Hellman para establecer la clave privada con la que trabajará. También emplea firmas digitales ciegas para preservar la anonimicidad del emisor a la vez que se autentifica. Este protocolo emplea direcciones e identificadores efímeros para cada usuario, las cuales se envían al servidor en caso de contagio.

  Las comunicaciones con el servidor se realizan vía redes privadas como TOR con el fin de preservar el anonimato de los datos enviados. [8]

Por otro lado los protocolos que menos vulnerabilidades poseen son *Hamagen* y *COVID Safe Paths*. Sin embargo, emplean constante geolocalización y los datos enviados son las rutas de los usuarios contagiados, por lo que a nivel de privacidad deja mucho que desear.

Existen más protocolos descentralizados como son PACT (East-coast), PACT (West-coast), DP-3T Unlinkable, TCN o DESIRE, que es híbrido. [7]

4.2. Elección del protocolo

Basándonos en los argumentos proporcionados por el estudio realizado por Serge Vaudenay [75] hemos decidido el uso de DP-3T para nuestra aplicación. Esto es debido a los siguientes puntos:

- En el caso de los sistemas centralizados, los ataques suelen tener consecuencias más graves, pues el acceso es a un único lugar, poniendo en peligro la privacidad de los usuarios ante usuarios malintencionados. El uso de un protocolo descentralizado nos permite mitigar el riesgo a que los usuarios sean rastreados por terceros.
- Si ponemos el foco en el grafo de contactos entre los usuarios, los sistemas centralizados pueden llegar a revelar parte del grafo a un servidor malintencionado. En cambio, los descentralizados solo pueden revelar a un determinado usuario si ha habido contacto entre únicamente dos usuarios.
- Aunque los protocolos descentralizados pueden permitir ciertos ataques de robo de las identidades de usuarios contagiados, DP-3T busca cubrir algunos de ellos. Una de sus soluciones es ante Ataques de Paparazzi. Esta consiste en enviar el identificador "desmenuzado.[en]viando la clave secreta y la fecha necesarias para generarlo por separado. Sin embargo, es cierto que desde este enfoque sería mejor un protocolo centralizado debido a que otros ataques como *Nerd attack* o el Militia attack no están cubiertos en DP-3T. [76]

Sin embargo, debido a que a grandes rasgos cara a la privacidad el riesgo de manejar un protocolo centralizado es mayor que uno descentralizado, optamos por el uso de DP-3T. Esto es porque consideramos más peligroso el hecho de que se pueda acceder a una parte del grafo de contactos de los usuarios o el rastreo de los mismos, que determinar que un individuo puntual esté contagiado.

4.3. Imprevistos respecto al protocolo elegido y consecuencias

Tras comenzar a investigar sobre la implementación de DP-3T para la aplicación, surge el inconveniente de que, si bien su código es abierto, es necesario cumplimentar una solicitud[1] para tener acceso a la API.

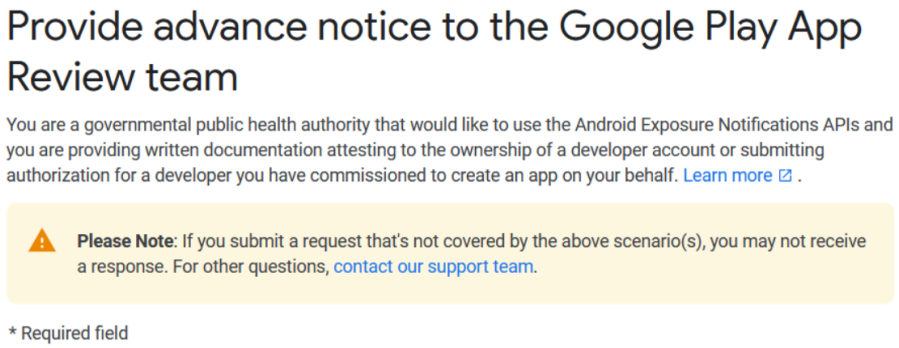


Figura 9: Solicitud a cumplimentar para tener acceso a la API

[1] https://support.google.com/googleplay/android-developer/contact/expo_notif_api

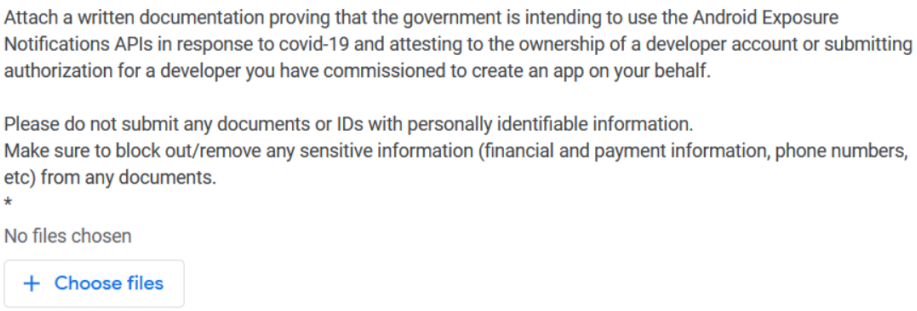


Figura 10: Documentación requerida

Uno de los requisitos exigidos para tener acceso a la API es ser representante o tener el permiso de una Autoridad Sanitaria Pública Gubernamental, así como proveer de una documentación atestando que se está en posesión de una cuenta Google de desarrollador o dando dichos permisos a otra cuenta de la que no se es propietario. Dado que no es nuestra situación, no es posible implementar la aplicación con el protocolo inicialmente elegido.

A causa de este contratiempo, es necesario reorientar el desarrollo a un nuevo protocolo. El hecho de generar nuevos planteamientos entra dentro de la metodología inicialmente elegida para el desarrollo e implementación de la aplicación, en este caso, Extreme Programming. Así pues, aunque se trate de algo imprevisto, es un cambio factible y que no trastocará la planificación inicial.

Como previamente se realizó un estudio de los posibles protocolos candidatos, se procede a considerar cuáles pueden ser otras buenas opciones.

Uno de los requisitos más importantes a tener en cuenta para el desarrollo de la aplicación es el uso de un protocolo basado en el intercambio de claves y no en otros métodos, como la geolocalización o uso de datos personales, que son mucho más invasivas en lo que refiere a la privacidad de los usuarios. Cualquier método que no cumpla los requisitos impuestos por la European Data Protection Board, 2020, será descartado.

Con esto en mente, se vuelven a analizar las alternativas con las que contamos, esta vez considerando en mayor profundidad sus aportaciones y adaptación a la idea definida en los requisitos.

(Google/Apple) Exposure Notification (GAEN) system

Esta opción es descartada de base, pues de hecho es la causa de no poder emplear DP-3T para el desarrollo. Debido a la alta correlación existente entre DP-3T y GAEN, ambos protocolos emplean la misma API de Google. Esto es debido a que DP-3T está implementado empleando GAEN como base. De este modo, el problema no estaría resuelto y la implementación seguiría siendo imposible con el alcance que se posee actualmente.

BlueTrace

A la hora de analizar este protocolo, se encuentra que es de código abierto y su implementación sería posible debido a que todo lo necesario para ello se puede encontrar en GitHub[2].

[2]https://github.com/OpenTrace-community

Si bien inicialmente parece un buen candidato, ya que su estructura basada en el intercambio de claves encaja en la idea plasmada en los requisitos, nos encontramos con dos motivos a considerar en su contra.

- ***Upload What You Observed.*** BlueTrace emplea una metodología centralizada. En concreto, esto se traduce en que el protocolo emplea un paradigma *Upload What You Observed* en lugar de *Upload What You Sent*, que es el usado por DP-3T o GAEN.

  Esto implica que el procesamiento de qué identificadores han estado en contacto con un individuo infectado se lleva a cabo de forma centralizada en el servidor. Por el contrario, con un protocolo descentralizado este procesamiento recaería en los clientes.

  Así pues, el uso de un paradigma *Upload What You Observed* se ajusta peor que uno *Upload What You Sent* al requisito *La confianza en el servidor debe ser limitada*.

  Dado que se trata de un requisito propiamente dictaminado por el European Data Protection Board, se considera importante su cumplimiento.

  Otra razón de peso es que además el uso de protocolos centralizados ha sido fuertemente criticado. Esto es debido a la recopilación de datos de los usuarios que pueden realizar, pues el servidor sabría no solo qué identificadores están contagiados, sino sus contactos y por lo tanto quiénes han quedado expuestos. [62] [37] [23] [27] [36] [12] [41]

- **Identificación de los usuarios.** BlueTrace requiere de un registro por parte del usuario, el cual se lleva a cabo cediendo su número de teléfono. Esta información se emplea únicamente con el fin de contactar a pacientes potencialmente infectados. El hecho de usar esta información personal no es realmente necesario como otros protocolos han demostrado.[11]

  Son diversos los requisitos que no se verían cumplidos de implementar esta opción, entre ellos algunos de los dictaminados por el EDPB en lo que refiere a aplicaciones de rastreo de contactos.

Como puede deducirse de todo lo previamente mencionado, este protocolo puede resultar un tanto invasivo para la privacidad de los usuarios, incumpliendo diversos requisitos dictaminados por el European Data Protection Board. Es por ello que queda descartado.

PEPP-PT/PEPP

Este protocolo, al igual que BlueTrace, posee un planteamiento centralizado, que como ya se ha visto, ha sido ampliamente criticado.

Por otro lado, a diferencia de BlueTrace, PEPP-PT/PEPP no hace uso del número de teléfono del usuario como metodología de registro. En su lugar, el servidor procesa datos personales pseudónimos. Si bien parece una aproximación más respetuosa con la privacidad, la desanonimización de los datos es posible. Dado uno de estos identificadores, es posible etiquetar y clasificar a un usuario de tal modo que terceros puedan reconocerlos sin necesidad de acceder a la base de datos del servidor. Esto es debido a que el *backend* puede reconvertir cualquier identificador en un identificador permanente.

El problema se agranda en el momento en el que el servidor es capaz de relacionar qué pseudónimos han estado en contacto mediante las subidas de identificadores al servidor al reportar un contagio. De este modo, es posible averiguar no solo si un individuo está o no contagiado, sino sus movimientos y con quién ha estado en contacto. [1]

Este planteamiento supone una gran invasión de la privacidad de los usuarios pues, aunque se trata de datos pseudónimos, el hecho de poder realizar una conversión inversa de los mismos expone una gran cantidad de información personal y sensible del usuario, tanto relacionada con su salud como con sus desplazamientos.
A razón de estos hechos, este protocolo ha sido ampliamente juzgado, hasta el punto de que el 20 de abril de 2020 una carta pública fue firmada por 300 académicos de seguridad y privacidad de hasta 26

países diferentes, criticando su funcionamiento y alegando que «solutions which allow reconstructing invasive information about the population should be rejected without further discussion» [Aquellas soluciones que permitan reconstruir información invasiva sobre la población deben rechazarse sin mayor discusión.] (Joint Statement on Contact Tracing, Párrafo 2) [45] [56]

Es por ello que el protocolo queda obviamente descartado.

Protocolos similares a DP-3T en desarrollo

Como se mencionó en el apartado de Estado del Arte: Protocolos, existen tres protocolos que pudieran ser alternativas a DP-3T con un grado similar de privacidad.

ConTra Corona

El primero de ellos es **ConTra Corona**. Este protocolo supone una combinación de metodología centralizada junto a descentralizada. Si bien la gestión de notificaciones de contagios es realizada a la inversa, subiendo cada usuario el listado de identificadores obtenidos en lugar de los propios, el funcionamiento es muy similar.

Inicialmente puede parecer un buen candidato, pero nos encontramos con dos principales inconvenientes.

- **Protocolo recientemente concebido.** Debido a que DP-3T junto a GAEN son los protocolos más implementados a nivel global, esta derivación similar de ellos ha sido todavía poco explorada. Existe documentación que ciertamente habla sobre su implementación en código, pero esta no se encuentra disponible en fuentes abiertas, o bien por materias relacionadas con la propiedad intelectual, o bien porque no haya llegado a implementarse en una aplicación real.

  A causa de dichas carencias, implementar el algoritmo de cero, sin adecuadas referencias o documentación supervisada, extensa o útil, puede suponer un impacto contundente en la planificación, que debido a la limitación de tiempo no es posible.

- ***Upload What You Observed.*** El uso de paradigmas centralizados ha sido abiertamente criticado como se pudo ver en las anteriores opciones, BlueTrace y PEPP-PT/PEPP. Si bien este protocolo no es completamente centralizado, pues delega ciertas funciones a otros sistemas, el hecho de recaer en el paradigma *Upload What You Observed* en vez de en *Upload What You Sent*, puede tener consecuencias negativas en cuanto a privacidad se refiere, como se vio en el apartado de BlueTrace.

  De igual modo que BlueTrace, el requisito *La confianza en el servidor debe ser limitada* queda vagamente cumplimentado, siendo un gran punto en su contra.

Debido a los motivos previamente mencionados, ConTra Corona queda descartado.

EpiOne

El siguiente protocolo a considerar es **EpiOne**. Es un protocolo novedoso que parece solucionar múltiples problemas que los anteriores presentaban. Por un lado presenta un paradigma basado en cardinalidad de intersección de conjuntos privados o *PSI-CA*. Esto permite averiguar si existe una intersección entre el conjunto de identificadores contagiados registrados en el servidor y los identificadores que ha recolectado un usuario concreto. Al informar únicamente sobre la existencia o no de dicha intersección, sin revelar datos sobre qué identificadores son los infectados, da solución al principal problema de DP-3T y GAEN, los cuales buscan coincidencias exactas entre los identificadores contagiados y los almacenados por el usuario.

Por otro lado, los servidores no son centralizados, poniendo solución al problema planteado por BlueTrace, PEPP-PT/PEPP o ConTra Corona.

De este modo, EpiOne parece presentarse como un buen candidato.
Si bien la idea principal es buena, el protocolo todavía carece de una implementación en código, lo que implicaría desarrollarlo desde cero siguiendo la documentación que puede encontrarse sobre ello. [64] [65]

Pronto-C2

Otro protocolo que se ha desarrollado partiendo de la idea de DP-3T es **Pronto-C2**. Este emplea un protocolo descentralizado y utiliza el algoritmo de Diffie-Hellman para establecer la clave privada con la que trabaja, así como firmas digitales ciegas para autentificar a los usuarios. Además, todas las comunicaciones con el servidor se realizan mediante redes privadas, lo cual dificulta la interceptación de la información a la vez que anonimiza los datos enviados.

Este acercamiento resulta bastante atractivo pues, al igual que DP-3T, trabaja con intercambios de claves entre usuarios, las cuales se suben al servidor según el paradigma *Upload What You Sent*. Sin embargo, al igual que ocurre con EpiOne, todavía no se ha realizado una implementación de este protocolo. El agravante es que, además, la documentación referente a Pronto-C2 es mucho más escasa, incluso, que la que se puede encontrar de EpiOne.

De realizar este acercamiento, este debería ser desarrollado de cero y con muy escasa documentación. Dada la limitación de tiempo, tratar de realizar este protocolo con tan poca información disponible en fuentes abiertas, queda descartado. [8]

Decisión final

De entre lo anteriormente visto, se opta por la realización e implementación de un protocolo desde cero. Inicialmente se plantea el desarrollo de EpiOne por las facilidades que ofrece en cuanto a la privacidad.

Sin embargo, se produce un contratiempo debido a la escasez de documentación que se puede encontrar sobre este protocolo, así como a la inexistencia de un código que lo implemente y se pueda tomar como referencia.

Por lo tanto, se opta por la implementación de lo dictaminado por el protocolo inicialmente elegido, DP-3T, del cual hay mucha más información.

La diferencia con respecto al plan inicial reside en que, en lugar de implementar el protocolo mediante el código abierto que puede encontrarse en GitHub[3], el cual recae en la API inaccesible, **se llevará a cabo una implementación desde cero del protocolo, siguiendo los algoritmos, pasos y paradigmas descritos por DP-3T.**

[3] https://github.com/DP-3T/

4.4. Implementación de los requisitos acorde a nuestra versión de DP-3T

Una vez se ha decidido el desarrollo de una versión propia del protocolo DP-3T, es necesario comprobar que este sea capaz de cumplir los requisitos estipulados en la fase de análisis y explicar cómo se llevará a cabo.

Implementación de los Requisitos No Funcionales

- **Rotación de identificadores.** Con el fin de incrementar la confusión y difusión, diariamente se generan un número fijo de identificadores que rotan cada cierto tiempo. Estos identifican de manera anónima a los usuarios.

Implementación de los Requisitos Funcionales

De cara al usuario

- **Visualización de mi riesgo de exposición.** Se mostrará un panel que variará de color según el nivel de riesgo.
  - **Nulo.** Verde.
  - **Riesgo.** Amarillo.
  - **Contagiado.** Rojo.
- **Posibilidad de comunicación de contagio.** Cuando un usuario haya dado positivo, la autoridad sanitaria le proporcionará un código que este podrá introducir en la aplicación. Dicho código se enviará al servidor junto a las semillas generadoras de los identificadores del usuario y la fecha de PCR positiva o de inicio de síntomas. El servidor se encargará de corroborar que dicho código es válido y, de serlo, almacenar los datos enviados.
- **Notificación y muestra de la exposición a un contagio.** Se avisará al usuario de un posible contagio mediante una notificación en su *smartphone*. También dentro de la aplicación mediante el cambio de color del panel de riesgo de exposición.

De cara a otros dispositivos

- **Intercambio de identificadores.** Estos se generarán criptográficamente como especifica DP-3T e identificarán usuarios de manera anónima. El intercambio se llevará a cabo al transcurrir 15 minutos seguidos de contacto con otro dispositivo.
- **Permitir la identificación de otros dispositivos con la aplicación.** La aplicación detectará otros dispositivos cercanos que posean la aplicación activa también. El radio abarcado es de dos metros.
- **Medición del tiempo de exposición a un contacto.** Para medir el tiempo de exposición DP-3T utiliza la atenuación de los paquetes de datos transmitidos mediante Bluetooth, de forma que si dicha atenuación se encuentra por encima de unos valores, se comenzará a contar el tiempo. La estimación se realiza utilizando un conjunto de beacons recibidos de un dispositivo concreto, los cuales se envían cada cierto tiempo (entre 2 minutos y medio y 5 minutos) a modo de cerciorarse de que los dispositivos permanecen en rango del otro.

- **Informar de un contagio.** De manera periódica el servidor emitirá mediante *broadcast* las semillas generadoras asociadas a identificadores contagiados que tiene almacenados. Cuando la aplicación recibe dichas semillas, genera los identificadores en local y los compara con los almacenados. En caso de producirse una coincidencia, la aplicación informa al usuario que ha estado en las cercanías de un individuo contagiado.
- **Contacto tras exposición a otro dispositivo.** A partir de los 15 minutos de exposición se considera contacto.
- **Cálculo de la distancia entre dispositivos.** Se considera distancia cercana cuando se detecta una atenuación inferior a 50dB. Con ello tenemos una muy alta certeza de que la distancia es inferior a dos metros. Para corregir discrepancias entre modelos de dispositivos móviles, al comienzo del encuentro se realiza una calibración. [2]

  Al igual que en DP-3T, existirán los siguientes estados:[3]

  - *Sin riesgo.* El usuario no ha estado en contacto con ningún usuario contagiado.
  - *Con riesgo.* La aplicación ha detectado un identificador contagiado entre sus almacenados, lo cual indica que el usuario ha estado en contacto cercano con algún usuario contagiado.
  - *Contagiado.* El usuario ha proporcionado un código de contagio válido al servidor, lo cual indica que ha obtenido positivo en una PCR, pues una autoridad sanitaria le ha cedido un código válido.
- **Envío de códigos de contagio al servidor de la autoridad sanitaria.** Para realizar el envío del código de contagio DP-3T utiliza un objeto de tipo GaenRequest. Este objeto es una adaptación de la petición básica del protocolo HTTP realizada por el protocolo GAEN. [4]
  Debido a la imposibilidad para acceder a un servidor propio de la autoridad sanitaria, la aplicación se orientará inicialmente a una red local. Para ello lo que se hará es enviar el paquete cifrado por la red a un puerto TCP concreto donde se llevarán a cabo las pruebas.
- **Comunicación con el servidor para obtener lista de nuevos contagios.** El servidor emitirá de manera periódica el listado de todas las semillas generadoras de identificadores contagiados que posea. La aplicación cliente recibirá dicho listado y comparará lo recibido con lo almacenado para determinar si el usuario ha estado expuesto.

De cara a la propia aplicación

- **Cálculo de los identificadores.** Se calculan empleando el algoritmo de DP-3T explicado en la sección dedicada al protocolo DP-3T.
- **Cálculo de estado de exposición.** El estado de exposición pasa a ser de riesgo en el momento en el que se reciben las semillas generadoras de un identificador recibido previamente, es decir, se ha estado en contacto con un usuario contagiado a menos de 2 metros. El estado de contagiado aparece en el momento en el que se envía un código de contagio al servidor y este es validado como tal. [3]

Implementación de los Requisitos de Información

Información almacenada en local

- **Identificadores anónimos propios de cada usuario.** Se usarán identificadores generados de manera pseudoaleatoria, como se ha explicado en el apartado dedicado DP-3T, con el fin de no proporcionar ningún dato personal del usuario.[24]

- **Información anonimizada sobre qué usuarios han estado en contacto.** Se usarán identificadores efímeros que se intercambiarán entre usuarios que mantengan contacto.

Información almacenada en el servidor

- **Datos asociados a los identificadores infectados.** Se almacenarán las semillas generadoras de los identificadores, es decir la clave primigenia y fecha generadora, y la fecha de recepción.
- **Fecha de recepción.** Se almacenará la fecha de recepción de los datos anteriores con el fin de eliminarlos a los 14 días (tiempo que permanece activo el virus).

Base de Datos. Modelo Físico

La arquitectura de este proyecto requiere la utilización de dos bases de datos, una situada en el servidor y encargada de gestionar los identificadores contagiados, y otra en los clientes de la aplicación, en local, encargada de almacenar tanto los identificadores propios como los obtenidos por intercambio.

Para su implementación es necesario llevar a cabo un análisis sobre qué gestores de bases de datos nos ofrecen las características óptimas para cada una de ellas.

Base de Datos Local

La base de datos local es aquella que se creará en las instancias cliente de la aplicación, en este caso los dispositivos Android de cada uno de los usuarios. En este caso, las alternativas que se han encontrado son:[52]

- **Oracle Berkeley DB.** Es un familia de productos que ofrece librerías para gestionar datos con un gran rendimiento y escalabilidad. Proporciona flexibilidad ya que se puede manejar o bien como una base de datos clave-valor o bien como una base de datos relacional cuando sea necesario. Su almacenamiento requiere de en torno a 1 MB como mínimo.

  A pesar de aparentar ser una buena opción, dado que la base de datos local es muy sencilla, y no haremos uso de su escalabilidad y necesidad de alto rendimiento, es preferible buscar opciones que requieran de menor almacenamiento.

- **Interbase ToGo.** Se trata de un sistema gestor de bases de datos relacionales que requiere de mínimo 400 KB de almacenamiento. Es una base de datos SQL empotrada disponible para Android e iOS. Posee módulos propios que permiten integrar opciones offline a la aplicación, y también, eliminar la necesidad de implementar drivers de cliente que se conecten a la versión servidor de esta base de datos.

  Aunque es mucho menos pesada que Oracle Berkeley DB, su licencia es privada y no de dominio público.

- **SQLite.** Se define como un gestor de bases de datos relacional. La principal característica de este gestor es que no consta de una arquitectura cliente-servidor. En su lugar, se enlaza con el programa llegando a formar parte del mismo, ya que toda su funcionalidad esta contenida en una biblioteca de código relativamente pequeña.

  La principal ventaja de este gestor es su tamaño, ya que su tamaño mínimo es de tan solo 500 KB en memoria, pues se almacena como un fichero que la biblioteca bloquea o desbloquea automáticamente en el momento que sea necesario. Además su licencia es de dominio público y existe una amplia documentación sobre su uso en dispositivos Android, lo que facilita su implementación.

La decisión que se ha considerado más apropiada es utilizar el gestor SQLite, debido al poco espacio de almacenamiento que ocupa, las facilidades que ofrece a la hora de implementarse y tener licencia de dominio público.

La primera tabla almacenará aquella información relacionada con los identificadores propios. Además, el valor de *id* sirve para facilitar el acceso a los datos de la tabla.

Tabla ids_propios

- **id** Es el identificador de cada fila de la base de datos. Es de tipo ***INTEGER*** y se autogenera cada vez que se añade un registro. Simplifica el acceso ordenado a la base de datos.
- **identificador_ef** Se trata del identificador efímero empleado para referir a cada usuario de manera unívoca y anónima. Es de tipo ***TEXT***.
- **clave_gen** Es la clave generadora, miembro del par que conforma la semilla generadora del identificador efímero. Es de tipo ***TEXT***.
- **fecha_gen** Es la fecha generadora, miembro del par que conforma la semilla generadora del identificador efímero. Es de tipo ***TEXT***.

La segunda, refiere a aquellos identificadores efímeros obtenidos mediante intercambio BlueTooth.

Tabla ids_ajenos

- **id** Es el identificador de cada fila de la base de datos. Es de tipo ***INTEGER*** y se autogenera cada vez que se añade un registro. Simplifica el acceso ordenado a la base de datos.
- **identificador_ef** Se trata del identificador efímero empleado para referir a cada usuario de manera unívoca y anónima. Es de tipo ***TEXT***.
- **fecha_rec** Es la fecha de recepción del identificador efímero. Es de tipo ***TEXT***. Se emplea para saber cuándo un identificador recibido por intercambio BlueTooth deja de ser contagioso y se elimina (a los 14 días).

Figura 11: Modelo de datos de la base de datos de los clientes

Base de Datos del Servidor

La base de datos del servidor es aquella que almacenará los identificadores de los usuarios infectados y a la cual tendrán acceso las autoridades sanitarias pertinentes.

Para su implementación se han analizado los principales gestores de bases de datos. Se han considerado como candidatos aquellos cuyas características mejor se adaptaban a la aplicación y a su desarrollo futuro. [51] [54] [22]

- **MySQL.** Es un sistema gestor de base de datos relacional de código abierto, considerado el más popular por una amplia mayoría de usuarios. Se utiliza principalmente para el desarrollo de páginas web, aunque su uso en software libre también está muy extendido.

  En el caso de nuestra aplicación, los factores que nos han llevado a considerarlo un buen candidato son principalmente:

  - **Facilidad de uso.** Al ser uno de los gestores más populares, la documentación, los usuarios y los ejemplos de uso son abundantes.
  - **Buen rendimiento.** MySQL destaca por tener un buen rendimiento para bases de datos con una cantidad de datos no muy elevada.

  Precisamente este último punto ha sido determinante y nos ha llevado a descartarlo, pues el objetivo de la aplicación es llegar al mayor público posible y por tanto la cantidad de datos sería elevada. [53]

- **MariaDB.** Este gestor de bases de datos es una derivación de MySQL, por lo que son completamente compatibles. Posee una gran escalabilidad y ofrece buena seguridad y velocidad a la hora de realizar transacciones. Además, es de código abierto, por lo que su licencia es de dominio público.

  Frente a MySQL, su optimizador funciona mejor ante cargas complejas, poseyendo un mejor rendimiento. También ofrece una mayor usabilidad, pues aporta estadísticas de tablas, mejoras en comandos y mayor precisión en algunos tipos de datos, así como facilidades a la hora de realizar testeos. [50]

- **PostgreSQL.** Este gestor está optimizado para gestionar grandes volúmenes de datos, por lo que puede funcionar algo peor con cantidades de datos menores.

  Posee una buena flexibilidad en cuanto a lenguajes de programación y es multiplataforma, por lo que puede adaptarse a múltiples proyectos. Además, dispone de una herramienta mucho más visual, pgAdmin, para gestionar las bases de datos.

  Se caracteriza por ser robusta, eficiente y estable.

  Sin embargo, optimizar su uso y recursos requiere de un mayor conocimiento del gestor. Además, dado que para los casos de prueba se emplearán volúmenes de datos menores, no funcionará de una manera tan optimizada como haría con grandes cantidades de datos.

Se elige, por tanto, MariaDB por las facilidades que ofrece tanto a nivel de testeo, como de documentación al tratarse de un gestor de código abierto.

Tabla ids_infectados

- **id** Es el identificador de cada fila de la base de datos. Es de tipo ***SERIAL*** y se autogenera cada vez que se añade un registro. Simplifica el acceso ordenado a la base de datos.
- **clave_gen** Es la clave generadora, miembro del par que conforma la semilla generadora del identificador efímero. Es de tipo ***VARCHAR***.
- **fecha_gen** Es la fecha generadora, miembro del par que conforma la semilla generadora del identificador efímero. Es de tipo ***DATE***.

- **fecha_rec** Es la fecha de recepción del par clave y fecha generadoras. Es de tipo ***DATE***. Se emplea para saber cuándo un par clave-fecha generadoras deja de ser contagioso y se elimina (a los 14 días).

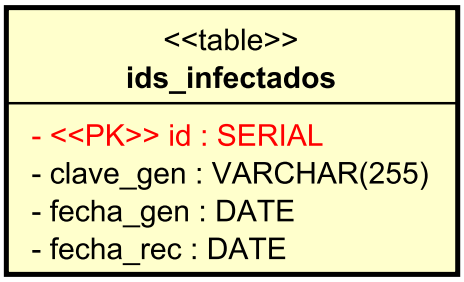


Figura 12: Modelo de datos de la base de datos del servidor

4.4.1. Tecnologías utilizadas

Las tecnologías, y sus correspondientes versiones, empleadas en este proyecto, son las siguientes:

- **Java.** JDK 1.8
- **Android.** Versión 10
- **MariaDB.** Versión 10.5.9
- **SQLite.** Versión 3.28

4.5. Implementación de los Requisitos de Usabilidad

Una vez han sido definidos, se debe pensar la manera de implementarlos en la aplicación a desarrollar.

Veamos, entonces, la propuesta que se ha diseñado para cada aspecto de usabilidad y accesibilidad destacado en la sección de *Análisis*:

Esto, por tanto, obliga a que la interfaz de usuario tenga que estar pensada y diseñada a conciencia.

Los requisitos que debe cumplir el diseño de la interfaz son los siguientes:

- **La interfaz debe ser sencilla.** La funcionalidad de la aplicación se condensará en tres pantallas, principal, información y ajustes de idiomas, sin una navegación entre menús excesiva.
- **La interfaz debe ser suficientemente intuitiva.** Se usará simbología fácilmente identificable y utilizada universalmente en la mayoría de aplicaciones del mercado.
- **La interfaz debe ser agradable a la vista.** Para ello se utilizarán colores suaves. Se buscará que sean gamas de colores afines, evitando contrastes fuertes o visualmente agresivos.
- **Accesibilidad para personas con daltonismo.** Se utilizará un simulador de daltonismo para ajustar los colores de forma que estos sean lo suficientemente diferenciables para personas con trastornos visuales tales como protanopia, deuteranopia y tritanopia.
- **El texto debe ser claro y conciso.** Se evitará el uso de tecnicismos así como de palabras redundantes con el fin de hacerlo más fácil de entender a un mayor número de personas.
- **Legibilidad para todas las edades y capacidades visuales.** Se emplearán tipografías claras y sencillas, así como tamaños de letra lo suficientemente grandes. En el caso de que esto no sea posible (por el tamaño del botón o espacio en la pantalla), se emplearán símbolos visuales para complementar el concepto referenciado.
- **La interfaz debe invitar a publicitar su uso.** Se dibujará a Aga y Gava, que actuarán como mascotas de la aplicación, para hacerla más distinguible.
- **La interfaz debe concienciar sobre los síntomas del estado de exposición del usuario.** Dependiendo del estado, sin riesgo, con riesgo o infectado; Aga y Gava, así como los colores del botón de recomendaciones, aparecerán de un modo u otro.
  - **Sin riesgo.** Aga y Gava aparecerán felices.
  - **Con riesgo.** Aga y Gava aparecerán tomando precauciones y en cuarentena.
  - **Contagiado.** Aga y Gava aparecerán con un termómetro y en una camita siendo cuidados por el enfermero Donehre, otro personaje.
- **Internacionalización.** Se incluirá un apartado de ajustes de idioma para facilitar la accesibilidad a personas con distintas lenguas.

4.6. Diseño de la Interfaz e Implementación de los Requisitos

Esta aplicación está dirigida a un espectro de usuarios muy amplio. Esto impone que la interfaz de usuario sea muy intuitiva y sencilla, con el fin de que un usuario sin experiencia pueda hacer uso de ella con facilidad.

El elemento a destacar en la interfaz de la aplicación es el menú de navegación.

Figura 13: Menú de navegación

Consta de tres botones con amplia superficie para garantizar la máxima precisión a la hora de seleccionar cada uno. Además, se ha cuidado de que únicamente se implementen las funcionalidades necesarias, dejando de lado elementos superfluos. En orden de izquierda a derecha son:

- **Acceso a pantalla principal.** Nos permite ingresar en la pantalla principal de la aplicación cuando nos encontremos en cualquier otra pantalla, salvo en el formulario de comunicación de contagio.
- **Acceso a pantalla de información.** Nos permite ingresar en la pantalla de información cuando nos encontremos en cualquier otra pantalla, salvo en el formulario de comunicación de contagio.
- **Acceso a pantalla de cambio de idioma.** Nos permite ingresar en la pantalla de cambio de idioma cuando nos encontremos en cualquier otra pantalla, salvo en el formulario de comunicación de contagio.

A continuación procederemos a explicar brevemente cada diseño preliminar de las respectivas pantallas. En las imágenes se señala con una flecha en cuál de los botones del menú inferior se encuentra.

4.6.1. Pantalla Principal

La pantalla principal cuenta con un título además de las siguientes áreas:

Figura 14: Pantalla Principal

- **Pantalla Riesgo de Contagio.** Aquí se indicará el nivel de riesgo de contagio del usuario. Dependiendo del nivel de riesgo el color de este recuadro cambiará:
  - **Verde.** El usuario no ha estado en contacto con ningún individuo contagiado, ende el riesgo de contagio **no existe**.
  - **Naranja.** El usuario ha estado cerca de algún individuo contagiado, ende posee **riesgo de contagio**.
  - **Rojo.** El usuario ha enviado un código proporcionado por una entidad sanitaria y el servidor lo ha validado, ende estando **contagiado**.

  Como hemos detallado en los requisitos de usabilidad de la aplicación los colores elegidos son suaves y poco impactantes.
- **Botón Comunica tu contagio.** Si el usuario desea comunicar su contagio, ha de pulsar este botón. Cuando lo haga aparecerá una pantalla como la siguiente:

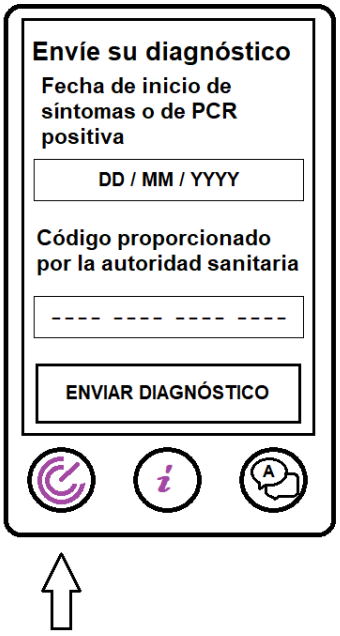


Figura 15: Botón Comunica tu contagio

Esta pantalla incluye dos cuadros, el primero deja introducir la fecha de inicio de síntomas o PCR positiva, el segundo, el código proporcionado por la autoridad sanitaria, el cual seguirá un patrón con el fin de evitar envío de diagnósticos falsos.

Una vez pulsado el botón de *Enviar diagnóstico*, la aplicación procede a mostrar el siguiente cuadro confirmativo.

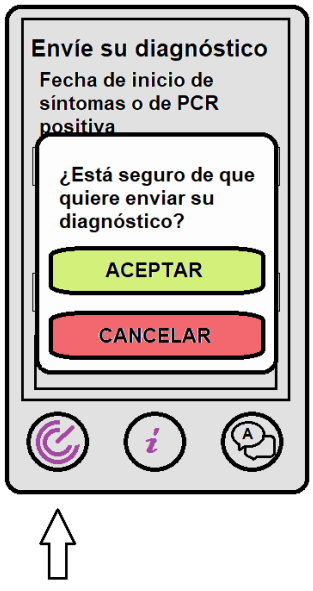


Figura 16: Confirmación del envío

Los colores elegidos para los botones de *Aceptar* y *Cancelar* son verde y rojo suaves para minimizar el impacto visual. También, debido a que son bastante intuitivos y que para los tres tipos de daltonismo más habituales, protanopia, deuteranopia y tritanopia, se mantiene una diferenciación suficiente entre los colores.[78]

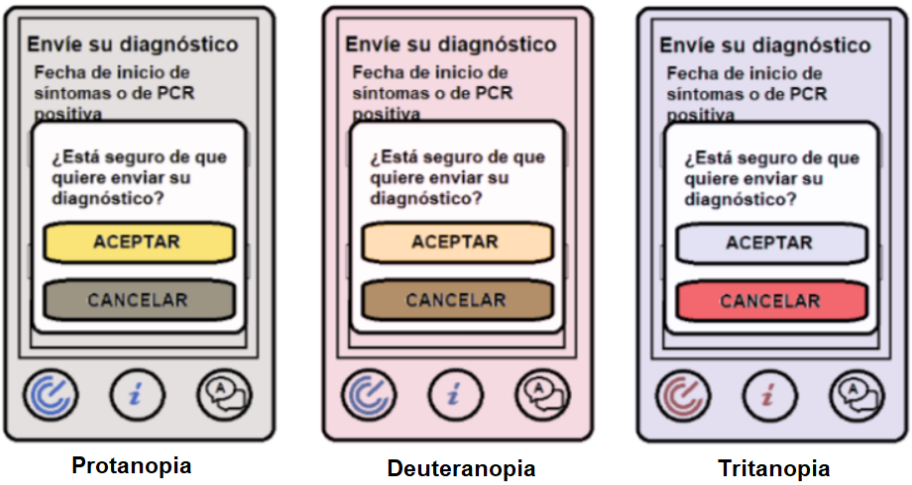


Figura 17: Simulación de los distintos tipos de daltonismo

4.6.2. Pantalla de Información

La pantalla de información contiene una pequeña presentación de la aplicación, así como donde se ubicaría la política de privacidad, para que esta pueda ser consultada en cualquier momento.

Figura 18: Pantalla de información

4.6.3. Pantalla de Ajustes de Idioma

En la pantalla de ajustes de idioma aparecen una serie de botones de selección del idioma en el que se quiere poner la aplicación. Tras seleccionarlo hay que pulsar el botón de aceptar, con el fin de confirmar la acción para evitar que el usuario cometa un error.

Figura 19: Pantalla de idiomas

4.7. Diagrama de paquetes

Se procede a mostrar la estructura de paquetes que poseerá la aplicación, así como las dependencias entre ellos.

Debido a la estructura Android de la aplicación cliente, esta consta de dos partes. Una, el paquete *java*, el cual alberga aquellos paquetes que contienen las clases de la aplicación.

Dentro se encuentran las *Activity*, es decir, las pantallas de la aplicación; así como paquetes que gestionan recursos que estas emplean, como tipos de comunicaciones con el exterior, cifrado o base de datos.

Por otro lado, se almacenan los paquetes con aquellos recursos útiles para las clases anteriores, como imágenes o *strings* de idiomas. Todo ello se encuentra en el paquete de *res*.

La aplicación servidor, se ha diseñado estructurando los paquetes por comunicaciones con el exterior y manera de gestionarlo, así como acceso a base de datos y cifrado.

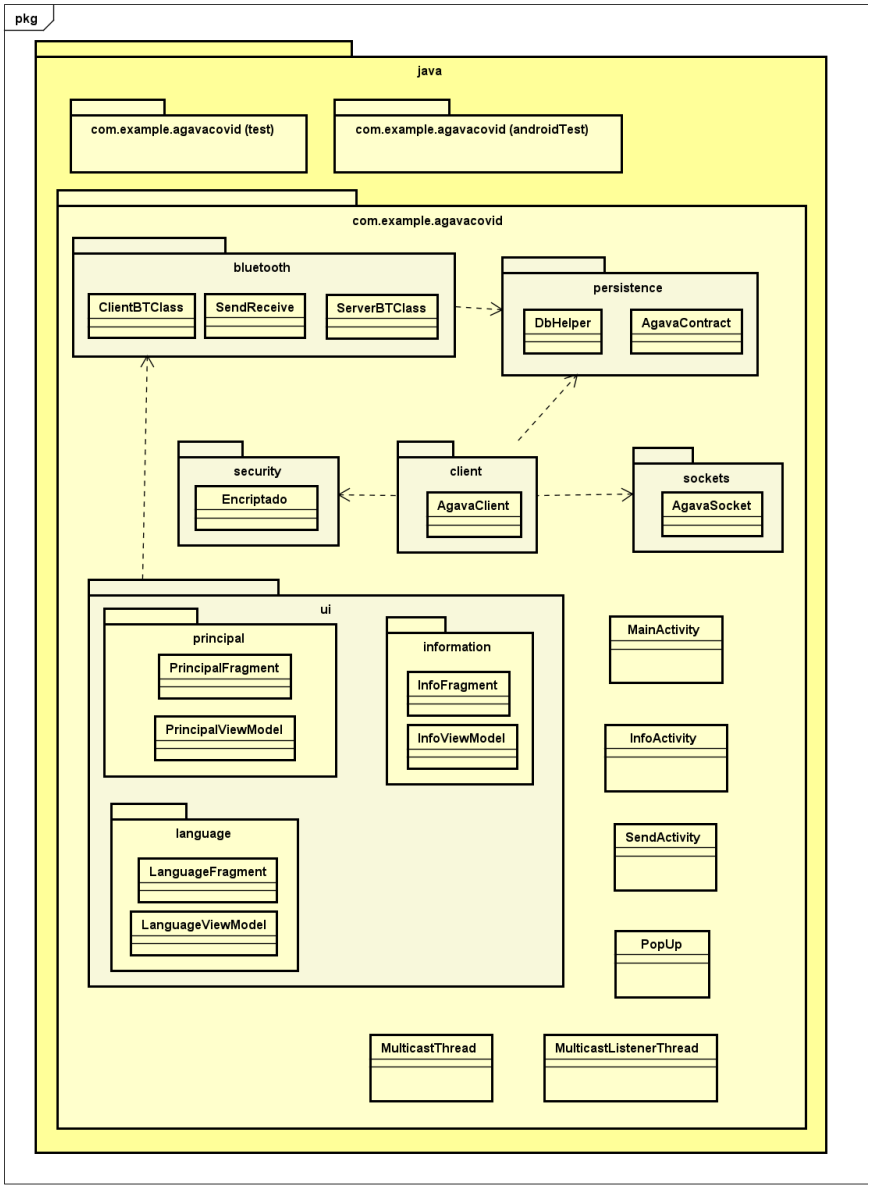


Figura 20: Diagrama de Paquetes Aplicación Cliente JAVA

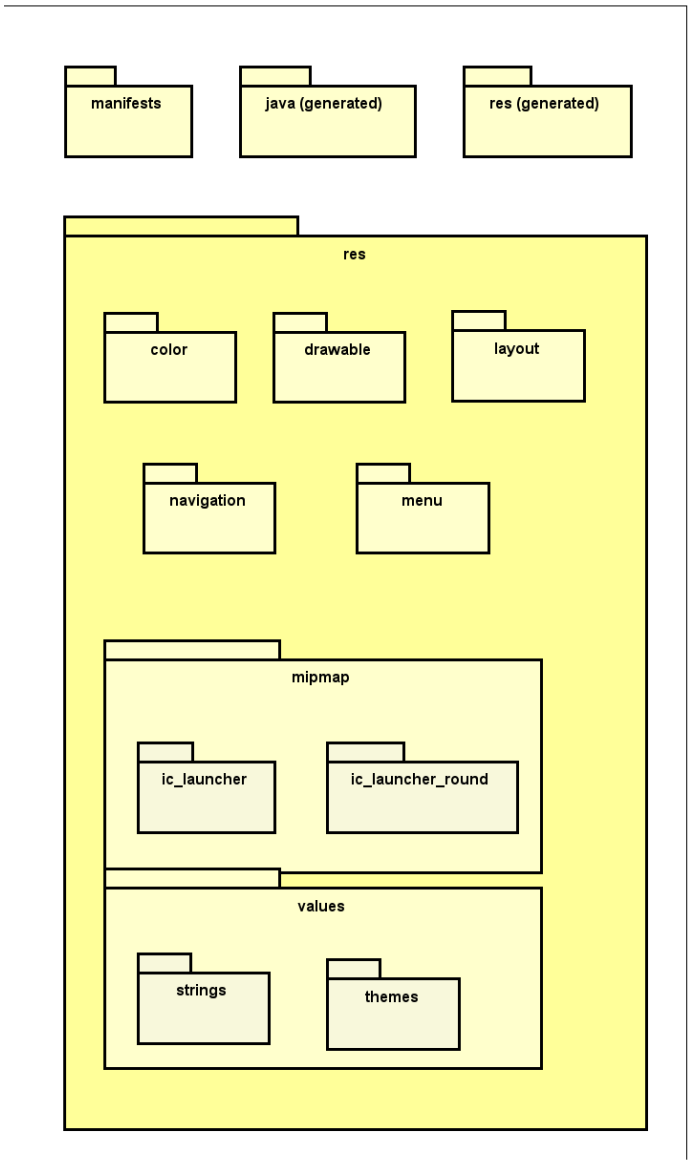


Figura 21: Diagrama de Paquetes Aplicación Cliente RES

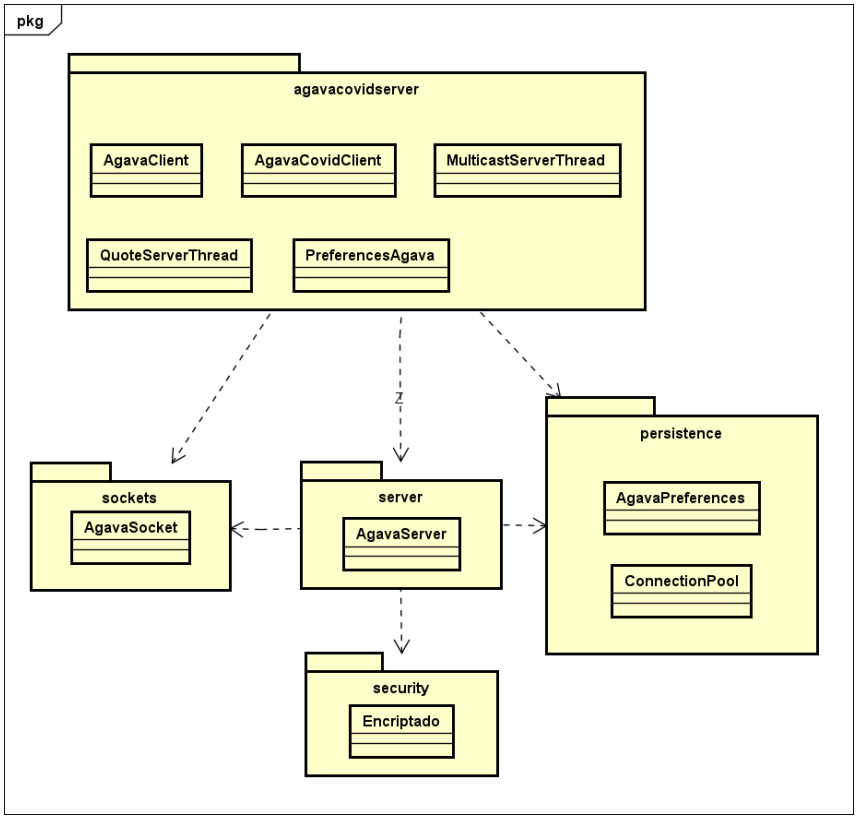


Figura 22: Diagrama de Paquetes Aplicación Servidor

¿Quiénes son Aga y Gava?

Aga y Gava son dos dragoncitos, pero los nombres provienen de *agaporni* y *gaviota*, respectivamente.

Aga fue creada a inicios de la carrera por María Ruiz Molina y ha sido su marca en diversos trabajos de asignaturas, así como proyectos individuales.

Gava fue creado por Juan Velázquez García como intento de dibujar a Aga. Su nombre proviene de que en su primera versión, la cual fue un intento de dibujar a Aga, su pelo parecía una gaviota. El diseño ha evolucionado perdiendo la forma de pico de gaviota a un pelo más refinado.

Posteriormente Gava se añadió al universo de Aga junto a otros tantos personajes que se crearon, varios de ellos a modo de avatares o *agatares* de amigos del grupo de la facultad.

Si bien la idea final de estos personajes es incorporarlos a futuro como parte de un videojuego o historietas cómicas, debido a su simpleza y diseño lindo se ha decidido incluirles también en este trabajo a modo de mascotas y marca personal.

Como anécdota, el primer trabajo universitario realizado conjuntamente por ambos autores incluyó también a Aga y Gava, en el primer cuatrimestre de segundo de carrera, y desde entonces Aga ha acompañado prácticamente todas las entregas de María Ruiz Molina.

4.8. Implementación de la aplicación

4.8.1. Implementación final de la interfaz

A la hora de implementar la interfaz en la aplicación, se optó por usar colores distintos de los de Radar Covid. Así, se cambió la paleta de colores de morado a azules suaves.

La aplicación queda con la siguiente interfaz, respetando los los requisitos anteriores.

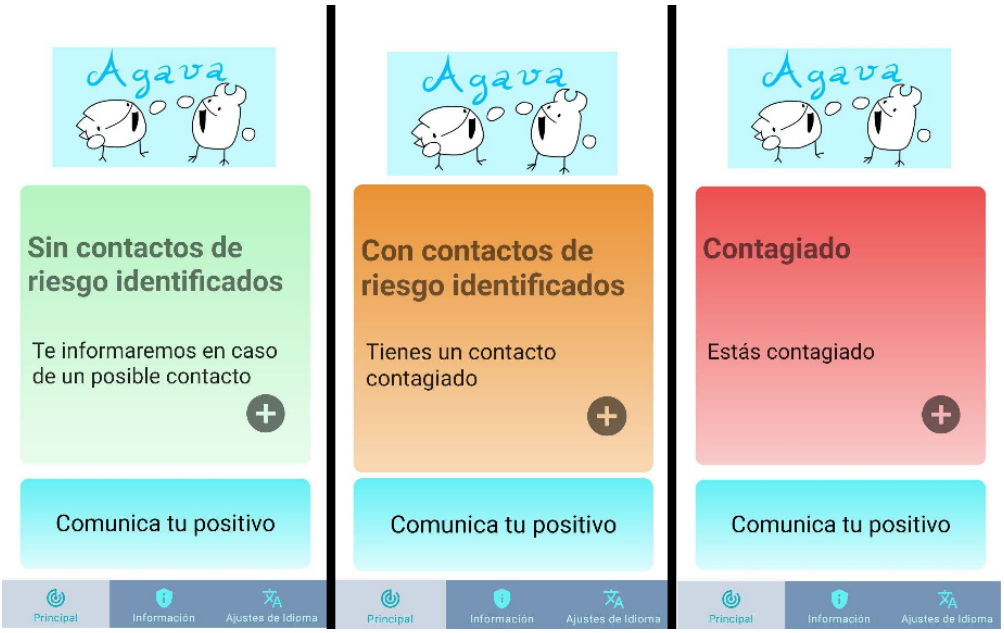


Figura 23: Pantalla de inicio. Colores de la aplicación

Como puede verse, los colores siguen siendo lo suficientemente diferenciados, permitiendo la visualización de los iconos del menú de abajo, así como la legibilidad.

En cuanto a las pruebas de daltonismo, los colores del menú de abajo siguen contrastando lo suficiente. Los colores del nivel de alerta puede que se vean peor en ciertos casos, pero al ir acompañados de un mensaje sobre el estado de contagio, se compensa el menor contraste entre colores asociados a estados.

Estas pruebas se han realizado con el simulador online
https://www.color-blindness.com/coblis-color-blindness-simulator/

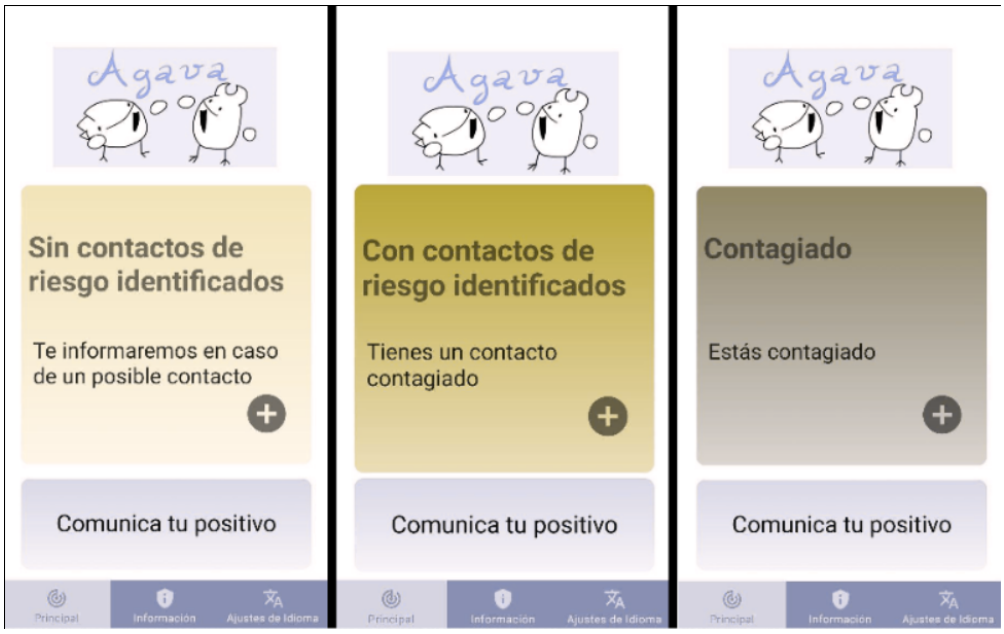


Figura 24: Pantalla de inicio. Protanopia

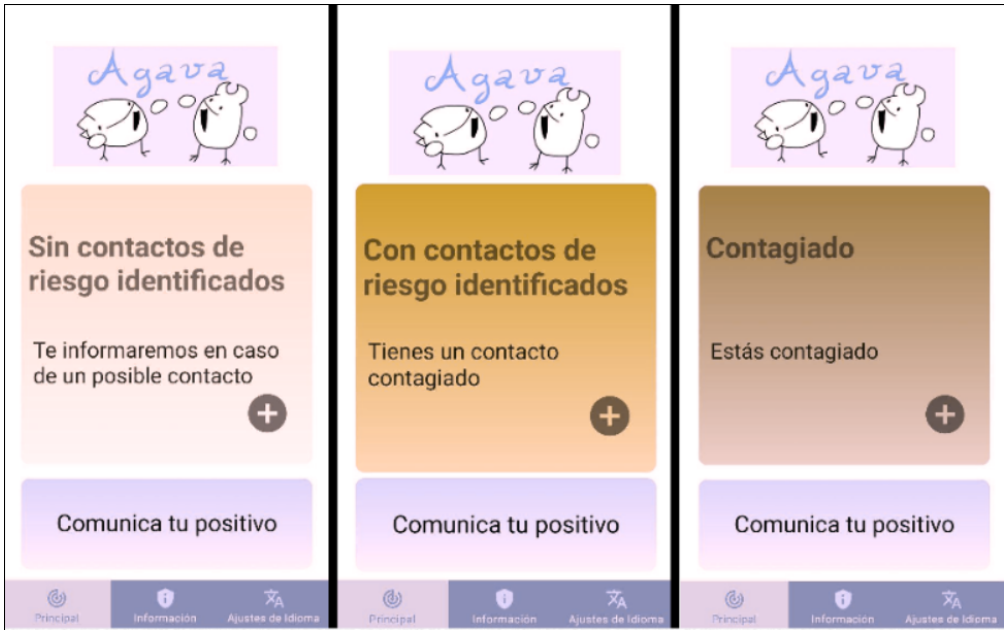


Figura 25: Pantalla de inicio. Deuteranopia

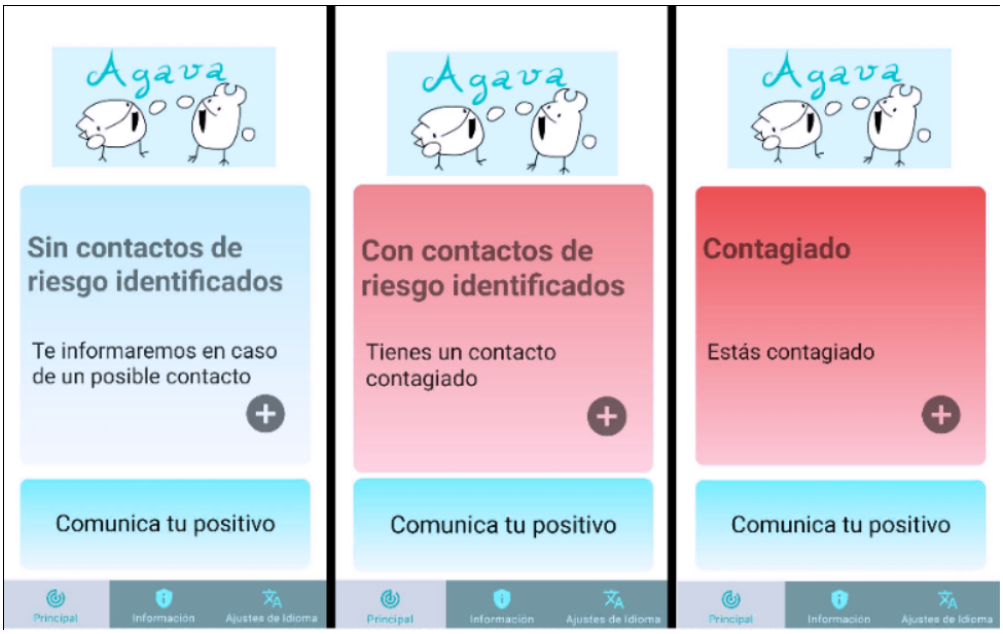


Figura 26: Pantalla de inicio. Tritanopia

4.8.2. Conexiones de red TCP

En la elaboración del protocolo, así como de las conexiones que realizará el cliente con el servidor, los envíos de códigos contagiados se harán por red empleando el protocolo TCP. De este modo, nos aseguramos de que el orden de envío de los paquetes se mantenga, así como evitar la pérdida de estos. Esto es importante, pues el código solo va a transmitirse una vez por red.

4.8.3. Conexiones de red UDP

Por otro lado, el servidor enviará identificadores contagiados de manera periódica. Este envío se realiza a todos los clientes, por lo que se trata de un multicast, donde los clientes pertenecen a un grupo concreto, definido con una dirección IP de grupo de multicast.

Debido al uso de multicast el protocolo empleado es UDP, pues la conexión no se realiza solamente entre dos dispositivos, sino que es de un servidor a todos los clientes.

4.8.4. Decisión sobre Bluetooth

El hecho de desarrollar una versión del protocolo DP-3T desde cero, aunque este utilice Bluetooth Low Energy, plantea un dilema sobre qué tipo de protocolo Bluetooth es más aconsejable.

Durante el estudio realizado con anterioridad sobre protocolos de rastreo de contactos, se encontró que BLE era una tecnología que ofrecía como principal ventaja la eficiencia energética, pero a cambio de necesitar permisos de geolocalización para poder funcionar. Debido a la orientación hacia la privacidad de este proyecto, esta ventaja debía ser lo suficientemente considerable como para justificar su uso.

Además, debido a que BLE funciona por broadcast, enviando la MAC Bluetooth, el UUID e información sobre el servicio abierto, los mensajes y esta información pueden ser interceptados con mayor facilidad.

Bluetooth por otro lado crea canales seguros mediante el pareado de dispositivos. De este modo se realiza un intercambio de clave y el canal se cierra a agentes externos. [**blvsble1**]

Tras muchas búsquedas, se encontraron estudios sobre sus diferencias, aunque los resultados no proporcionaban información sobre la disparidad de gasto energético en cuanto a tiempo de uso de la batería.

Por esta razón, se decidió realizar un estudio propio de forma rápida, para comprobar cuánta batería puede llegar a consumir Bluetooth en un periodo de tiempo amplio.

Para ello, se conectaron unos cascos inalámbricos a un dispositivo móvil y se reprodujeron pistas de audio durante 1 hora. Tras este tiempo, se comprobó el porcentaje aproximado de consumo de batería que ofrece el sistema. El valor obtenido fue un consumo de batería aproximado de un 2 % en 1 hora de reproducción de audio. Si extrapolamos a la cantidad de datos que se van a intercambiar con esta aplicación, por supuesto mucho menor, podemos concluir que la diferencia en cuanto a consumo de energía entre Bluetooth y BLE no justifica el solicitar permisos de geolocalización al usuario.

Es por ello que esta aplicación utilizará el protocolo Bluetooth.

4.8.5. Cifrado de los datos

El cifrado de los datos se llevará a cabo durante la transmisión TCP del cliente al servidor, para de este modo evitar la interceptación o escucha de lo identificadores contagiados y del código de contagio proporcionado por la autoridad sanitaria.

Para ello, el servidor envía al cliente su clave pública. Con ella, el cliente cifraría una clave simétrica AES-256 y se la enviaría al servidor. Una vez que ambos poseen la clave simétrica, el cliente enviaría el paquete con el código e identificadores contagiados. De este modo, solo el servidor podría descifrar el mensaje.
Tras dicho intercambio, se ha optado por un cifrado de clave simétrica de los datos debido a su menor complejidad computacional
Además, el tipo de cifrado usado se ha determinado calculando la entropía generaba tras cifrar un mensaje formateado como un código y los identificadores a enviar.

En la siguiente imagen pueden verse los resultados obtenidos con los tipos de cifrado, en orden, ECB (*Electronic Codebook*), CBC (*Cipher Block Chaining*) y CFB *Cipher Feedback*):

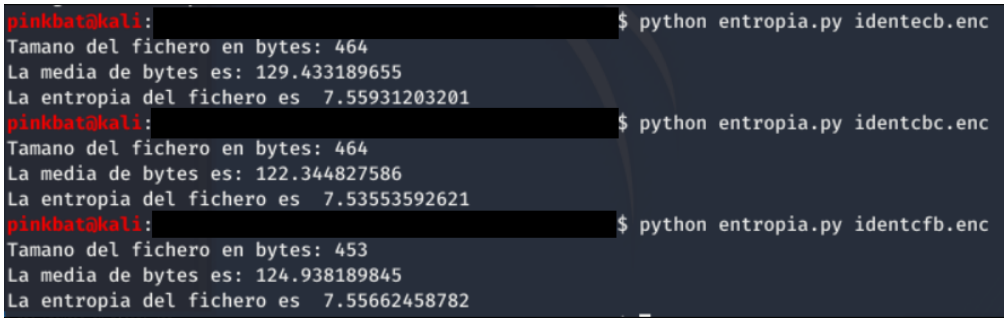


Figura 27: Resultado de entropías

Si bien el tipo de cifrado que mejor entropía obtiene es ECB, debido a su funcionamiento es descartado. Al tratarse de un cifrado que a iguales bloques da iguales resultados, es susceptible de ataques de repetición. Esto se debe a que emplea la misma clave para cifrar cada bloque, y de haber dos iguales, el resultado sería

el mismo.

Es por ello que se escoge CFB, por ser el siguiente con mejor entropía, además de estar orientado a cifrado de textos.

4.8.6. Adaptaciones realizadas cara al prototipo

Debido a la complejidad de la aplicación, así como del protocolo a implementar, y las restricciones de tiempo, se han realizado simplificaciones en algunos aspectos del desarrollo de la aplicación, llevando a cabo un prototipo que implementa todas las funciones pero con algunos cambios.

Estos o bien facilitan el seguimiento de las pruebas y análisis posteriores, o bien facilitaron la programación de algunos aspectos, permitiendo cuadrar los tiempos dentro de la planificación. Las adaptaciones son las siguientes.

- **Intercambio activo de los identificadores.** Para un mayor control de las pruebas, se ha implementado un botón en la aplicación prototipo. Al pulsarlo se realiza el envío de los identificadores vía Bluetooth a los dispositivos pareados.
- **Envío activo de ruido.** Para evitar postergar en la planificación las fases posteriores a la implementación de la aplicación, se ha prescindido de esta funcionalidad en este prototipo.
- **Generación y rotación manual de los identificadores.** Debido a la planificación prevista, no ha sido posible implementar un contador que generase y gestionase los identificadores en segundo plano. Para la realización de las pruebas, en caso de necesitar esta funcionalidad, se simula la rotación empleando diferentes versiones, cada una con un identificador distinto.
- **Cambio manual de estado de *Contagiado* a *Sin contactos* tras 14 días.** Debido al tiempo del que se disponía, se decidió realizar el cambio de estado tras 14 días de *Contagiado* a *Sin contactos* cada vez que se reiniciase la aplicación. De otro modo, debería mantenerse la aplicación activa en todo momento en segundo plano, para que pasados los 14 días cambiase el estado. Otro modo sería realizarlo tan solo al abrir la aplicación, comprobando la fecha, pero esto falsearía realmente el borrado tras 14 días, pudiendo ser más de no abrirse la aplicación transcurrido exactamente ese tiempo.
- **Eliminado manual de los identificadores con fecha extinta.** Debido al tiempo del que se disponía, se decidió realizar el borrado en el servidor de manera manual desde la consola de MariaDB, mediante el comando:

```
DELETE FROM ids_infectados WHERE CURDATE()-fecha_rec > 14;
```

- **Intensidad de la señal de Bluetooth predeterminada.** Debido al tiempo del que se disponía, se decidió dejar la distancia predeterminada, pues para realizar los casos de prueba y el análisis de privacidad, esta no iba a influenciar.
- **Pareado manual previo al intercambio de identificadores vía Bluetooth.** El pareado se realiza desde Ajustes del teléfono. Tras varios intentos sin éxito de programar el pareado para que la aplicación lo realizase automáticamente, se decidió no dedicar más tiempo a esto para evitar afectar a la planificación.

  Ya que el realizar el pareado dentro o fuera de la aplicación no era algo fundamental para poder realizar las pruebas de intercambio de identificadores, se decidió hacer de manera manual.

- **Simplificación del proceso de cifrado de los datos que viajan por red.** La idea original es que, con cada intercambio de información entre cliente y servidor, se realice el siguiente cifrado.
  El servidor poseerá un par clave pública-privada. A su vez, el cliente generará una clave simétrica AES256.
  Cuando el cliente realice una conexión TCP con el servidor, este le enviará al cliente su clave pública, con la que el cliente podrá cifrar la clave simétrica para enviársela al servidor y así comenzar a intercambiar la información cifrada.

  En el prototipo de la aplicación desarrollado, se establecerá manualmente una clave simétrica entre servidor y cliente para realizar las pruebas y análisis. Esta clave será constante y no viajará por la red.

  Al trabajar con CFB, modo de cifrado que necesita de un Vector de Inicialización, este también estará fijado, para asegurar mismos resultados en ambos lados de la aplicación, cliente y servidor. De nuevo, esto es una medida tomada con el fin de simplificar el proceso de cifrado en el prototipo.
- **No retroalimentación sobre si el envío del código es o no correcto.** El servidor no enviará al cliente un mensaje de retroalimentación sobre si el código es o no es correcto. Esta funcionalidad, si bien sería clave cara a una aplicación con una buena usabilidad, se ha prescindido cara al posterior análisis de privacidad.
- **No implementación de la notificación al usuario.** Dado que no es una funcionalidad fundamental para el prototipo, se ha prescindido de que la aplicación avise al usuario mediante una notificación cuando haya un cambio de estado.

5. Casos de prueba

El objetivo de las pruebas software es comprobar que la aplicación cumple con los requisitos que se han dictaminado en la *Fase de Análisis*.

Aunque se pueden clasificar de diversas formas, las principales clases de prueba son pruebas de caja negra y pruebas de caja blanca. Como añadido, se pueden categorizar dependiendo de qué aspecto se está probando.

En este caso, se han clasificado en pruebas de caja negra y caja blanca, y dentro de estas funcional y no funcional. Las pruebas englobarán pruebas de seguridad, de disponibilidad, de interacción y de comunicación Bluetooth, UDP y TCP.

5.1. Pruebas de caja negra

Las pruebas de caja negra son aquellas que se realizan sin saber la especificación de cómo se ha hecho aquello que se prueba, en otras palabras, solo nos interesa ver qué salidas o *outputs* producen las entradas o *inputs* que se introducen en el software.

Las pruebas se consideran correctas cuando se obtiene el resultado esperado, cumpliendo los requisitos previamente definidos. En caso contrario, se detallan los errores obtenidos y cómo se solucionaron hasta obtener el resultado deseado.

5.1.1. Intercambiar un ID entre dos dispositivos vía BT en rango

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones Bluetooth**

Se inician dos dispositivos móviles y se activa Bluetooth en ambos. Después, se parean de forma manual a través del menú de Ajustes de cada dispositivo.

En un principio, este proceso previo se iba a realizar de forma automática en la aplicación. La idea inicial era hacerlo mediante código y sin pareado. Se consiguió la detección de los dispositivos, pero la aplicación se cerraba en cuanto ocurría.

Siguiendo la metodología escogida, tras varios intentos sin éxito se tomó la decisión de realizar este proceso pareando los dispositivos y desvinculándolos al terminar la operación, pues investigando, se encontró que la comunicación mediante dispositivos Bluetooth pareados se realiza mediante un canal cerrado.[10]

Finalmente, al obtener los mismos resultados, se decidió optar por realizar el pareo de forma manual.

Se inicia la aplicación y se presiona el botón que se ha habilitado para realizar las pruebas Bluetooth.

Los primeros resultados fueron negativos. Los dispositivos conectaban pero la aplicación se cerraba al instante.

Esto era debido a un mal uso de las funciones Bluetooth, en concreto *cancelDiscovery()*, pues se situó en un lugar erróneo del código.

Posterior a ello, debido a cambios anteriores, había quedado una conexión insegura de Bluetooth, realizada mediante *listenUsingInsecureRfcommWithServiceRecord* y *createInsecureRfcommSocketToServiceRecord*. Debido a esto se producía una incoherencia, pues ese tipo de conexión permite conectar dispositivos sin previo pareado, cuando la aplicación funciona mediante dispositivos pareados.

Tras realizar la conexión en modo seguro de nuevo, uno de los dispositivos recibió el mensaje con los identificadores de manera correcta, manteniendo la aplicación abierta.

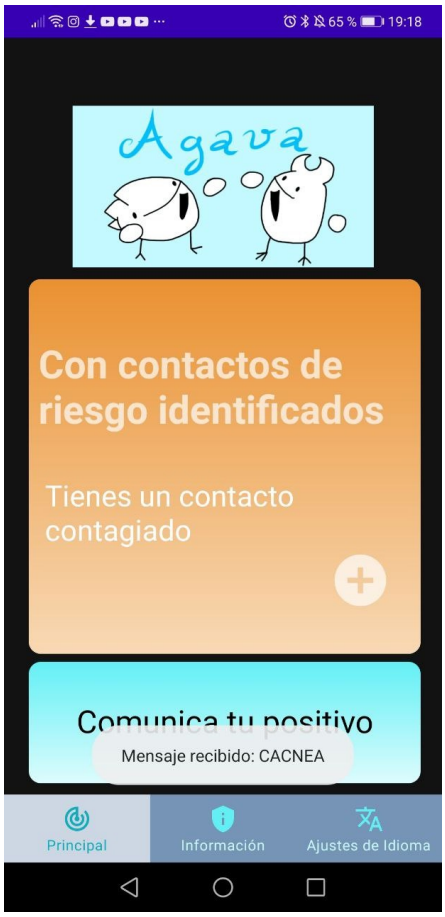


Figura 28: Intercambio de ID entre dispositivos

5.1.2. Intercambiar un ID entre dos dispositivos vía BT en el límite del rango

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones Bluetooth**

Se inician dos dispositivos móviles con Bluetooth activado. Tras ello se realiza el pareado manual a través del menú de Ajustes del teléfono.

A continuación, se buscó el límite del rango que alcanza la señal de Bluetooth.

Inicialmente se calculó mal y se realizó desde una distancia más cercana. Se fue buscando el punto límite hasta que se perdía la señal.

Una vez encontrado el punto límite se realizó el envío de un identificador.

En el dispositivo que actuaba como servidor apareció el mensaje de *Conectado*, pero no llegó el identificador.

Este resultado es lógico, pues la recepción de la conexión ocupa menos slots que el propio identificador. Esto se debe a que el mensaje se divide en múltiples slots que son enviados y por lo tanto la pérdida de uno es más probable, haciendo que el mensaje ya no llegue completo y correctamente.[61]

5.1.3. Intercambiar un ID entre dos dispositivos vía BT fuera de rango

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones Bluetooth**

Se inician dos dispositivos con Bluetooth activado. Estos se parean manualmente desde la sección de Ajustes del sistema.

Tras ello, se inicia la aplicación, y estando los dispositivos lo suficientemente alejados, se realiza un envío.

Como era de esperar, no se recibe ningún tipo de señal.

5.1.4. Intercambiar un ID entre dos dispositivos vía BT y desconectarse justo en el momento del envío

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones Bluetooth**

Se inician dos dispositivos con Bluetooth activado. Se parean desde Ajustes del teléfono de manera manual.

Tras ello, se inicializa la aplicación y se pulsa el botón de envío de identificador.

En el momento en el que en el dispositivo que actúa como servidor aparece el mensaje de *Conectado*, se desactiva Bluetooth del dispositivo que actúa como cliente.

Como era de esperar, la conexión se interrumpe, no llegándose a enviar el mensaje con el identificador, por lo que el servidor no recibe la información.

5.1.5. Intercambiar un ID entre dos dispositivos vía BT y desconectarse justo 1 segundo antes del momento del envío

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones Bluetooth**

Se inician dos dispositivos con Bluetooth activado. Se parean desde Ajustes del sistema de manera manual.

Tras ello, se inicializa la aplicación y se pulsa el botón de envío de identificador.

Inicialmente aparecen los mensajes de creación de los sockets, mensajes dispuestos a modo de comprobar su correcto despliegue en la aplicación prototipo.

Una vez ambos dispositivos han creado el socket correspondiente para comunicarse entre sí, se desconecta Bluetooth.

Debido a ello, como era de esperar, se obtiene el mensaje de *Conexión fallida*, pues no se llega a establecer la conexión entre dispositivos.

5.1.6. Intercambiar un ID entre dos dispositivos vía BT y desconectarse justo 1 segundo después del momento del envío

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones Bluetooth**

Se inician dos dispositivos con Bluetooth activado.

Tras ello se parean de manera manual mediante la opción Ajustes del teléfono.

Inicialmente aparece el mensaje de *Conectado*. Tras ello el de mensaje recibido, con el identificador.

Se desconecta Bluetooth tras ello, y como era de esperar, el identificador se almacena correctamente. Esto es porque el mensaje ya se recibió previamente a la desconexión, en el momento en el que sale un aviso en la pantalla de la aplicación.

5.1.7. Uso de la aplicación sin conexión a BT

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Disponibilidad**

Se abre la aplicación y nada más ejecutarse sale el siguiente aviso en pantalla:

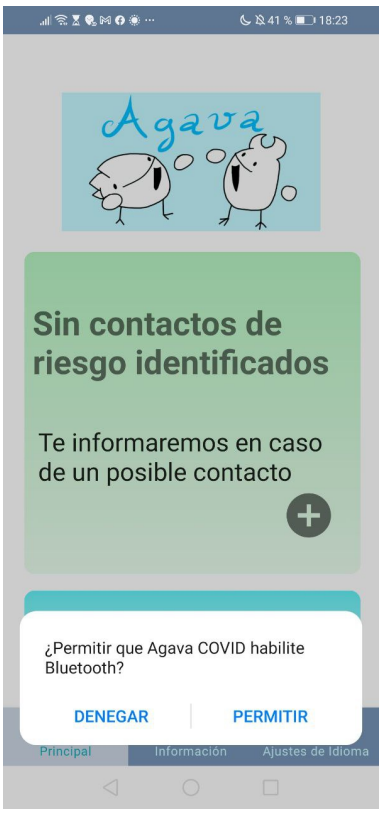


Figura 29: Solicitud para activar Bluetooth

De este modo, la aplicación nos comunica que el dispositivo no tiene activado Bluetooth.

Tras este aviso, pulsamos *Denegar* para que la aplicación siga funcionando sin Bluetooth. Tras ello, la aplicación funciona sin problemas.

Permite las conexiones con el servidor vía Internet, es decir, pueden enviarse códigos de contagio al servidor y recibir el multicast sin problemas.

El funcionamiento es el esperado, pues aquellas funcionalidades relacionadas con Bluetooth dejan de poderse ejecutar al no activarlo.

Por otro lado, aquellas que no necesitan de Bluetooth, funcionan correctamente.

5.1.8. Envío de un código correcto al servidor. (Código correcto: el pedido por la aplicación, otorgado por la autoridad sanitaria.)

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones TCP**

Se abre la aplicación estando el dispositivo conectado a Internet.

Tras ello, dentro de la aplicación se selecciona *Comunica tu positivo* y se rellenan los datos. La fecha solo permite seleccionar aquellas fechas entre la actual y 14 días atrás. El código solo permite introducir doce caracteres numéricos, formato de los códigos de contagio.

Inicialmente, el caso de prueba tenía un error, pues permitía enviar claves y fechas generadoras al servidor sin la necesidad de rellenar previamente los datos de fecha y código. De este modo se permitía el envío de claves y fechas generadoras sin comprobación, pudiendo enviar claves y fechas generadoras de una persona no contagiada como si lo estuviera.

El problema se solucionó obligando a que el envío recogiera la información introducida en esos campos.

De este modo, la información enviada por la red recoge tanto la fecha, como el código, como la lista de claves y fechas generadoras de identificadores, obtenida de la base de datos local.

Inicialmente hubo un problema, pues el servidor no recibía información. Esto es porque se abrían puertos no habituales, y por lo tanto, el firewall del ordenador empleado para las pruebas impedía la llegada de los paquetes generados por el dispositivo cliente, el móvil.

Se desactivó el firewall para realizar la recepción con el fin de finalizar el caso de prueba, y efectivamente se recibió la información en el servidor sin problemas.

Una vez el servidor recibe esta información, comprueba que el código sea uno de los generados por la autoridad sanitaria, y por lo tanto correcto y activo, y de ser así introduce a la base de datos las claves y fechas recibidas. Dicha comprobación la realiza comparándolo con un listado de códigos que tiene almacenado en un *Array*, y en este caso se produce una coincidencia, por lo que permite la inserción.

Una vez recibido, se cambia el valor de la variable que define el estado de contagio. Con ello, la imagen y los textos de la pantalla principal cambian al estado *Contagiado*. Inicialmente, para que este cambio fuese visible en la interfaz, el usuario debía reiniciar la aplicación. Este error se solucionó haciendo que la aplicación se reiniciara automáticamente cada vez que cambiase el estado.

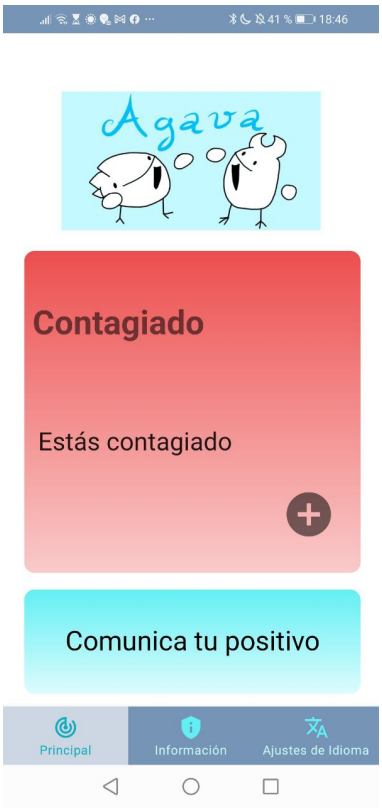


Figura 30: Aplicación con contagio

5.1.9. Envío de un código incorrecto al servidor

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones TCP y Usabilidad**

Se abre la aplicación estando el dispositivo conectado a Internet.

Tras ello, dentro de la aplicación se selecciona *Comunica tu positivo* y se rellenan los datos. La fecha solo permite seleccionar aquellas fechas entre la actual y 14 días atrás. El código solo permite introducir doce caracteres numéricos, formato de los códigos de contagio. De este modo se impide la inserción de un código estructurado de manera incorrecta.

Una vez el servidor recibe la información, comprueba que el código sea uno de los generados por la autoridad sanitaria, y por lo tanto correcto y activo, y de ser así introduce a la base de datos las claves y fechas recibidas. Dicha comprobación la realiza comparándolo con un listado de códigos que tiene almacenado en un *array*. En este caso no se produce ninguna coincidencia, por lo que rechaza la información recibida y no la introduce en la base de datos.

Este caso de prueba queda parcialmente incompleto, pues el cliente no recibe una retroalimentación sobre si el código es o no uno de los aceptados por el servidor; pero el servidor actúa como se esperaba, rechazando los identificadores recibidos y sin almacenarlos en la base de datos.

5.1.10. Envío de diagnóstico sin fecha

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones TCP y Usabilidad**

Se realiza el mismo proceso que en el anterior caso y se rellena únicamente el campo del código con uno correcto, dejando el de fecha vacío. A continuación se pulsa en aceptar y se confirma el envío. El mensaje se envía sin problema y nos proporciona el resultado esperado, que es la recepción del mensaje en el servidor.

5.1.11. Envío de diagnóstico sin inserción de código

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Usabilidad**

El proceso para llegar a la pantalla de envío es el mismo que en el caso anterior. Una vez se llega al formulario de *Comunica tu positivo* se pulsa el botón de *Aceptar* para enviar el código. Dado que no se han introducido datos, aparece un mensaje que indica que el código está incompleto.

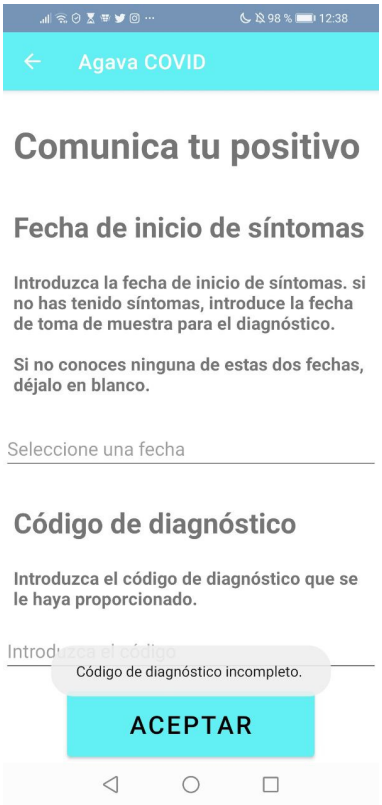


Figura 31: Aviso sobre código incompleto

5.1.12. Envío de código con menos de 12 cifras

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Usabilidad**

El proceso para llegar a la pantalla de envío es el mismo que en el caso anterior. Una vez se llega al formulario de *Comunica tu positivo* se escoge una fecha en el rango y se rellena el campo del código con 11 cifras. Tras ello se pulsa sobre aceptar.

Figura 32: Aviso sobre código incompleto

El resultado es el esperado, la aparición de un mensaje que nos indica que el código está incompleto y que no permite avanzar a la siguiente pantalla para aceptar el envío.

5.1.13. Envío de código con más de 12 cifras

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Usabilidad**

Se realiza el mismo proceso que en el caso anterior y se intenta insertar un código de 13 cifras. El campo del código nos impide superar las 12 cifras, por tanto el resultado es el esperado.

5.1.14. Envío de código con caracteres no numéricos

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Usabilidad**

Se realiza el mismo proceso que en el caso anterior y se intenta insertar un código con caracteres alfabéticos o símbolos. El propio campo de la aplicación cliente no acepta ese tipo de caracteres, por lo que la escritura de ellos no se admite.

5.1.15. Envío con fecha anterior a 14 días

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Usabilidad**

Se abre la aplicación estando el dispositivo conectado a Internet.

Tras ello, dentro de la aplicación se selecciona *Comunica tu positivo* y se rellenan los datos. La fecha solo permite seleccionar aquellas fechas entre la actual y 14 días atrás. Con esta restricción se impide el envío de identificadores con fecha inválida al servidor, lográndose el caso de prueba correctamente.

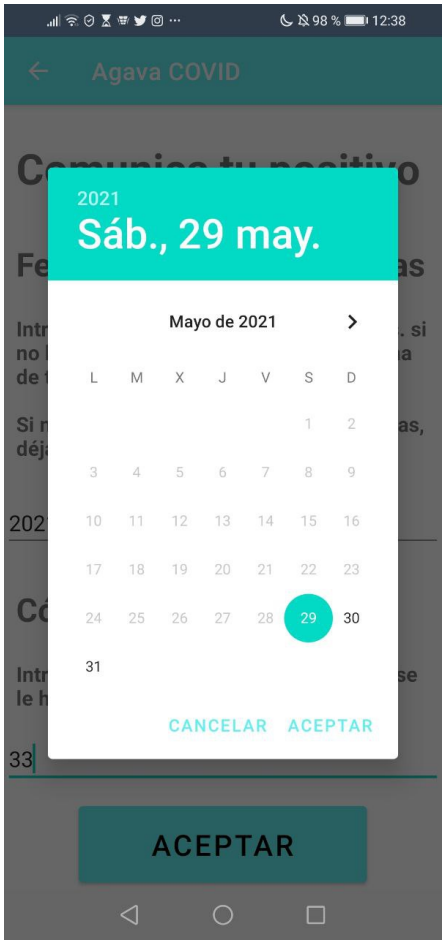


Figura 33: Límite anterior de aceptación de fechas

5.1.16. Envío con fecha posterior a 14 días

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Usabilidad**

Al igual que en el caso anterior, la aplicación cliente solo permite la selección de fechas 14 días anteriores a la actual. Con ello se impide el envío de identificadores con fechas futuras.

Figura 34: Límite posterior de aceptación de fechas

5.1.17. Multicast de IDs infectados desde el servidor a los clientes

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones UDP**

Para realizar el envío de identificadores, se activa el servidor. De manera periódica realiza un envío multicast del contenido de su base de datos, el cual es las claves y fechas generadoras asociadas a identificadores contagiados.

Inicialmente el envío no funcionaba correctamente, pues se empleaba una dirección IP para multicast fuera del rango de las reservadas para ello (es decir, las comprendidas entre la 224.0.0.0 hasta la 239.255.255.255).

Una vez se asignó una dirección de multicast correcta, se pudo comprobar empleando un *sniffer* de paquetes (*Wireshark*), que el envío y la petición de unión a dicha dirección se realizaba correctamente y de manera periódica.

5.1.18. Uso de la aplicación sin conexión a Internet (y sin recepción de multicast)

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Disponibilidad**

Se abre la aplicación y nada más ejecutarse sale el siguiente aviso en pantalla:

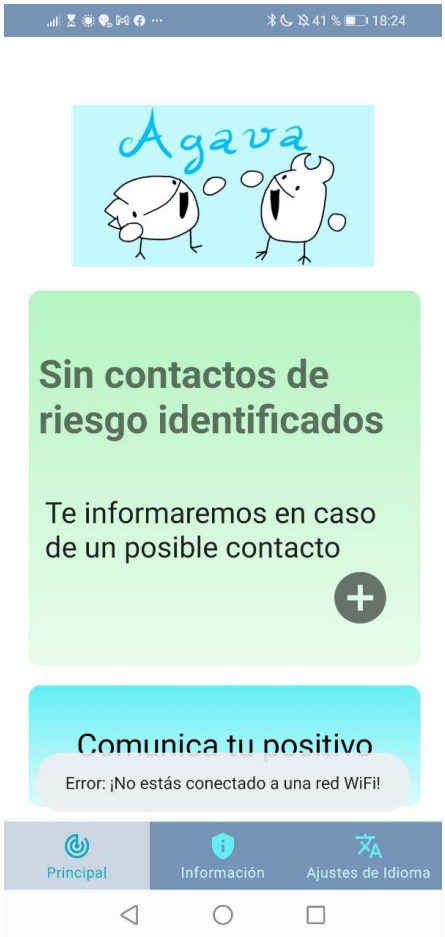


Figura 35: Aviso de desconexión

De este modo, la aplicación nos comunica que no está conectada a una red WiFi. Debido a ello algunas funciones no podrán realizarse, como es el envío de un código al servidor o la recepción de multicast.

Sin embargo, se puede navegar por la aplicación y esta no se queda colgada.

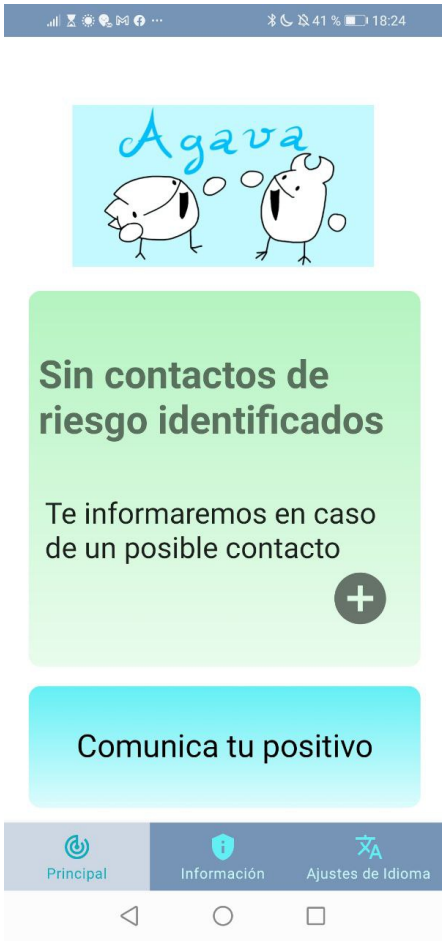


Figura 36: Aplicación ejecutándose correctamente

Del mismo modo, las conexiones Bluetooth se pueden realizar correctamente.

El funcionamiento es el esperado, pues al no haber conexión a Internet, es lógico que las funcionalidades dependientes de la red no puedan llevarse a cabo.

Por otro lado, aquellas que no dependen de una conexión a Internet siguen funcionando correctamente.

5.1.19. Recepción por multicast de un ID infectado que se encuentra en la base de datos local

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones UDP**

Teniendo el servidor activado realizando el envío periódico, se abre la aplicación, estando el dispositivo conectado a Internet.

Para la comprobación de la recepción del multicast se genera un pequeño mensaje en el cual se avisa con un texto de la recepción de un paquete desde una dirección IP.

Inicialmente la recepción no funcionaba debido a que la dirección empleada dentro del rango de direcciones multicast no era para redes internas. Tras cambiarla una vez más, la recepción funcionaba pero el resto de funcionalidades se bloqueaban.

Para solucionarlo se crearon dos hilos de ejecución, uno para la recepción del multicast y otro para las funcionalidades de la aplicación.

Tras separar el hilo de ejecución de cada proceso, la recepción del multicast dejaba de bloquear a las demás funcionalidades de la aplicación.

De este modo el funcionamiento es el correcto, pudiendo recibir multicast de manera periódica a la vez que se permite el uso de la aplicación correctamente.

Una vez se recibe este multicast, se obtiene la lista de claves y fechas generadoras, se calculan los identificadores y se comparan con los almacenados en la base de datos local, en concreto los de la tabla *ids_ajenos*. Se encuentra una coincidencia y con ello se cambia el valor de la variable que define el estado de contagio. Una vez cambiado, la imagen y los textos de la pantalla principal cambian al estado *Con contactos contagiados*.

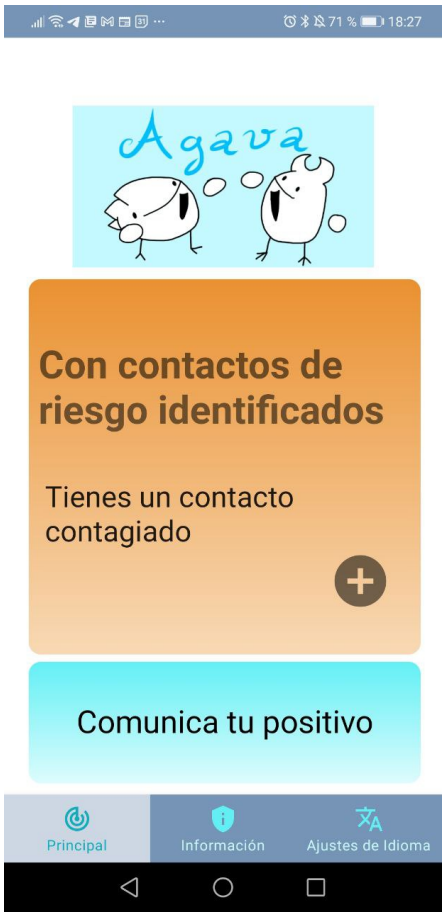


Figura 37: Aplicación ejecutándose correctamente

5.1.20. Recepción por multicast de IDs infectados y ninguno se encuentra en la base de datos local

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones UDP**

Teniendo el servidor activado realizando el envío periódico, se abre la aplicación, estando el dispositivo conectado a Internet.

Para la comprobación de la recepción del multicast se genera un *toast*, en el cual se avisa con un texto de la recepción de un paquete desde una dirección IP.

Una vez se recibe este multicast, se obtiene la lista de claves y fechas generadoras, se calculan los identificadores y se comparan con los almacenados en la base de datos local, en concreto los de la tabla *ids_ajenos*. Al no encontrarse ninguna coincidencia, no cambia el valor de la variable de contagio, permaneciendo en *sin contactos*.
El caso de prueba acaba con éxito.

5.1.21. Recepción por multicast de IDs infectados y ninguno se encuentra en la base de datos local, pero el ID es un número por encima de uno almacenado

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones UDP**

Teniendo el servidor activado realizando el envío periódico, se abre la aplicación, estando el dispositivo conectado a Internet.

Para la comprobación de la recepción del multicast se genera un *toast*, en el cual se avisa con un texto de la recepción de un paquete desde una dirección IP.

Una vez se recibe este multicast, se obtiene la lista de claves y fechas generadoras, se calculan los identificadores y se comparan con los almacenados en la base de datos local, en concreto los de la tabla *ids_ajenos*. Al no encontrar ninguna coincidencia, pues el identificador generado con la clave y la fecha ha de ser exacto a alguno de los almacenados para detectar un contacto, no cambia el valor de la variable de contagio, permaneciendo en *sin contactos*.
El caso de prueba acaba con éxito.

5.1.22. Recepción por multicast de IDs infectados y ninguno se encuentra en la base de datos local, pero el ID es un número por debajo de uno almacenado

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones UDP**

Teniendo el servidor activado realizando el envío periódico, se abre la aplicación, estando el dispositivo conectado a Internet.

Para la comprobación de la recepción del multicast se genera un *toast*, en el cual se avisa con un texto de la recepción de un paquete desde una dirección IP.

Una vez se recibe este multicast, se obtiene la lista de claves y fechas generadoras, se calculan los identificadores y se comparan con los almacenados en la base de datos local, en concreto los de la tabla *ids_ajenos*. Al no encontrar ninguna coincidencia, pues el identificador generado con la clave y la fecha ha de ser exacto a alguno de los almacenados para detectar un contacto, no cambiaría el valor de la variable de contagio, permaneciendo en *sin contactos*.
El caso de prueba acaba.

5.1.23. Recepción por multicast de IDs infectados y no se posee ningún ID en la base de datos local con los que comparar

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones UDP**

Teniendo el servidor activado realizando el envío periódico, se abre la aplicación, estando el dispositivo conectado a Internet.

Para la comprobación de la recepción del multicast se genera un *toast*, en el cual se avisa con un texto de la recepción de un paquete desde una dirección IP.

Una vez se recibe este multicast, se obtiene la lista de claves y fechas generadoras, se calculan los identificadores y se procede a comparar con la base de datos local. Esta, al estar vacía, no devuelve nada y por lo tanto no se encuentran coincidencias.
Tras esto, el caso de prueba termina exitoso sin realizar ningún cambio en la interfaz.

5.1.24. Envío del multicast pero sin recepción (clientes inactivos)

- **Tipo de prueba: Prueba funcional**
- **Ámbito de la prueba: Comunicaciones UDP**

Se inicia el servidor, conectado a la red, pero sin abrir la aplicación cliente en ningún momento.
El servidor realiza de manera periódica, un envío a la dirección IPv4 del grupo de multicast con la información de las claves y fechas generadoras de los identificadores contagiados.
También, de manera periódica hace un llamamiento a que los dispositivos de dicho grupo se unan a él.
Estos envíos se realizan sin necesidad de que haya clientes de dicho grupo de multicast conectados.
El resultado es el esperado, pues es necesario que este envío se realice en todo momento para que siempre puedan unirse nuevos clientes cuando se conecten a la red.

5.1.25. Escucha, por parte del cliente, de multicast pero sin envío (servidor inactivo)

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Disponibilidad**

Se inicia la aplicación cliente, con conexión a la red. La espera para recepción de paquetes vía multicast se queda en segundo plano, y el resto de la aplicación funciona correctamente, permitiendo su uso.

La aplicación no envía nada por red, como era de esperar, manteniéndose a la escucha de paquetes multicast de su grupo.

5.1.26. Cambio de idioma

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Usabilidad**

Se inicia la aplicación y se va a la pantalla de *Ajustes de idioma*. Una vez ahí, se selecciona el idioma al que se desea cambiar. En este caso, seleccionamos *Inglés*. Una vez pulsado *Aceptar*, la aplicación se reinicia, haciendo un breve pestañeo.
Tras ello, el idioma de los textos aparece cambiado al inglés.

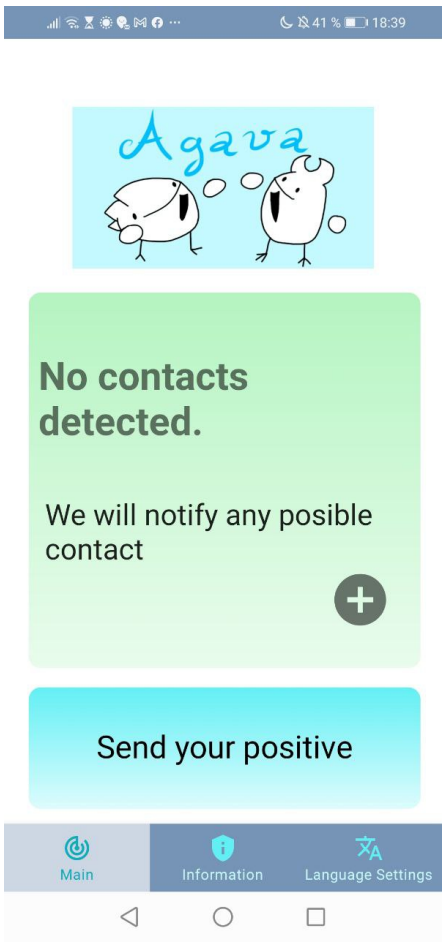


Figura 38: Aplicación en inglés

5.2. Pruebas de caja blanca

Las pruebas de caja blanca son aquellas diseñadas con conocimiento profundo del software. Esto nos permite comprobar funcionalidades concretas del propio software, como por ejemplo comprobar el nivel de seguridad de la aplicación desarrollada.

5.2.1. Escucha del canal de conexión en el momento de envío de un código con los IDs al servidor

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Seguridad**

Se abre la aplicación estando el dispositivo conectado a Internet.

Tras ello, dentro de la aplicación se selecciona *Comunica tu positivo* y se rellenan los datos.

Se inicia un programa de sniffing y se envía el código desde la aplicación.

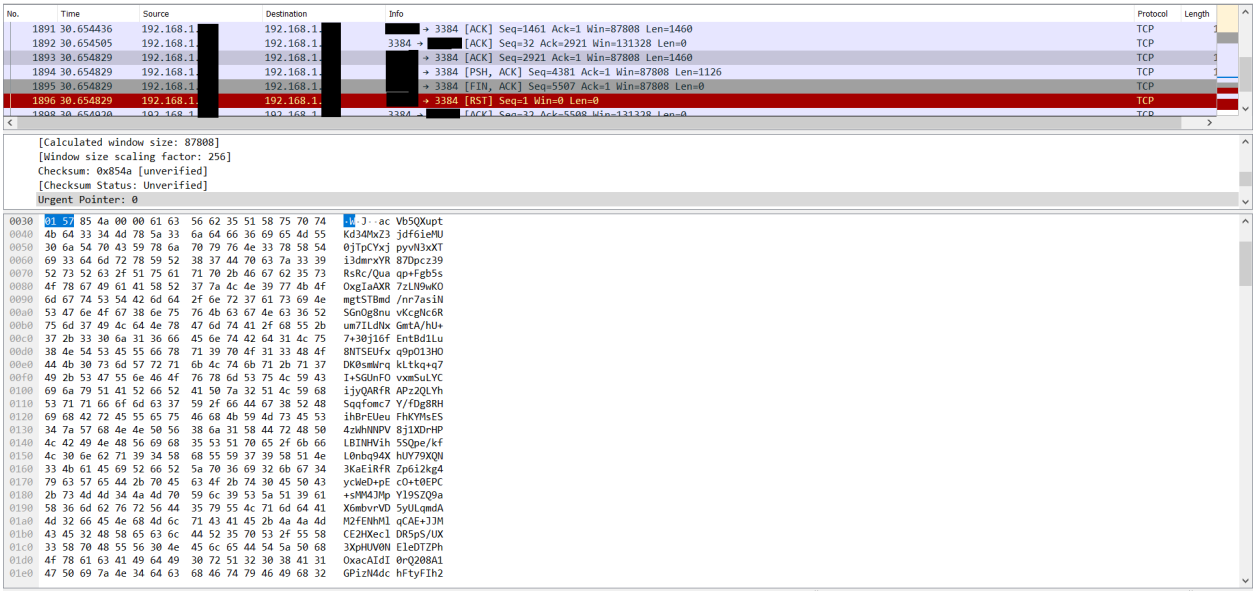


Figura 39: Resultado del programa de sniffing

El resultado nos dice que el mensaje que envía la aplicación viaja cifrado y que por tanto está salvo de observadores externos.

5.2.2. Escucha del canal de conexión en el momento de envío de un código incorrecto al servidor

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Seguridad**

El proceso es el mismo que en el caso anterior.

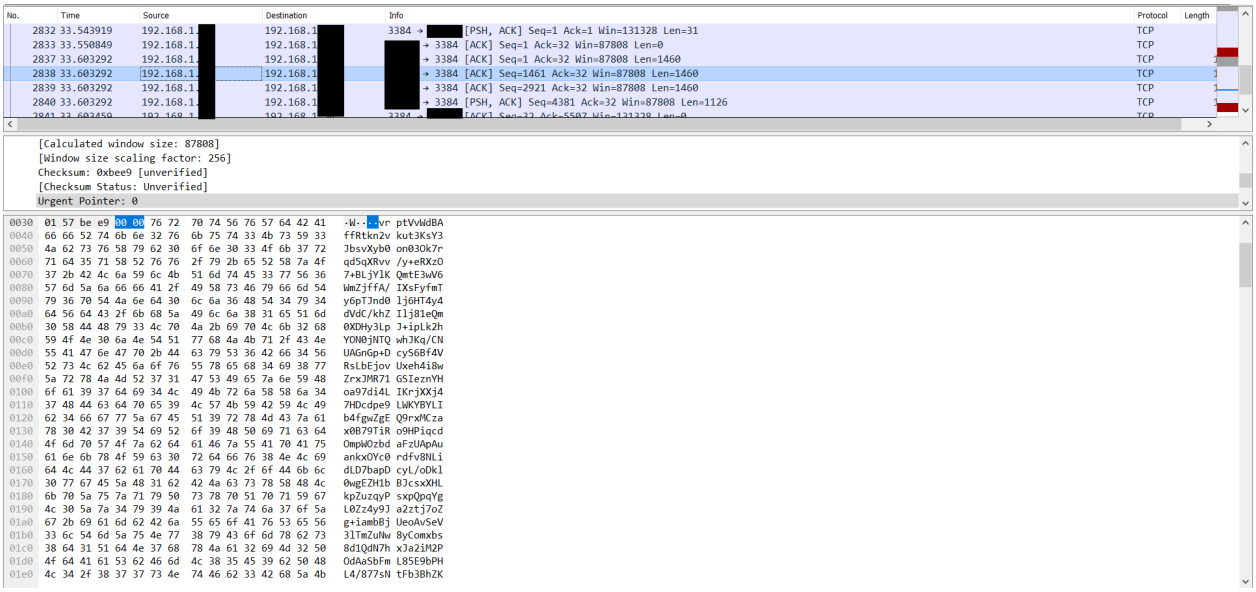


Figura 40: Resultado del sniffing

Como vemos, al igual que en el caso anterior, el resultado es que se puede ver el mensaje cifrado y que por tanto está a salvo de observaciones no deseadas.

5.2.3. Escucha del canal de conexión en el momento del envío del multicast

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Seguridad**

Se inicia un programa de sniffing filtrando las direcciones de multicast (224.0.0.0 en adelante hasta 224.0.0.255).[25] Se abre la aplicación estando el dispositivo conectado a Internet.

El resultado es que el mensaje de multicast se puede observar sin ningún tipo de impedimento. Esto se debe a que en las conexiones UDP no hay un *handshake* donde se puedan intercambiar claves para poder realizar un cifrado.

5.2.4. Barrido de puertos de la máquina cliente

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Seguridad**

Se inician tanto la máquina cliente como la máquina servidor. Para la realización de esta prueba se utilizó una máquina virtual Kali y la herramienta *nmap*. La información que devuelve nmap tras usarse, incluye el barrido de puertos realizado junto a el número de estos, protocolo, el nombre del servicio y su estado (abierto, cerrado, filtrado o no filtrado). Si el estado se marcara como *filtrado* esto significa que el puerto está siendo bloqueado por un firewall u otro software similar.

Para llamar a nmap se escribe el comando de la forma:

```
# nmap <Aquí los argumentos> <Aquí el nombre del host>
```

En concreto el comando utilizado fue:

```
sudo nmap -p <PUERTO> 192.168.1.N
```

Donde en <PUERTO>se especifica el puerto a escanear y donde N es el último bloque de la dirección IP en la red local.

Se analizó el puerto 4446 (multicast de recepción).

Inicialmente se obtuvo como respuesta que el puerto estaba cerrado. Esto no tenía sentido, pues el cliente estaba recibiendo paquetes en ese momento. El problema era que el análisis se estaba realizando vía TCP, forma predeterminada de *nmap*.

Tras especificar en el comando el análisis de puertos UDP con el parámetro *-sU*, quedando el comando:

```
sudo nmap -sU -p 4446 192.168.1.N
```

Se obtuvo una respuesta más lógica:

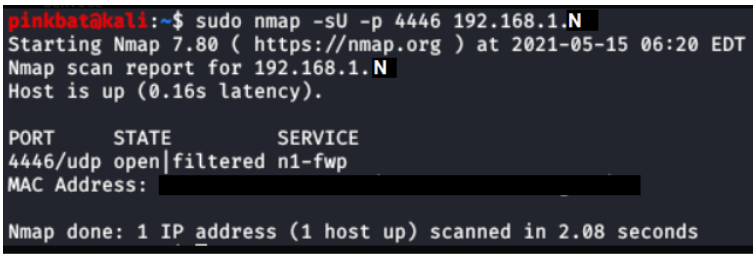


Figura 41: Puerto 4446. Recepción de multicast. UDP

Aquí se puede ver que el puerto está *open|filtered*. Dicho estado significa que puede estar abierto o filtrado. Al tratarse de paquetes UDP, los paquetes para realizar el escaneo se envían sin carga. Debido a esto, el puerto los descarta incluso estando abierto, pues no poseen contenido. Por tanto *nmap* no puede concretar si el estado es abierto o filtrado por un firewall.

Se ha podido comprobar que la aplicación solamente abre el puerto necesario para establecer las comunicaciones con el servidor, y ninguno más.

5.2.5. Barrido de puertos de la máquina servidor

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Seguridad**

Se inician tanto la máquina cliente como la máquina servidor. Para la realización de esta prueba se utilizó una máquina virtual Kali y la herramienta *nmap*. El comando utilizado es el siguiente:

Para llamar a nmap se escribe el comando de la forma:

```
# nmap <Aquí los argumentos> <Aquí el nombre del host>
```

En concreto el comando utilizado fue:

```
sudo nmap -p <PUERTO> 192.168.1.N
```

Donde en <PUERTO >se especifica el puerto a escanear y donde N es el último bloque de la dirección IP en la red local.

Se analizaron los puertos 4445 (multicast de envío), 3327 (puerto de conexión de MariaDB) y 3384 (puerto de recepción de datos vía TCP en el servidor).

Para el análisis del puerto 4445, el cual envía los paquetes de multicast, se emplea el parámetro *-sU*, pues el envío de dichos paquetes se realiza vía UDP.

Para los otros dos puertos se realiza un escaneo habitual, vía TCP, que es el protocolo por el que admite las conexiones habituales desde un cliente.

Los resultados son los siguientes:

Para el envío de multicast vía UDP:

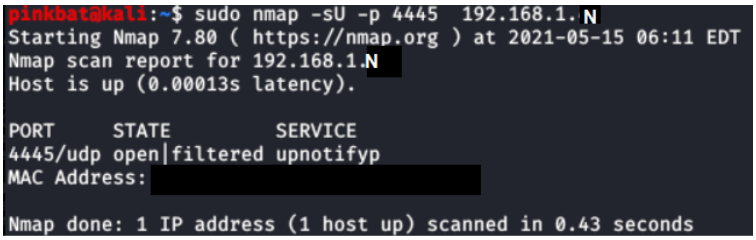


Figura 42: Puerto 4445. Envío de multicast. UDP.

El resultado obtenido es *open|filtered*. Dicho estado significa que puede estar abierto o filtrado.

Para la escucha de paquetes para la base de datos *MariaDB* vía TCP:

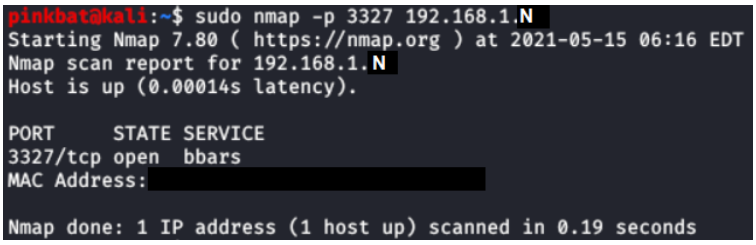


Figura 43: Puerto 3327. Acceso a base de datos. TCP.

Para la escucha de conexiones con el servidor de paquetes con códigos de contagio vía TCP:

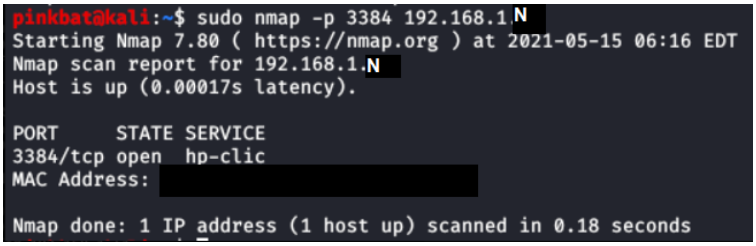


Figura 44: Puerto 3384. Envío de códigos. TCP.

El resultado en ambos casos es puertos abiertos. Por lógica, el puerto UDP también estará abierto.

En el caso de desplegar la aplicación servidor en una máquina servidor, una de las formas de evitar el escaneo de puertos sería la instalación de un cortafuegos o el uso de puertos *honeypot*, los cuales pueden servir para atrapar en bucle bots atacantes.

5.2.6. Ataque DDoS

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Seguridad**

Se inicia el dispositivo, conectado a la red. Para poder realizar el ataque se utilizó una máquina virtual Kali y la herramienta *inviteflood*. Este comando envía paquetes vía *UDP* de forma masiva con la intención de que el objetivo se sature y no pueda realizar sus funciones de red correctamente.

En concreto el comando utilizado fue el siguiente:

```
sudo inviteflood eth0 '' 192.168.1.N 192.168.1.N 700000 -D 4446
```

En dicho comando, hay argumentos obligatorios, los cuales son:

- **Interfaz.** En este caso es *eth0*, y es la interfaz o red donde tanto el objetivo como la máquina atacante deben estar conectados.
- **Usuario objetivo.** Aquí se debe especificar el usuario de la máquina objetivo al que se va a atacar. En este caso no hay, luego se deja con comillas vacías.
- **Dominio objetivo.** Este campo admite tanto direcciones URL como direcciones IPv4. Dado que es un servidor montado en una red local, aquí se especifica su IPv4 *192.168.1.N* donde N es el último bloque de la dirección IP en la red local.
- **Objetivo.** En este campo se especifica la dirección IPv4. En caso de haberla puesto en el anterior punto, se repite.
- ***Flood stage.*** En este campo se introduce el número de paquetes UDP que se mandarán para realizar el ataque. En este caso se realizó un ataque con 700000 paquetes.

Además, en este caso se va a realizar un ataque al puerto 4446.

Tras ejecutar el comando, y pasados unos 2 segundos, se obtuvo la siguiente pantalla:

Figura 45: Resultado de ataque de denegación de servicio.

Tras la recepción de dicha pantalla un par de veces, el dispositivo se saturó, y dejó de poder conectarse a la red hasta que se reinició su conexión.

Una forma de evitar esto es abriendo el puerto únicamente durante el envío del código. Una vez terminada la comunicación, este se cierra para evitar la llegada de tráfico indeseado al mismo.

5.2.7. Inyección de código desde la aplicación cliente

- **Tipo de prueba: Prueba no funcional**
- **Ámbito de la prueba: Seguridad**

Existen dos vías para la posible inyección de código:

- **Campo de selección de fecha.** Aquí el usuario puede seleccionar una fecha de un calendario que aparece cuando hace click, denegando la posibilidad de escribir caracteres de forma libre. Como esto ocurre siempre que se quiere cambiar este campo no es posible inyectar ningún tipo de código.
- **Campo de introducción de código de contagio.** En este campo se le pide introducir un código al usuario. Los caracteres están restringidos tal que solo se admiten dígitos. De este modo se bloquea la posibilidad de escribir inyecciones de código en la base de datos del servidor, pues no es posible introducir caracteres tales que % para especificar caracteres vía código URL, buscando nombres o direcciones, u otros admitidos en el lenguaje SQL (guiones, comillas...).

6. Análisis de riesgos de seguridad y privacidad

En este apartado se procede a realizar el análisis de riesgos a la aplicación realizada. De este modo se busca detectar aquellas malas prácticas en el tratamiento de los datos, así como la implementación de salvaguardas y medidas necesarias para contrarrestarlas

También se resaltan aquellas ventajas y desventajas de una aplicación de rastreo de contactos frente a otras opciones existentes para el control del *estado COVID*, así como comparar sus posibilidades y el nivel de protección de los datos que puede llegar a garantizar.

6.1. Estado del arte: Seguridad y Privacidad

Para elaborar este análisis primeramente se presenta el Estado del Arte referente a la seguridad y privacidad a nivel europeo. En este se tratarán aquellos puntos propios del RGPD que afecten a componentes de la aplicación aquí desarrollada como son por ejemplo, las comunicaciones Bluetooth o las aplicaciones móviles.

También se realiza un especial enfoque en la situación de la Unión Europea, área abarcada por la normativa del RGPD, con respecto a las aplicaciones de rastreo de contactos y otras alternativas existentes para el control del estado COVID.

6.1.1. Reglamento General de Protección de Datos

El Reglamento General de Protección de Datos es el reglamento en vigor dentro de la Unión Europea, relativo a la privacidad, circulación y protección de los datos de las personas físicas. Por ello, cualquier institución, pública o privada, que o bien pertenezca a la Unión Europea o bien trate con ella, debe cumplirlo.[59]

Entró en vigor el 24 de mayo de 2016, momento en el que las empresas tuvieron que ir adaptándose a este, con fecha límite el 25 de mayo de 2018, a partir de la cual se ha estado aplicando.

El RGPD ha dado directrices respecto a diversos ámbitos que se ven involucrados en el desarrollo de aplicaciones de rastreo de contactos. Una de ellas es entorno a las aplicaciones móviles, que al estar en dispositivos que siempre llevamos con nosotros, han de tratarse con ciertos matices.

La normativa general implica que deben especificarse los tratamientos de los datos en todos los aspectos, es decir, su recolección, conservación, respaldo, uso, posibilidad de modificación, comunicación, archivo y su destrucción. El cliente debe poder en todo momento ejercer sus derechos de borrado y modificación de esos datos.

Dentro del mundo de las aplicaciones móviles esto, por supuesto, no es excepción.

La normativa del RGPD ha de aplicarse a cualquier aplicación móvil que recolecte datos de los ciudadanos europeos y eso incluye la aplicación aquí desarrollada que, además, trata con información sensible como son datos de salud. Son por ello los aspectos que van a ser analizados.

- Consentimiento explícito por parte de los usuarios para recolectar sus datos personales.
- Protección de datos por diseño y por defecto.
- Acceso del usuario a los datos recolectados.
- Derecho de los usuarios a la portabilidad de los datos.
- Derecho al olvido.
- Implementación de las reglas de forma estricta.
- Derecho del usuario a conocer cuándo se han violado los datos personales.

Consentimiento explícito

Esta normativa viene marcada por el «Consideración 42 Pruebas y Requisitos para el Consentimiento» del Reglamento General de Protección de Datos.

Se ha de proporcionar consentimiento de forma activa, informando sobre qué datos se recolectarán, antes de recoger y procesar esa información personal de los usuarios.

De esta forma la institución dueña de la aplicación, debe de ser capaz de demostrar que tiene el consentimiento del sujeto cuyos datos se están recopilando, al igual que el usuario debe ser consciente del tratamiento de datos que se hace.

La manera en la que se solicita el consentimiento debe ser en un lenguaje conciso, accesible y con términos claros.

Además, el usuario debe ser consciente de la identidad del sujeto que recopilará su información.

Este consentimiento debe poder darse de manera libre, es decir, por elección propia del usuario, con posibilidad de rechazarlo y de modificarlo posteriormente.[72]

Derechos de los individuos

Además del consentimiento, los usuarios de aplicaciones móviles tienen derechos adicionales sobre el control y tratamiento de los datos.

Estos deben ser mencionados en la Política de Privacidad de la aplicación. Son los siguientes.

Derecho al acceso a los datos recolectados

En el Artículo 15 «Derecho de acceso del interesado» del RGPD se dictamina este derecho.[67]

El usuario debe poder conocer los siguientes puntos además de qué datos se están recolectando:

- El propósito del procesamiento.
- Las categorías de los datos personales recopilados.
- Los destinatarios o categorías de destinatarios a los que estos datos serán comunicados, sobre todo si se tratan de países terceros u organizaciones internacionales.
- Periodo durante el cual los datos serán almacenados.
- La existencia del derecho a borrar los datos o restringir el procesamiento de los datos.
- El derecho a presentar una queja ante una autoridad supervisora.
- Si no se recopilan datos personales directamente del usuario, fuentes de datos del mismo a las que tengan acceso.
- La existencia de toma de decisiones automatizadas, como la elaboración de perfiles. Se debe dejar a disposición del usuario la lógica que emplea el programa que toma dichas decisiones.

En caso de que se traspasen datos a otros países, el usuario debe ser consciente de ello. Este debe poder acceder a los datos que se están recopilando, y el derecho a obtener esta copia no debe afectar los derechos o libertades de otros.

Derecho a restringir el procesamiento de datos

Acorde al artículo 18 «Derecho a la limitación del tratamiento» del RGPD, los usuarios tienen el derecho a restringir el tratamiento de sus datos si se da alguno de los siguientes casos.[66]

- Los datos son incorrectos.
- El procesamiento es ilegal.
- Los datos no se necesitan para el propósito originalmente estipulado.
- El individuo se opone al procesamiento de sus datos.

El derecho a la portabilidad de los datos

En el caso de que los datos sean tratados de manera automatizada, los usuarios tienen el derecho a la *portabilidad de sus datos*.

Esto significa que tienen el derecho a transmitirlos a otras aplicaciones móviles o negocios sin que interfiera la aplicación que originalmente los ha recolectado.

Este usuario también puede solicitar que la aplicación original transmita esos datos a otra, y esto ha de cumplimentarse siempre y cuando no se viole la ley.

Derecho a oponerse a la recolección de sus datos

En el artículo 21 «Derecho de oposición» del RGPD[68] se describe el derecho a solicitar que la aplicación deje de procesar sus datos si se usan para alguno de los siguientes fines:

- Procesamiento para la elaboración de perfiles.
- Marketing directo.
- Procesamiento para investigación científica, estadística o histórica.

Derecho a la rectificación

El artículo 16 «Derecho de rectificación» del RGPD[69] dictamina que los usuarios deben tener el derecho a poder rectificar datos incorrectos referentes a su información personal. El cómo ejercer este derecho debe ser explicado en la Política de Privacidad.

Principio de transparencia

El principio de transparencia implica que los usuarios han de estar en posesión del derecho a ser informados. Esto implica que han de ser conscientes de qué datos se están recolectando y con qué fines. Esta información debe ser accesible de manera sencilla y gratuita, y ser fácil de comprender. Este derecho es definido en la consideración número 58 «Principio de transparencia» del RGPD.[71]

Especial atención en el caso de los niños, donde, si el comunicado fuera dirigido a ellos, debería emplearse un lenguaje que puedan entender.

Derecho al borrado

El derecho al borrado o al olvido implica que el usuario, en caso de que sus datos ya no sean necesarios para el propósito para el que habían sido recolectado, pueda solicitar su borrado.

También podrán ejercer su derecho en caso de que los datos se traten de manera ilegal.

6.1.2. Delegado de Protección de Datos

En algunos casos, será necesario contar con un Delegado de Protección de Datos, que se encargará de que se cumpla debidamente la legislación. [34]

Es necesario contar con uno si:

- Se es una entidad pública.
- Se procesan y monitorizan datos de ciudadanos de la Unión Europea de manera regular.
- Se tratan con categorías especiales de datos personales o datos personales relacionados con antecedentes criminales u ofensas.

6.1.3. Seguridad de los datos

El RGPD dictamina que los controladores y procesadores de datos deben asegurar la privacidad y seguridad de los datos. Esto incluye el uso de algoritmos criptográficos actualizados.

El artículo 32 del RGPD, «Seguridad del tratamiento», también recomienda el uso de pseudónimos a modo de identificar a los usuarios.[74]

Los propietarios de la aplicación deben asegurar la confidencialidad, integridad, disponibilidad y resiliencia de aquellos sistemas que procesen los datos.

En caso de un incidente, el procesador de datos debe restaurar la disponibilidad de los datos cuanto antes.

6.1.4. Evaluación de Impacto de Protección de Datos

Una Evaluación de Impacto de la Protección de Datos, como la que se elaborará más adelante en este proyecto, es una evaluación de los riesgos existentes de que se produzca una infracción de seguridad. Se debe realizar a todas las aplicaciones, con especial prioridad aquellas que traten datos sensibles de los usuarios.

En caso de suceder una brecha de información, el controlador de datos debe informar inmediatamente a los usuarios y autoridades.

También se ha de poseer un plan de contingencia y actuación, en el que se defina cómo actuar en caso de que se produzca una brecha de datos.

6.1.5. Unión Europea y aplicaciones de rastreo de contactos

La pandemia creó una situación ideal para la recolección de datos de usuarios. Es por ello, que lo más esperado hubiera sido crear protocolos que abiertamente hubiesen sido cien por cien centralizados, y hubiesen tratado a los usuarios como productos.

Sin embargo, no fue directamente así. La idea inicial fue crear un protocolo que emplease pseudónimos para identificar a los usuarios y así preservar su información.

Aun así, este protocolo comenzó a tener el apoyo de empresas como Google y Apple, los cuales comenzaron a implementar ciertos requisitos debido a sus políticas de desarrollo que, al final, no ayudaron a que estas aplicaciones fuesen del todo privadas. Requisitos como el uso de la geolocalización a la vez que Bluetooth Low Energy o la existencia de balizas que se comunican vía BLE con los dispositivos móviles para recolectar datos.

Aun así, inicialmente la Unión Europea luchó por mantener este protocolo lo más respetuoso con la privacidad posible e, inicialmente, se convirtió en una solución esperanzadora.[39]

Una de las aplicaciones de rastreo de contactos que adoptó el extendido protocolo DP-3T fue StopCOVID, propuesta en Francia.

Los miembros de la *Commission Nationale de l'Informatique et des Libertés* (CNIL) hicieron pública su opinión sobre la aplicación de rastreo de contactos StopCOVID el día 24 de abril de 2020.[18]

Indicó que, dada la situación excepcional de crisis pandémica, la aplicación podía ser recomendable siempre y cuando cumpliera una serie de requisitos técnicos y de privacidad. Además, la aplicación debía pertenecer a una estrategia general de salud y su utilidad debía ser demostrada.

Posteriormente, la aplicación fue presentada ante el parlamento francés, donde en caso de ser aprobada, la CNIL volvería a analizarla para examinar su implementación.

Finalmente la aplicación fue aprobada, aunque originó diversos debates. Incluso llegó a haber a una actualización, *TousAntiCovid*, mejorada con acceso a información de salud y basada en evidencias sobre la pandemia.[32]

La CNIL en 2020 admitió el uso de este tipo de aplicaciones, pues se han diseñado con el concepto de protección de datos por diseño, utilizando identificadores anónimos a modo de pseudónimos de los usuarios. Además, tampoco permiten recuperar listas de personas contaminadas, pues los identificadores almacenados en el servidor siempre permanecerán anonimizados.

Sin embargo, a lo largo del año comenzaron a surgir problemas con el protocolo. El primero, en octubre de 2020, implicaba que las aplicaciones diseñadas con DP-3T solo poseían tráfico de datos dirección cliente a servidor en el momento de notificar un contagio. De este modo, mediante un ataque de escucha, se podía identificar aquel tráfico de datos que solo se originaba desde clientes contagiados. De este modo, podía averiguarse qué usuarios habían dado positivo en una PCR.

Este problema se solucionó rápidamente, generando ruido que se envía de forma aleatoria desde todos los dispositivos clientes al servidor.

Sin embargo, debido a la poca acogida que estas aplicaciones tuvieron a lo largo de los dos años, 2020 y 2021, la Unión Europea comenzó a investigar soluciones alternativas. Los ciudadanos no parecían muy contentos con las aplicaciones de rastreo de contactos y muy pocos realmente la llevaban activa en el móvil. Este hecho, unido a que la aplicación necesita comunicarse con otros dispositivos con la aplicación para intercambiar los identificadores, imposibilita totalmente el que la aplicación logre su cometido.[33]

Esto se vio propiciado también debido a los cambios en las necesidades en torno al estado COVID.

Mientras que en el primer año de la pandemia se buscaba romper cadenas de contagio, comunicando a los ciudadanos cuándo habían mantenido contacto con una posible persona contagiada; actualmente, en 2021, se busca conocer qué personas están vacunadas, cuándo lo han hecho, cuántas dosis han recibido, si han pasado la enfermedad... Todo ello con el objetivo de controlar qué población está ya inmunizada.

Este nuevo enfoque ha ocasionado que, a lo largo del desarrollo de este proyecto, las aplicaciones de rastreo de contacto hayan perdido fuerza en la Unión Europea y en todo el mundo, dejándose de fomentar su uso y buscando nuevas soluciones para los nuevos problemas.

6.1.6. Unión Europea y estado COVID

Como se ha mencionado antes, las necesidades de la Unión Europea sobre el control del estado COVID de sus ciudadanos, ha variado mucho a lo largo de este año.

Inicialmente todo se centró en un plan de contingencia del virus y en la búsqueda de maneras para frenar su expansión. Es por ello que en la Unión Europea se fomentó el desarrollo de protocolos de rastreo de contactos.

Primeramente, el más popular fue el Pan-European Privacy-Preserving Proximity Tracing, un protocolo centralizado introducido el 1 de abril de 2020.

Sin embargo, tan solo 19 días más tarde, el Centro Helmholtz para la Seguridad de la Información se retiró públicamente del consorcio debido a una «falta de transparencia y gobernanza clara» y preocupados por si el protocolo era lo suficientemente cauto con los datos de sus usuarios.

Ese mismo 20 de abril se publicó la carta ya mencionada en el apartado sobre PEPP-PT/PEPP, donde más de 300 académicos de seguridad y privacidad criticaron el enfoque de este protocolo, pues permitía volver a obtener los datos anonimizados a partir de los pseudónimos.

Fue tras ello que el grupo formado por la École Polytechnique Fédérale de Lausanne, ETH Zurich, KU Leuven y el Institute for Scientific Interchange desarrollaron el protocolo Decentralized Privacy-Preserving Proximity Tracing.

Este marcó la diferencia siendo descentralizado, y a partir de ahí los subsiguientes protocolos seguirían en su mayoría, dentro de la Unión Europea, el paradigma descentralizado.

Sin embargo, las necesidades han ido cambiando a lo largo del año 2021, y con el desarrollo de las vacunas, ahora prima el control de datos sobre las dosis administradas.

Es por ello que la Unión Europea busca otorgar a sus ciudadanos algún tipo de certificado donde se puedan consultar datos como tipo de vacuna, número de dosis que ese individuo ha recibido, fecha, si ha pasado o no la enfermedad del COVID-19...

De la mano de esta necesidad comienza a incentivarse la investigación entorno a la Identidad Digital y la posibilidad de crear un sistema descentralizado donde los usuarios puedan ser realmente propietarios de sus datos. Estos se almacenarían en sus propios dispositivos, de manera local, y se llevaría únicamente un registro o histórico de las operaciones realizadas, anonimizando los datos de los involucrados y sin mostrar los datos presentados en ella.

De este modo, la privacidad es absoluta, priorizando ante todo la minimización de los datos.

El día 1 de junio de 2021, Ursula von der Leyen anunció que la Unión Europea busca otorgar a sus ciudadanos un nuevo modelo de identidad digital, descentralizado y privado, donde sean los usuarios propietarios de sus identidad.[57]

El día 3 de junio de 2021, se anunció el soporte para esta identidad digital a través del uso de *wallets*, o monederos digitales.[60]

Esta identidad digital apunta al modelo de Self-Sovereign Identity.

Con el modelo de Self-Sovereign Identity se logra un sistema descentralizado en el que el usuario es dueño de sus datos, elige qué información quiere realmente compartir con qué proveedores de servicios y organizaciones, y permite emplear las mismas credenciales para identificarse ante cualquier sistema.

Modelo Self-Sovereign Identity y Credenciales Verificables

Este modelo para la gestión de la Identidad Digital consta de tres tipos de entidades, las cuales interactúan entre sí mediante una serie de tokens y datos, llamados Credenciales Verificables y Presentaciones Verificables.

Se trata de un modelo descentralizado, cuya aplicación se da en la *blockchain*. Esta estructura almacena la información de las operaciones que se lleven a cabo en ella en múltiples *nodos* u ordenadores. De este modo, siempre se poseerá una copia de ello, evitando la pérdida de la información. Además, la información almacenada refiere únicamente a los resultados de las operaciones, con las firmas digitales de los involucrados. De este modo se confirma su validez y el no repudio. Cada operación se almacena en lo llamado *un bloque*, donde cada bloque, además, posee una referencia a la información del anterior a modo de resumen hash. De esta manera, la información se vuelve inmutable, pues para cambiar el contenido de un bloque, habría que alterar el contenido de todos los bloques subsiguientes.

Las Credenciales Verificables consisten en una serie de datos clave-valor sobre una entidad, en este caso modelados acorde al estándar del World Wide Web Consortium sobre Credenciales Verificables. [77]

Una de estas Credenciales Verificables pudiera ser la información del estado COVID de una persona, cuyos datos se modelan acorde al estándar.

Una iniciativa con mucho impulso que ya está trabajando en este tipo concreto de Credenciales Verificables es *COVID Credentials Initiative* (CCI).[21]

CCI es una comunidad global abierta que colabora para estandarizar Credenciales Verificables del ámbito de la salud. Buscan, ante todo, la preservación de la privacidad y que los datos se generen, almacenen y administren a prueba de manipulaciones.

Son miembros de la *Linux Foundation Public Health* (LFPH) desde diciembre de 2020, desarrollando especificaciones para dar a las Credenciales COVID un enfoque de código abierto basado en estándares públicos.

Todo tipo de Credenciales Verificables se almacenan en una *wallet* o monedero digital del usuario, que funciona de manera local en su dispositivo de preferencia. Estas Credenciales Verificables permiten presentar los datos mínimamente necesarios para acceder a algún tipo de servicio. Esto se logra mostrando a la entidad solicitante únicamente cada dato solicitado y ninguno más. Así se evita mostrar información excesiva e innecesaria como, por ejemplo, cuando se muestra el DNI para demostrar la mayoría de edad, donde además se están enseñando datos como *nombre*, *apellidos*, *fecha de nacimiento*...

Su validez se logra al ser firmadas por una entidad capacitada para ello. Para ello, dicha entidad hace uso de su clave privada, firmando el token representativo de la Credencial Verificable.

Dentro del reglamento de *electronic IDentification, Authentication and trust Services* (eIDAS), es eIDAS Bridge el que otorga a dichas entidades la capacidad de ser cualificadas, para que su firma sea legalmente válida, y por lo tanto regulada. Próximamente dicha regulación se verá extendida con la llegada de eIDAS 2 y la regulación de monederos digitales en la UE. [48] [29]

El registro, no repudio y la imposibilidad de modificación de estas se logra mediante la escritura de las mismas en el *ledger* o histórico de transacciones de la blockchain. Con ello la clave pública y dirección de la entidad que las ha validado, así como del usuario propietario de ellas, quedan visibles para todo el que quiera consultarlo.

Para evitar abusos en el registro de la blockchain, la Unión Europea cuenta con una infraestructura blockchain privada y permisionada, donde cada país miembro puede tener nodos. Esta es la *European Blockchain Services Infrastructure* o EBSI. Esta blockchain está especialmente diseñada para cumplir con todas las regulaciones y valores de la Unión Europea. [19]

Por otro lado, las Presentaciones Verificables consisten en un conjunto de Credenciales Verificables, firmadas por el propio usuario. Estas se presentan a aquellos proveedores de servicios, los cuales verifican tanto la firma del usuario como la validez de las Credenciales Verificables presentadas.

Las tres entidades que componen el sistema son las siguientes:

- **Holder.** Este tipo de entidades son los usuarios del sistema. Solicitan la creación de Credenciales Verificables a un *Issuer* y las presentan dentro de una Presentación Verificable a los *Service Provider*

o *Verifiers*. Todos los ciudadanos de la UE serían Holders en el sistema y contarían con una identidad digital propia.

- **Issuer.** Este tipo de entidad está capacitada para emitir y firmar Credenciales Verificables de un nivel igual o inferior de confianza (Level of Assurance). Su firma está validada, bien por eIDAS (cualificados), bien por el sistema interno (autoproclamados). Aquellas Credenciales Verificables con su firma o certificado digital poseen validez ante un *Service Provider*. En el caso de Credenciales Verificables sobre el estado COVID estos Issuers serían entidades gubernamentales sanitarias cualificadas por eIDAS.
- **Service Provider** o Verifier. Son aquellas entidades que admiten Credenciales Verificables contenidas en Presentaciones Verificables. Se encargan de comprobar la validez de las mismas cerciorándose de que la firma adjunta sea la de un Issuer válido. Ofrecen servicios a los Holders, los cuales se identifican con las Credenciales Verificables mínimamente necesarias. Un Service Provider que pudiera solicitar las Credenciales sobre el estado COVID de un ciudadano puede ser, por ejemplo, una agencia de viajes en el momento de compra de un billete.

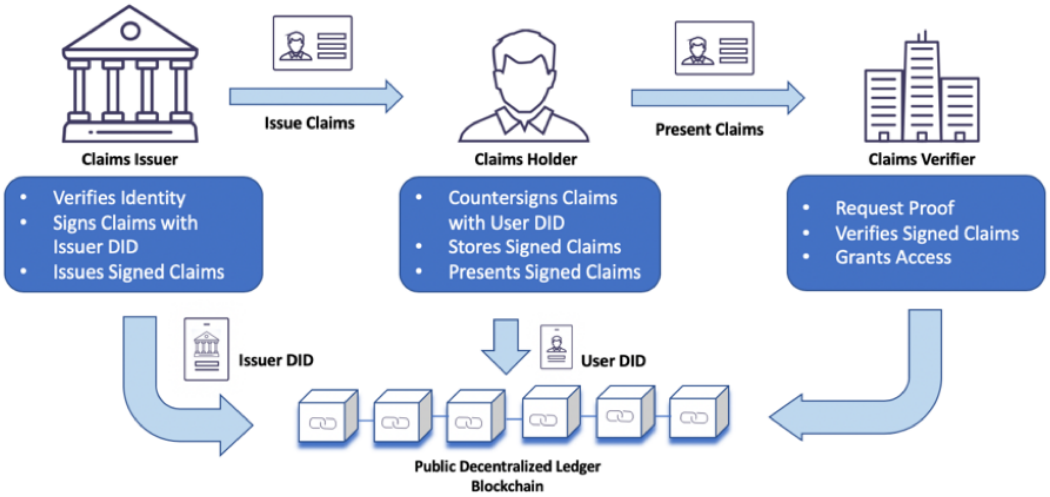


Figura 46: Diagrama del modelo de Self-Sovereign Identity.

6.1.7. Commission Nationale de l'Informatique et des Libertés

La *Commission Nationale de l'Informatique et des Libertés* o CNIL es una autoridad administrativa independiente, creada en 1978. Ejerce sus funciones acorde a la Ley de Protección de Datos de Francia.[15]

Debido a que Francia es país miembro de la Unión Europea, el 13 de febrero de 2018, la Asamble Nacional de Francia aprobó el Proyecto de Ley de Protección de Datos para transponer a derecho nacional el Reglamento General de Protección de Datos. Es por ello que la normativa de la CNIL va fuertemente ligada al RGPD en materias de privacidad de los datos. [6]

El análisis que se va a llevar a cabo a la aplicación desarrollada en este proyecto se hará con una herramienta de la CNIL.

La CNIL presenta una serie de recomendaciones y obligaciones en torno a las aplicaciones móviles, así como conexiones Bluetooth, características que posee esta aplicación.

Son las siguientes.[14]

Respecto a las aplicaciones móviles de salud

La CNIL, en su artículo, a la hora de evaluar cómo debiera afectar el RGPD al desarrollo de aplicaciones móviles de salud, primeramente se plantea si por definición entran en el ámbito de aplicación de la normativa sobre protección de datos personales.

Este planteamiento surge porque, en el caso de que la aplicación móvil de salud registre y almacene datos personales para uso exclusivamente local, sin comunicaciones con el exterior, y con un fin solamente personal, la legislación no aplicaría.
Aunque esto sea así, el proveedor ha de garantizar al usuario que se cumplen las medidas mínimas de seguridad, para evitar riesgos relacionados con la invasión de la privacidad.

Es por ello que la normativa se aplica a toda aquella aplicación cuyos datos salgan al exterior, bien sea para prestar servicios remotos o bien porque posea conexiones de cualquier tipo con el exterior.

Es por ello que la CNIL para estos casos dictamina que las aplicaciones móviles de salud con dichas características han de cumplir los puntos marcados por el Reglamento General de Protección de Datos.

Esto incluye, entre otros, el análisis de impacto en la privacidad.

La CNIL propone diversos fines que una aplicación móvil de salud puede tener que necesitan cumplir el RGPD. La aplicación aquí desarrollada posee los siguientes.

- **Ayudar al usuario en el seguimiento de su propia salud.** Pues la aplicación le informa sobre su estado de contagio, así como de posibles contactos contagiados.
- **Tener mejor control sobre la propia salud.** Debido a los consejos que aparecen en pantalla para evitar entrar en contacto con el virus, tener cuidado en cuarentena o en caso de contacto contagiado; la aplicación podría cumplir en parte con este fin.

Cabe destacar que, si bien la aplicación aquí desarrollada no comparte los datos entre diversas autoridades sanitarias, algunas aplicaciones de rastreo de contactos son interoperables. Este es otro caso que la CNIL considera.

Lo primeramente remarcado, es que los fines para los que los datos van a procesarse deben ser legítimos, explícitos y específicos. No pueden añadirse fines a posteriori, menos si son incompatibles con los anteriores.

Estos propósitos deben hacerse antes del diseño de la aplicación, para evaluar qué datos son realmente necesarios. En el caso de la aplicación del proyecto, como se especifica en los Requisitos de Información de la

fase de análisis, sería el identificador anónimo del usuario, su estado de contagio y qué contactos ha tenido y cuándo los ha tenido. Cara al servidor, este almacena identificadores contagiados y la fecha de su recepción.

Estos datos, al ser de salud, la CNIL indica que no solamente han de cumplir con el RGPD, sino que además deben permanecer también bajo aquellas disposiciones relativas a las condiciones para el intercambio y puesta a disposición de datos sanitarios (art. L. 1110-4 del CSP). [46]

Estas referencias son elaboradas por los representantes sanitarios mencionados en el art. L. 1111-24 del CSP. [47]

Posteriormente, son aprobados por orden del Ministro de Salud.

La CNIL también especifica que la asignación del rol de responsable del tratamiento de datos ha de evaluarse caso por caso. Esta entidad se encargará de determinar la naturaleza de la aplicación, sus funcionalidades y desarrollo.

La CNIL recalca que, además, los datos deben ser almacenados durante un periodo de tiempo limitado. Esto es, han de eliminarse llegado algún momento. La aplicación aquí desarrollada cumple esto al hacer uso de identificadores efímeros y borrado periódico de datos en el servidor; como se especificará en mayor profundidad en el análisis con la herramienta de la CNIL.

Es muy importante que los usuarios sepan en todo momento cuáles son sus derechos y cómo pueden ejercerlos. Además, es necesario su consentimiento, el cual ha de solicitarse después de haber informado sobre qué datos se van a recolectar y cómo se van a tratar.

Finalmente, la CNIL indica que un proceso obligatorio en caso de tener que tratar datos de los usuarios obtenidos mediante aplicaciones móviles de salud, es el análisis de impacto en riesgos de privacidad.

Además hace falta tomar una serie de medidas de seguridad. El encargado de procesar los datos, acorde al RGPD, tendrá estas obligaciones específicas, y deberá asegurar que los datos de los usuarios estén bajo buenas medidas de seguridad.

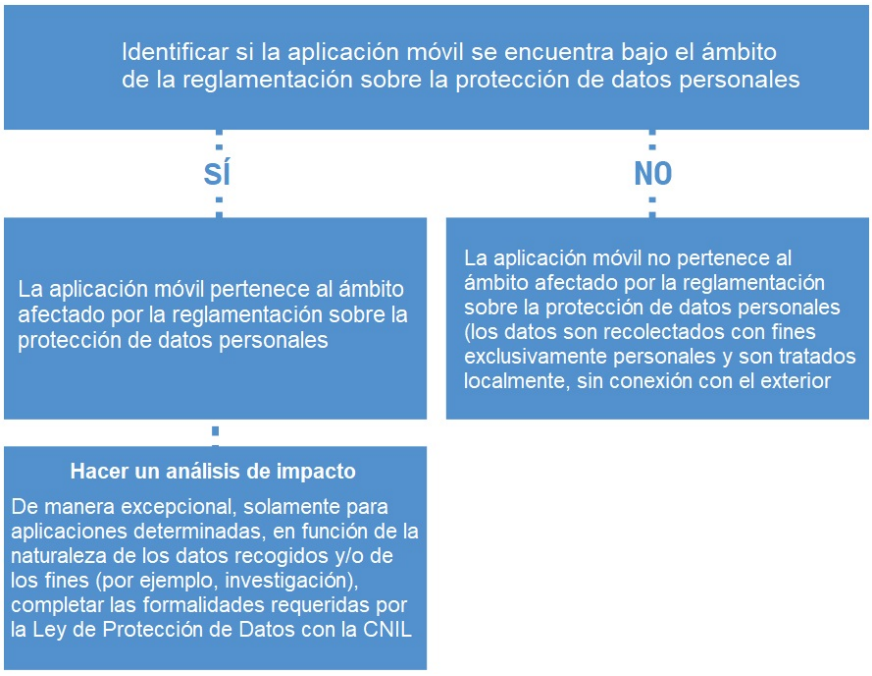


Figura 47: Esquema de la CNIL.

Respecto a las conexiones Bluetooth

Si bien la CNIL no parece tener artículos que hablen sobre la tecnología Bluetooth, del RGPD sí que pueden deducirse algunas consideraciones. [38]

Esto se debe al uso de balizas, las cuales pueden producir inconveniencias a la hora de cumplimentar el RGPD. Estas balizas o *beacons* son un tipo de dispositivo que puede colocarse en lugares, exteriores o interiores, con el fin de detectar individuos en un radio de 10 a 30 metros.

Estas balizas funcionan emitiendo una señal vía Bluetooth Low Energy a todos los dispositivos que se encuentren en su alcance. Cuando un dispositivo recibe esa señal, con ella obtiene un identificador único universal. Este, dentro del dispositivo permite determinar la ubicación del mismo, rastrearlo o activar ciertas funciones dependiendo de su localización.

Estas balizas solo son útiles si los dispositivos mantienen sus comunicaciones Bluetooth activas y poseen aplicaciones que puedan interpretar estas señales.

El hecho de que las aplicaciones de rastreo de contacto empleen tecnología Bluetooth y estén diseñadas tal que deban estar trabajando en segundo plano todo el rato, implica que los usuarios han de mantener las conexiones Bluetooth activas en todo momento.

Esto implica que un usuario pudiera tener que elegir entre el funcionamiento correcto de la aplicación de rastreo de contactos y su privacidad ante ciertas balizas.

Por lo tanto, aquellas aplicaciones que se aprovechen de balizas deben recolectar y tratar los datos del mismo modo que cualquier otro tipo de aplicación.

La ley especifica que se ha de informar al usuario sobre qué datos se están recolectando y con qué fines. También por cuánto tiempo esos datos se mantienen. En definitiva, es necesaria una política de privacidad.

Sin embargo una cuestión crucial surge en este tipo de aplicaciones, y esta es, cómo se obtiene el consentimiento de los usuarios para obtener sus datos por tráfico generado por balizas. Sobre todo, datos relativos a su geolocalización. Lo más habitual es que el usuario acepte estas condiciones al momento de hacer uso de alguna aplicación que se comunique con las balizas. Estas aplicaciones informan al usuario de que recopilarán datos en todo momento, a no ser que este indique lo contrario de manera activa. Permiten, para los casos de geolocalización, que los usuarios rechacen de entrada su constante recopilación.

Sin embargo, el hecho de ser una función automatizada, conectándose a balizas que pueden estar repartidas por cualquier localización, genera el dilema de que, activando Bluetooth para otras tareas, inevitablemente se comienza a aceptar el tráfico procedente de balizas.

6.1.8. Herramienta PIA

El nombre de la herramienta, PIA, proviene de *Privacy Impact Assessment*, es decir, Evaluación del Impacto de la Privacidad.[4]

Llevar a cabo un PIA consiste en realizar una descripción de aquellas operaciones para el tratamiento de los datos previstas, así como fines e interés legítimo. Es necesario justificar la necesidad y proporcionalidad de las operaciones llevadas a cabo para ello, siendo conformes con la finalidad especificada. [70]

También hace falta realizar una evaluación de los riesgos que puedan afectar a los derechos y/o libertades de los interesados, así como las medidas previstas para afrontar dichos riesgos. Estas medidas incluyen garantías, medidas de seguridad y mecanismos que garanticen la protección de los datos personales tratados. [70]

Con dicho PIA se busca demostrar la conformidad del proyecto con el Reglamento General de Protección de Datos.

La herramienta PIA es un software de código abierto que ayuda a realizar análisis de impacto en la privacidad de los datos. De este modo, ayuda a los controladores de datos a demostrar el cumplimiento del Reglamento General de Protección de Datos.

La herramienta está disponible oficialmente en francés y en inglés. Además, PIA también pretende facilitar el uso de las guías publicadas por la CNIL para *Privacy Impact Assessment* (PIA).[16]

Esta herramienta está dirigida a los controladores de datos, para ayudarles en el proceso de evaluación y análisis de impacto en privacidad. La aplicación puede descargarse para ordenadores Windows, Mac OS y Linux. También existe una versión web que puede ser desplegada en los servidores de cualquier organización. Para ello, la página proporciona las descargas del *front-end* y del *back-end*. De este modo es posible integrarla con otras herramientas o sistemas que se usen internamente en dicha organización.

PIA está diseñada con tres principios en mente.

- **Una interfaz didáctica para poder llevar a cabo análisis**. La herramienta posee una interfaz fácil de usar para permitir al usuario un manejo sencillo de sus evaluaciones. Divide el proceso de análisis paso por paso. Además, para facilitar la comprensión de los riesgos, incorpora diversas herramientas de visualización.
- **Posee una base de conocimientos legal y técnica.** La herramienta acompaña el proceso del análisis con varios puntos legales. De este modo asegura la licitud del tratamiento y los derechos de los usuarios. Este contenido se va adaptando según la información introducida en la herramienta. Todos los datos son extraídos del Reglamento General de Protección de Datos, las guías para evaluación de impacto en la privacidad de la CNIL y las guías de seguridad de la CNIL aplicables a los aspectos tratados.
- **Una herramienta modular.** Con ello permite la personalización de su contenido, especificando necesidades o sector de negocio. Al ser una herramienta de código abierto, además, permite añadir características o implementarla dentro de otras herramientas.

La herramienta PIA permite además exportar un informe con toda la evaluación realizada una vez se ha terminado el análisis. Este informe se presenta en formato *PDF*.

Esta también posee ciertas medidas de control del avance del usuario. Para poder redactar secciones que refieren a pasos posteriores al en el que se está trabajando en el momento, PIA exige que todos los campos se hayan rellenado. De este modo, evita que se salten pasos y así se genere un informe completo. Del mismo modo, al editar un riesgo, es obligatorio introducir una medida que lo aborde.

[4]De ahora en adelante se empleará «PIA» para referirse a Evaluación del Impacto de la Privacidad por ser las siglas comúnmente utilizadas en el ámbito tecnológico.

Para llevar a cabo una mejor evaluación se realizará el análisis teniendo en cuenta también las guías de la CNIL para PIA. Estas incluyen metodología a seguir, plantillas y bases de conocimiento.[16]

La aplicación PIA puede descargarse en el siguiente enlace de la página de la CNIL: https://www.cnil.fr/en/open-source-pia-software-helps-carry-out-data-protection-impact-assesment.

El propósito de realizar un análisis con esta aplicación es demostrar que esta cumple con la normativa del Reglamento General de Protección de Datos, así como que los controladores y procesadores de los datos cumplen las normativas estipuladas.

6.2. PIA: Evaluación de Impacto de la Privacidad

Se describe el proceso realizado con la herramienta **PIA**, de la autoridad francesa para la protección de datos; la **Commission Nationale de l'Informatique et des Libertés** o **CNIL**.

Esta herramienta se ha utilizado para llevar a cabo el análisis de privacidad de la aplicación. También se presentan aquellas conclusiones obtenidas de su uso, así como las salvaguardas a implementar para cubrir los riesgos encontrados.

Los apartados aquí presentados son los mismos que se siguen durante una evaluación de PIA, para así facilitar el seguimiento.

La versión utilizada es la 2.2.0 para escritorio de Windows-64 bits.

A la hora de emplear la herramienta, se consideran cuatro fases, las cuales se repiten de forma iterativa cada vez que se implementa o diseña un procesamiento de datos. Estas fases son **Contexto**, **Principios Fundamentales**, **Riesgos** y **Validación**.

6.2.1. Creación del proyecto

Para comenzar a realizar el análisis de la aplicación, primero se crea un nuevo proyecto mediante la opción *Nuevo PIA*.

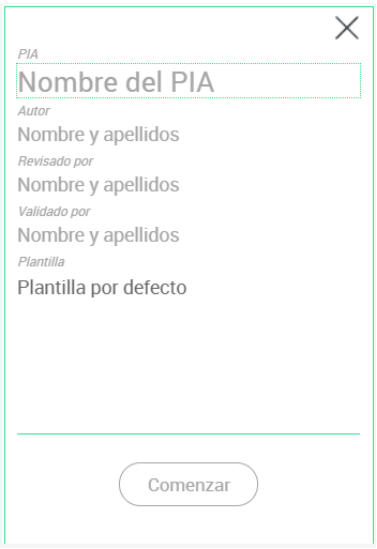


Figura 48: Creación de nuevo proyecto PIA.

Se rellenan los datos para la creación del proyecto. Se solicita un nombre para identificar el PIA, así como el autor, quién lo revisa y quién lo valida. El único tipo de plantilla que permite seleccionar al inicio es la de por defecto.

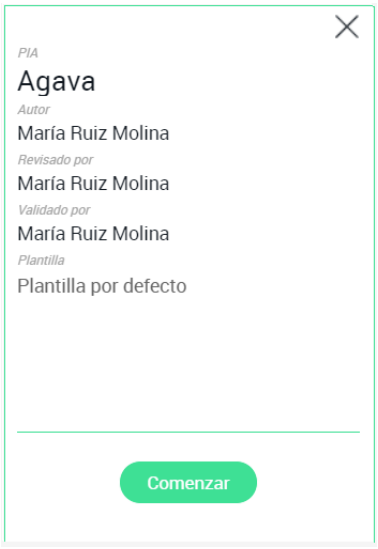


Figura 49: Creación de nuevo proyecto PIA con datos.

Una vez se han completado estos datos y pulsado *Comenzar*, aparece la pantalla principal de PIA para poder empezar el análisis.

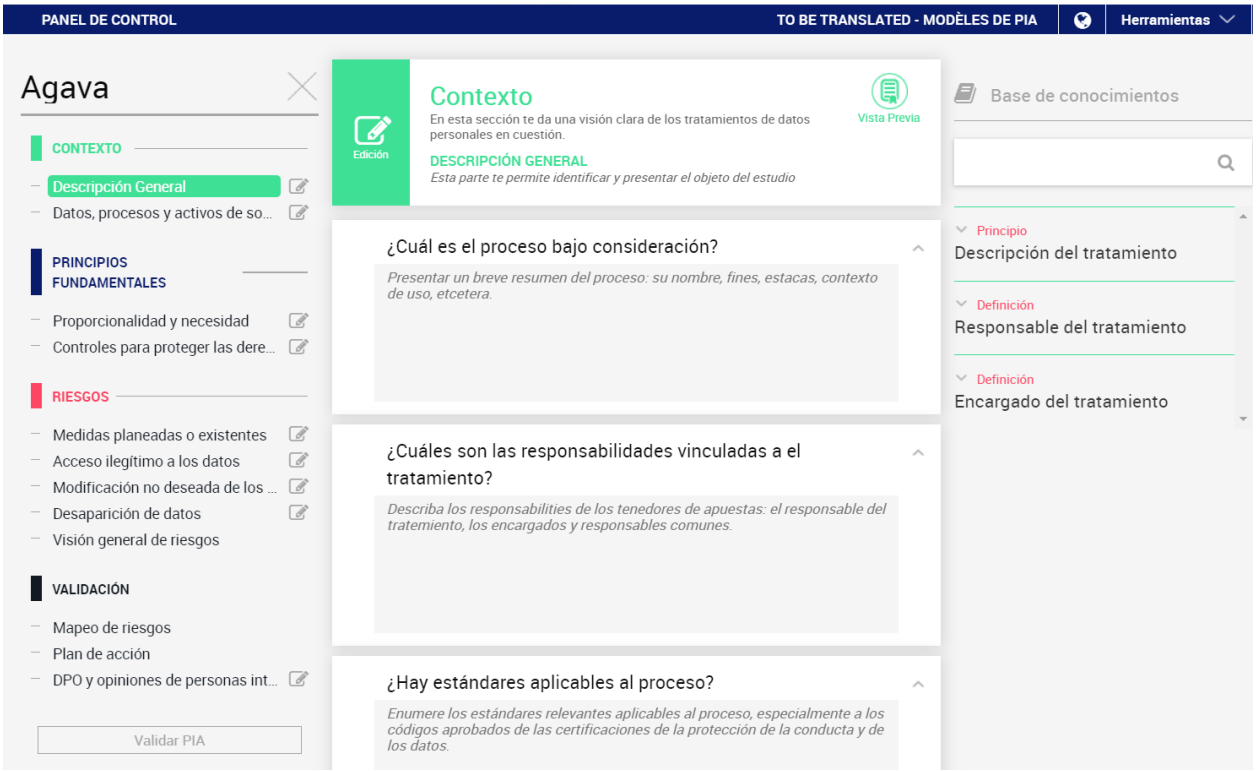


Figura 50: Interfaz PIA.

Dentro de esta interfaz hay tres zonas a destacar.

La primera es la zona de la izquierda con los pasos que se irán tomando a lo largo del desarrollo del análisis. Por el momento solo se puede avanzar a la edición de uno de los pasos siguientes al actual (Contexto) una vez se hayan completado todos los apartados del mismo.

Figura 51: Pasos para el análisis PIA.

En la zona de la derecha aparece un menú informativo. Aquí aparecen aquellas regulaciones, observaciones, definiciones... Que el RGPD y la CNIL dictaminan con respecto a lo tratado en la fase de la pantalla en la que se encuentre en ese momento la aplicación.

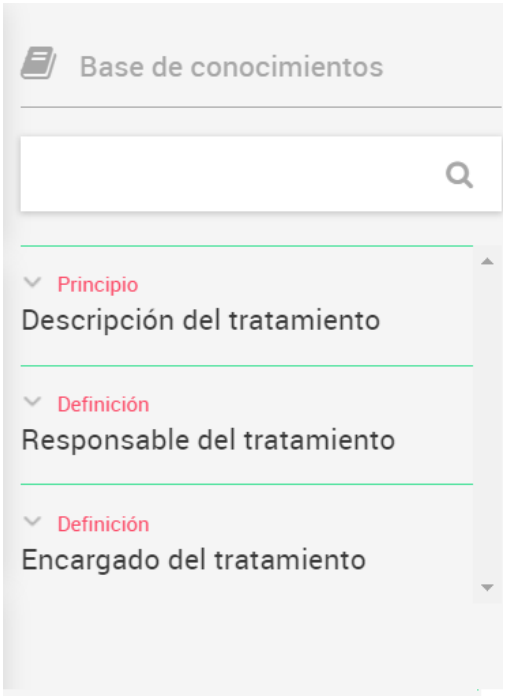


Figura 52: Recomendaciones para el análisis PIA.

Finalmente, está la zona central. Dentro de esta aparecen los pasos a dar dentro de la fase del análisis en la que nos encontremos. Se deben ir rellenando en orden, siendo algunos campos obligatorios, pues serán necesarios para fases posteriores del análisis.

Estas pantallas se irán mostrando en los siguientes apartados, cada una en su fase correspondiente.[17]

6.2.2. Contexto

La primera fase a llevar a cabo con la aplicación PIA es el *Contexto*.[17] En esta fase se describe el contexto bajo el que se procesarán los datos recopilados en cuestión.

Se busca ganar una visión clara de las operaciones que llevan a cabo el procesamiento de los datos.

Aquí se describe la naturaleza de los datos, su alcance, el contexto, propósitos y fines para los cuales son recopilados y por cuánto tiempo se almacenan.

También se identifica al controlador de los datos y a los encargados del procesamiento.

Otro paso es señalar aquellas referencias aplicables al procesamiento de los datos, las cuales deben ser cumplidas en todo momento. También se referencian aquellos certificados referentes a la protección de datos con los que el proyecto cuenta (Art. 42 del RGPD) y códigos de conducta aprobados (Art. 40 del RGPD).[73]

La aplicación PIA divide este paso en dos, *Overview* y *Data, Processes and Supporting Assets*.

La primera permite presentar el objeto de estudio, la finalidad del proyecto.

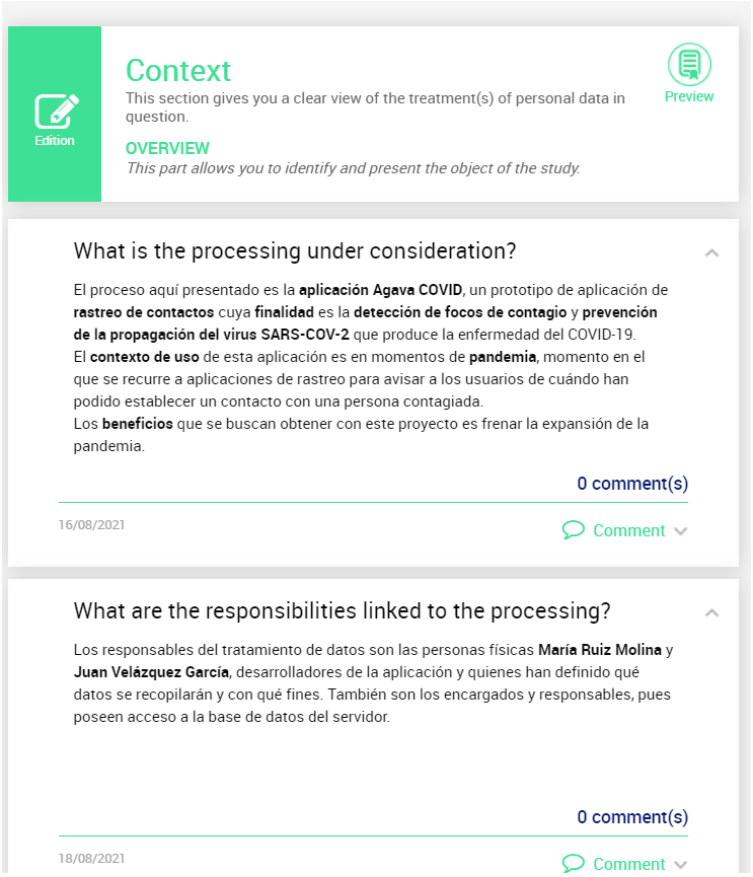


Figura 53: Pantalla Overview de Context en PIA.

La segunda, definir el alcance del procesamiento de datos al detalle.

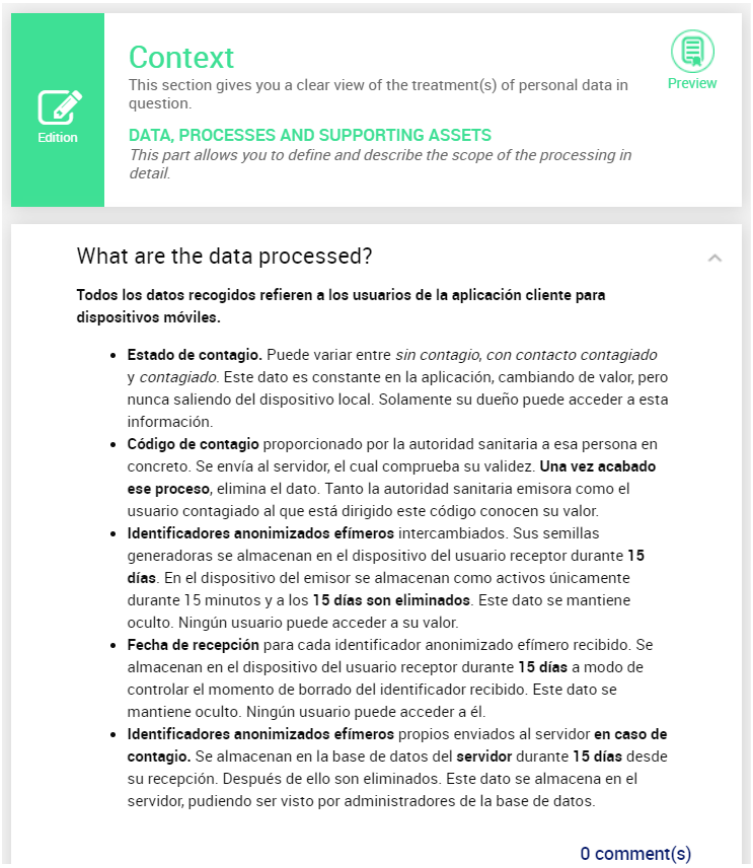


Figura 54: Pantalla Data, Processes and Supporting Assets de Context en PIA.

Más adelante puede verse todo este subapartado al completo, en el análisis completo de PIA al final de este apartado.

6.2.3. Principios Fundamentales

El objetivo de este punto del análisis es dar forma al sistema que asegura el cumplimiento de los principios de privacidad y protección de los datos.[17]

Primeramente se realiza una evaluación sobre aquellos controles que garantizan la proporcionalidad y necesidad del tratamiento de los datos. Esto significa justificar aquellas decisiones tomadas para tratar los siguientes puntos.

- **Propósito.** Es propósito para el cual los datos son recopilados debe ser legítimo, explícito y específico. (Art. 5.1 (b) del RGPD). [73]
- **Fundamentos.** Cuál es la legalidad del procesamiento y especificar la prohibición de un uso indebido de los datos y sus finalidades. (Art. 6 del RGPD).[73]
- **Minimización de los datos.** Los datos recopilados deben ser relevantes, limitados y adecuados al fin para el que se están recopilando. (Art. 5.1 (c) del RGPD).[73]
- **Calidad de los datos.** Los datos recopilados deben ser precisos, ciertos y estar actualizados. (Art. 5.1 (d) del RGPD).[73]
- **Periodos de almacenamiento.** Estos han de ser limitados en el tiempo, ningún dato debe mantenerse almacenado de forma indefinida. (Art. 5.1 (e) del RGPD).[73]

También ha de especificarse que el cumplimiento con el RGPD no puede mejorarse más, y de este modo demostrando que la aplicación hace todo lo posible por el cumplimiento de la legislación. De ser necesario, estos apartados deben revisarse y añadir aquellos controles que puedan irse encontrando.

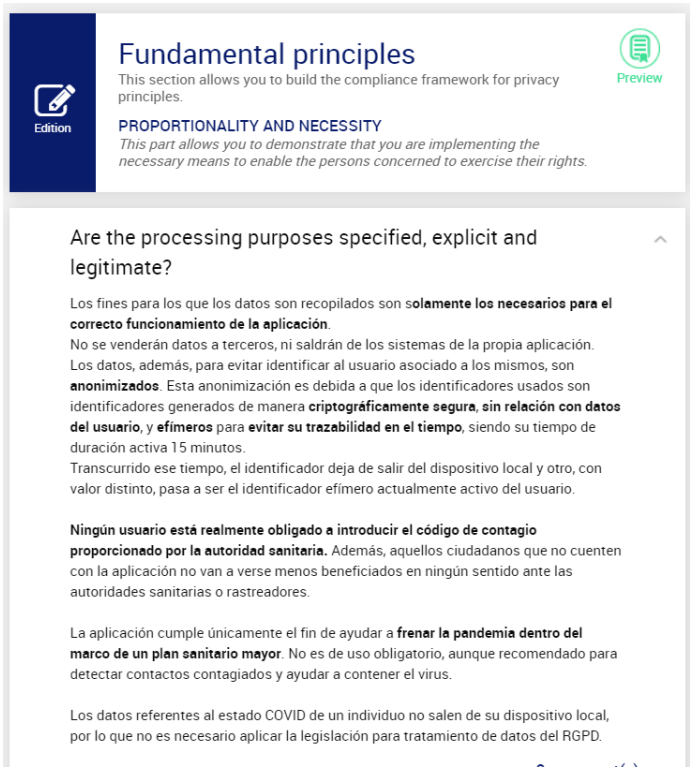


Figura 55: Pantalla Proportionality and Necessity de Fundamental Principles en PIA.

Una vez definidos los principios por los cuales el tratamiento es llevado a cabo, el siguiente paso es especificar los controles llevados a cabo para asegurar la protección de los datos de los usuarios. También se describen aquellos que aún no se hayan implementado pero se planee hacer y cómo se harán.

Por lo tanto, la información aquí contenida abarca los siguientes puntos.

- **Información para los sujetos de los datos.** Esta debe escribirse de forma justa, clara y transparente. (Art. 12, 13, 14 del RGPD). [73]
- **Obtener el consentimiento de los usuarios para el procesamiento de los datos.** Se ha de poder demostrar que efectivamente se ha obtenido su consentimiento. (Art. 7, 8 del RGPD).[73]
- **Cómo pueden ejercer los usuarios su derecho a la portabilidad de los datos.** (Art. 15, 20 del RGPD).[73]
- **Cómo pueden ejercer los usuarios su derecho al borrado y/o rectificación de los datos.** (Art. 16, 17 del RGPD).[73]
- **Cómo pueden ejercer los usuarios su derecho a la restricción del procesamiento de sus datos y/o a su oposición.** (Art. 18, 21 del RGPD).[73]
- **Se ha de identificar a los procesadores de los datos.** Estos deben estar identificados y gobernados bajo las condiciones de un contrato. (Art. 28 del RGPD).[73]
- **Se ha de especificar cómo es la transferencia de datos fuera de la Unión Europea.** (Art. 44 a 49 del RGPD).[73]

De nuevo, se debe demostrar que cada control establecido es inmejorable y que con ello mejorar el cumplimiento del RGPD no es posible. Cuando sea aplicable, se deberán revisar estas descripciones y, de ser necesario, proponer controles adicionales.

En PIA la pantalla referente a esta parte es la siguiente.

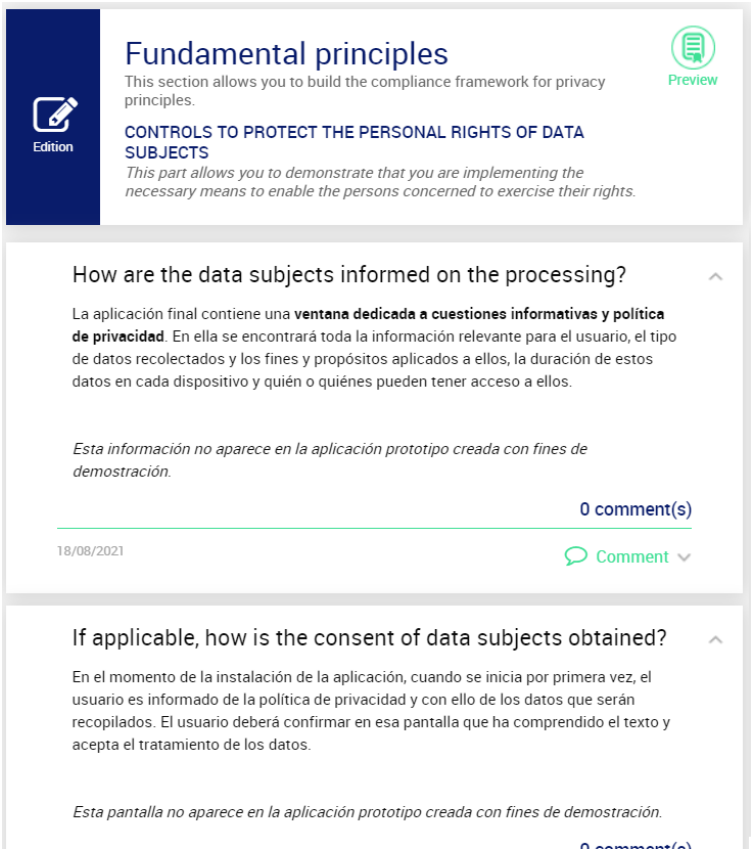


Figura 56: Pantalla Controls to Protect the Personal Rights of Data Subjects de Fundamental Principles en PIA.

6.2.4. Riesgos

Un riesgo, según la definición de las guías de la CNIL para PIA, es un escenario hipotético que describe un evento temido y todas las amenazas que pueden provocarlo.[17]

Esto incluye las posibles fuentes que desencadenen ese riesgo, las vulnerabilidades que pueden ser explotadas, el contexto que lleva a dichas amenazas y permite que sucedan, afectando a los datos personales de los usuarios almacenados y provocando así impactos negativos en su privacidad.

El **nivel del riesgo** se obtiene de considerar la **probabilidad** de que este suceda y la **severidad** del mismo.

La probabilidad de que un riesgo se materialice depende de las vulnerabilidades del proyecto, así como de las capacidades de la fuente que puede producir ese riesgo.

La severidad depende de la naturaleza perjudicial y el impacto negativo que puede tener ese riesgo en caso de materializarse. A mayor sea el daño causado, más severo se considera el riesgo.

Este apartado consta de dos fases.

Evaluación de controles existentes o planificados

El objetivo en este apartado es definir los controles establecidos que contribuyen a la seguridad del sistema.

Se busca identificar o determinar la existencia de planes de contingencia ya existentes. Estos pueden englobarse en tres formas diferentes.

1. **Controles relacionados directamente con los datos que se procesan.** Esto incluye cifrado, anonimización, descentralización, control de acceso, trazabilidad...
2. **Controles de seguridad general aplicados al sistema que realiza el tratamiento de datos.** Esto incluye copias de seguridad, seguridad del hardware...
3. **Controles organizacionales o gobernanza.** Política, gestión del proyecto, personal encargado, gestión de incidentes o brechas de seguridad y datos, relaciones con terceros...

Se ha de comprobar que cada control aplicado debe ser máximo y no puede mejorarse. De ser necesario, se revisarán las descripciones y se propondrán medidas adicionales.

Figura 57: Pantalla Planned or Existing Measures de Risks en PIA.

Evaluación de riesgos: Posibles violaciones de la privacidad

El objetivo aquí es conocer y comprender bien las causas y consecuencias de los riesgos.

Los riesgos referentes a los datos que se consideran en PIA son los siguientes:

- **Acceso ilegítimo a los datos.**
- **Modificación de los datos.**
- **Borrado de los datos.**

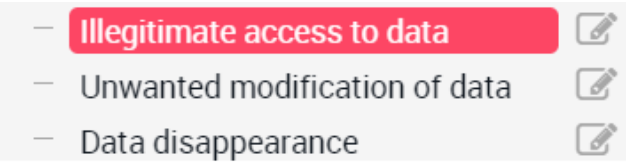


Figura 58: Tipos de riesgos en PIA.

Tras definir los riesgos para cada categoría, hay que determinar el impacto potencial en la privacidad de los datos de los usuarios en caso de ocurrir. Tras ello, se tendría que estimar su severidad, sobre todo teniendo

en cuenta el impacto potencial y, si es aplicable, qué salvaguardas disponer para controlarlos.

También se han de identificar las amenazas posibles que pueden llevar a que este riesgo se materialice.

Una vez se dispone de toda esta información se calcula la probabilidad de que ocurra, teniendo en cuenta las vulnerabilidades existentes en el proyecto y las capacidades de las amenazas para explotarlas.

En PIA se encuentra una pantalla distinta para cada tipo de riesgo, todas con apartados donde se etiqueta cada característica.

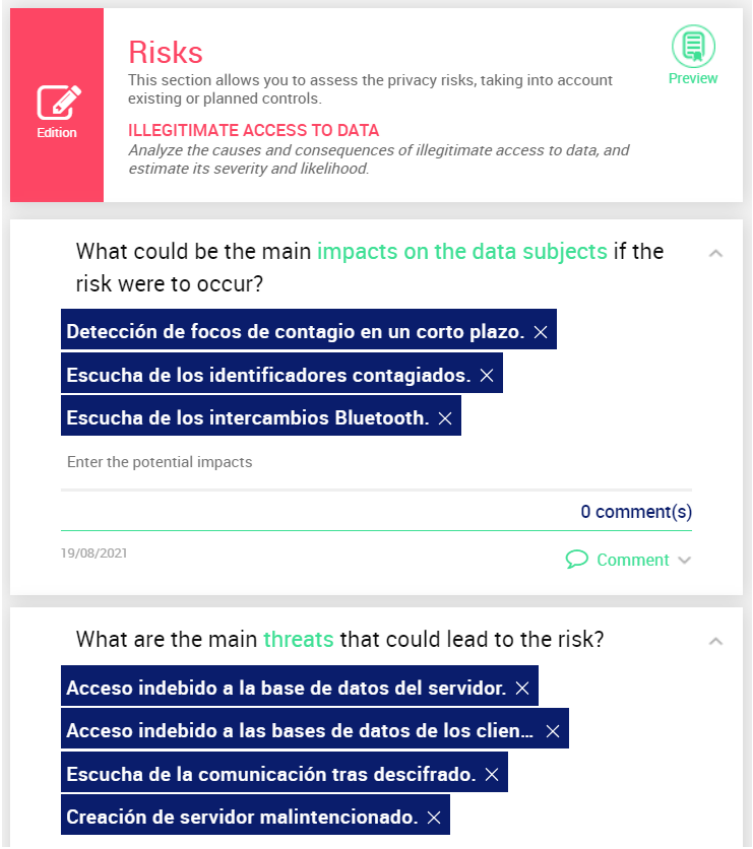


Figura 59: Pantalla ejemplo de Risks en PIA.

Se ha de considerar si los riesgos son asumibles o no tras realizar este apartado. De no serlo, se han de especificar las salvaguardas que habría que incorporar y se ha de revisar este apartado actualizando los riesgos para que sean residuales.

PIA genera una pantalla para visualizar los riesgos, amenazas, fuentes y medidas, y las relaciones que poseen con cada uno de los posibles incidentes.

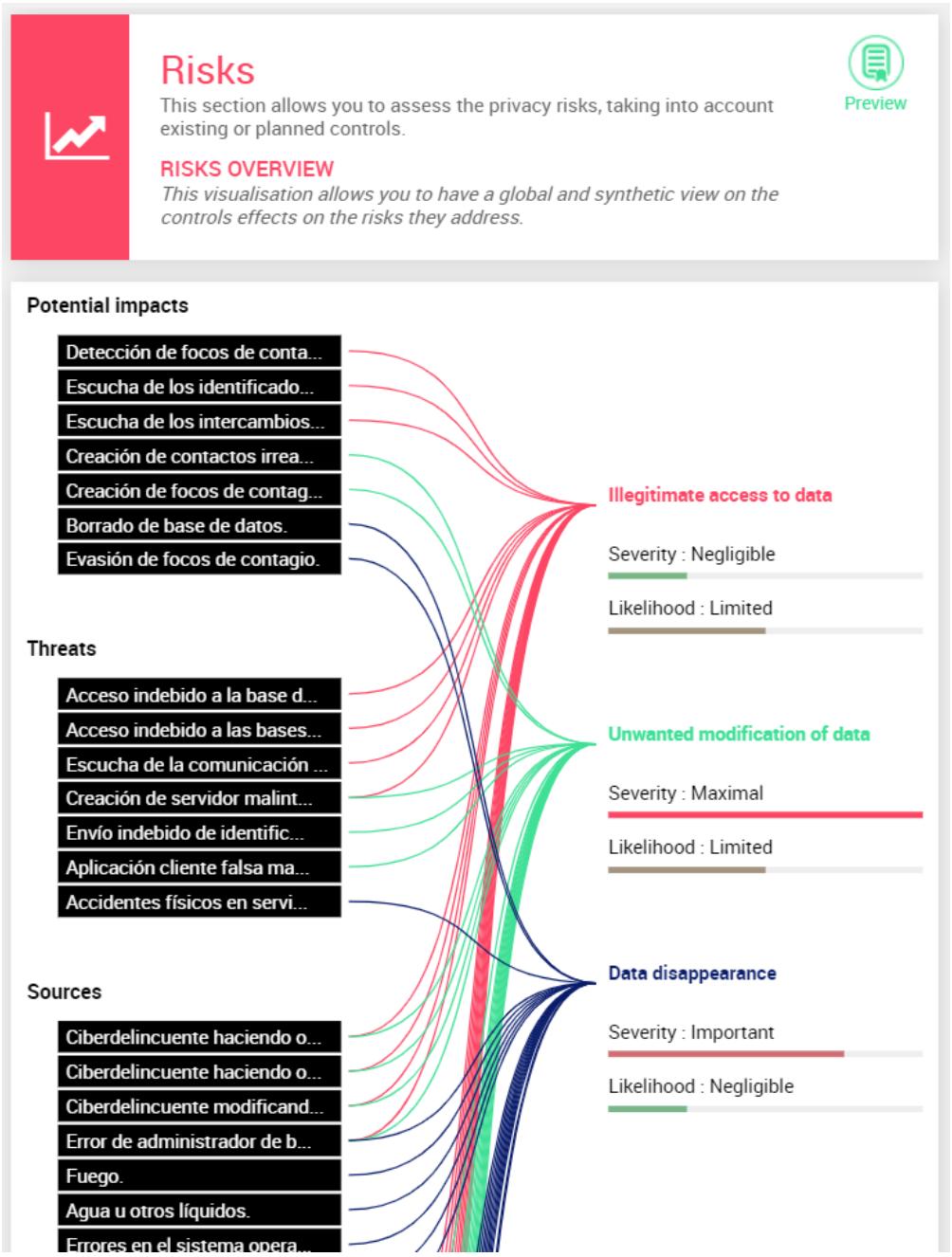


Figura 60: Pantalla Risks Overview de Risks en PIA.

6.2.5. Validación

En este paso el objetivo es validar o no el análisis realizado. De este modo, se da o no el visto bueno al análisis. En caso de que se encuentre algún aspecto a mejorar se realizaría una nueva iteración del análisis.[17]

Primeramente se ha de preparar el material necesario para la validación.

- Una presentación visual de los controles seleccionados para asegurar el cumplimiento del reglamento.
- Una presentación visual de los controles seleccionados para contribuir a la seguridad de los datos.
- Mapa de riesgos visual, inicial y residual, dependiendo de su severidad y probabilidad.
- Plan de acción para los controles adicionales necesarios detectados en las etapas anteriores, así como su responsable de la implementación, el coste y tiempo que llevará realizarlo.

Figura 61: Imagen visual generada en PIA.

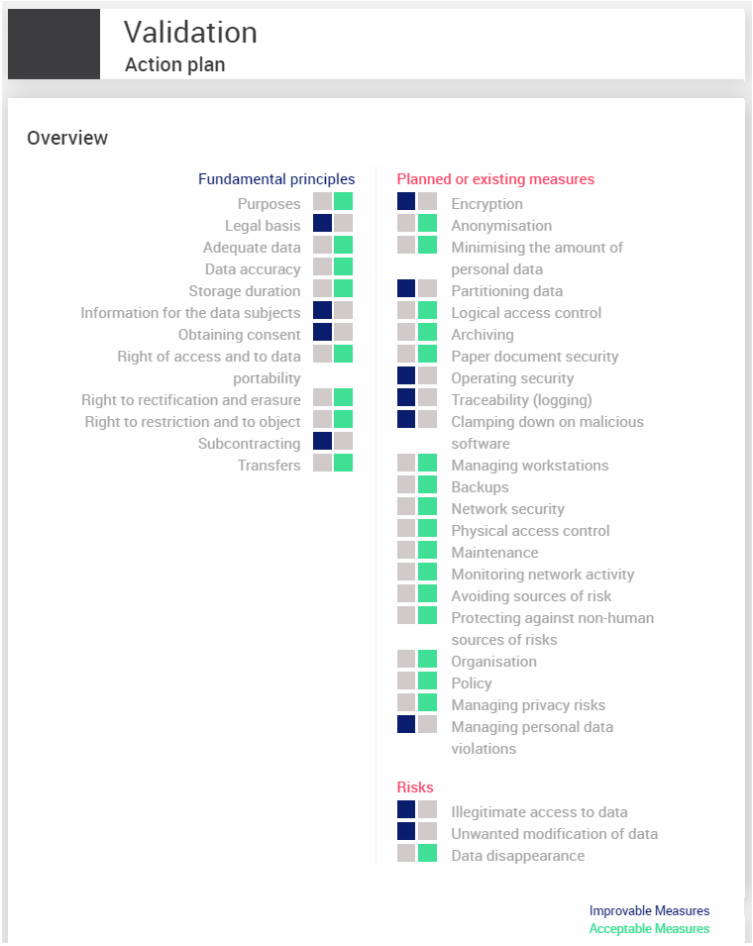


Figura 62: Plan de riesgos generado en PIA.

Todo ello requiere del consejo del responsable de los aspectos de protección de datos, así como de los interesados de los datos o de sus representantes.

Tras ello se lleva a cabo la validación formal. Se decide qué controles, riesgos residuales y planes de acción son aceptables de manera justificada. De este modo el análisis PIA queda en uno de los siguientes tres estados.

1. **Validado.**
2. **Mejorable.**
3. **Rechazado.**

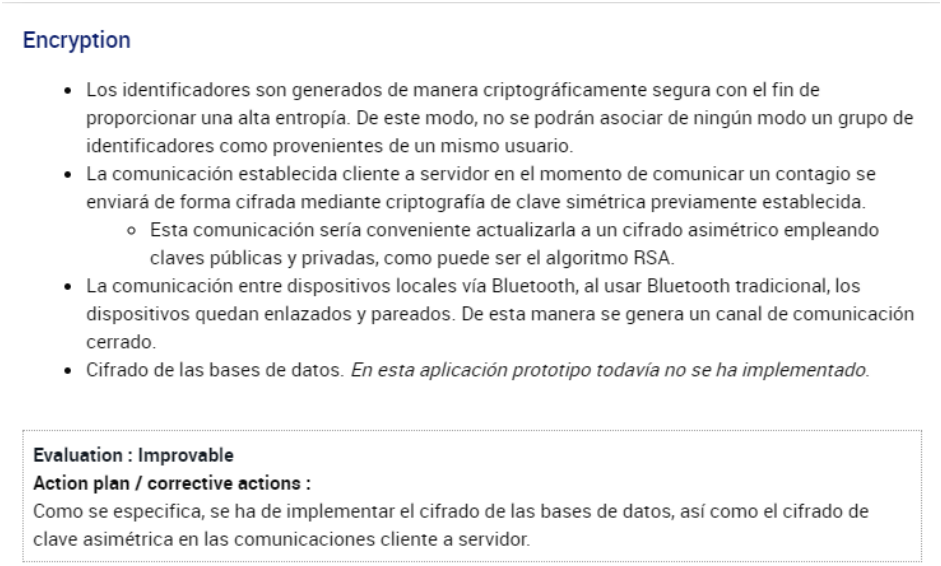


Figura 63: Pantalla de comentario en Validación en PIA.

En caso de que el resultado no sea **Validado**, se repite otra iteración de los pasos hasta que sea así.

6.3. PIA: Proceso de la Evaluación

El proceso de evaluación se ha realizado siguiendo los pasos anteriormente descritos. Se muestra un resumen del mismo a continuación y tras ello, el resultado al detalle obtenido de la herramienta PIA.

6.3.1. Contexto

Agava COVID es una aplicación de rastreo de contactos cuya finalidad es detectar focos de contagio para prevenir la propagación del virus SARS-COV-2. El contexto de su uso es durante la pandemia de COVID-19.

Los responsables del tratamiento son María Ruiz Molina y Juan Velázquez García. Son los únicos individuos con acceso a la base de datos, y se han de asegurar de que todos los datos procesados por la aplicación estén seguros. Para ello se aplican las recomendaciones de la EDPB, la legislación del RGPD, las guías de la CNIL y se aplica el protocolo DP-3T.

Los datos procesados serán el **estado de contagio**, el **código de contagio**, los **identificadores anonimizados efímeros obtenidos**, la **fecha de recepción** de estos y los **identificadores anonimizados efímeros propios**. Todos estos datos son eliminados en el momento en el cual dejan de ser necesarios para el funcionamiento correcto de la aplicación.

6.3.2. Principios Fundamentales

Los fines para los cuales se tratan los datos son exclusivamente los necesarios para el correcto funcionamiento de la aplicación. No salen en ningún momento de los sistemas de la propia aplicación. Todos ellos, además, se mantienen anonimizados.

Los usuarios en todo momento pueden restringir los datos que comparten con la aplicación, limitando las funcionalidades de la aplicación simplemente decidiendo no interactuar con ellas.

Se ha minimizado la cantidad de datos recolectados, haciendo uso de pseudónimos efímeros en forma de identificadores generados criptográficamente.

Todos los datos se mantienen actualizados, pues además sus periodos de vida son relativamente cortos. En el momento en el que un dato caduca este es eliminado.

Se especifica cómo los usuarios pueden ejercer sus derechos, como es dando su consentimientos a través de la aplicación (*función a implementar, pantalla con aceptación de política de privacidad*) o contactando con los responsables del tratamiento a través de un correo.

6.3.3. Riesgos

Se realiza una recopilación de todas las medidas existentes o planificadas. Estas pueden encontrarse en un mayor nivel de detalle más adelante en el documento obtenido de la herramienta PIA.

Una vez recopiladas todas las medidas se estiman los riesgos existentes.

- **Acceso ilegítimo a los datos.** Este riesgo se califica como de bajo impacto debido a la anonimización de los identificadores y al hecho de que estos sean efímeros. El hecho de visualizarlos no aporta información útil. La probabilidad de que ocurra, sin embargo, es algo mayor. Esto es debido a que el cifrado asimétrico de las comunicaciones, así como el cifrado de la base de datos, todavía no se han implementado.
- **Modificación no deseada de los datos.** La severidad de este riesgo es muy alta. Esto es debido a que la modificación de los datos contenidos en la base de datos podría generar focos de contagio

irreales, pudiendo en el peor caso hacer que todos los usuarios estuviesen supuestamente contagiados. La probabilidad de que esto ocurra, sin embargo, es limitada. Esto es debido a que el acceso a los datos se mantiene protegido por un usuario y contraseña únicos. Sin embargo no es muy bajo debido a que el cifrado de las bases de datos está por implementar todavía.

- **Desaparición de datos.** La severidad de este riesgo es bastante alta. Esto se debe a que podría eliminar focos de contagio todavía latentes. Esto podría ocasionar que el virus se propagase con mayor facilidad. Sin embargo, la probabilidad de que esto ocurra es muy baja. Esto es debido a la existencia de copias de seguridad. En caso de desaparición de los datos estos podrían recuperarse, y el riesgo solo podría materializarse si se lograse el borrado en todas las copias de seguridad de forma simultánea.

6.3.4. Validación

Si bien el PIA se valida debido a que la aplicación es un prototipo con fines investigativos, hay algunos apartados que debieran implementarse en caso de sacar la aplicación a un público real. Estos son los siguientes.

Es necesario implementar en la aplicación una pantalla de aceptación de la política de privacidad. De este modo se obtiene el consentimiento de los usuarios de manera activa.

Otro punto a mejorar refiere al cifrado de las comunicaciones, las cuales deberían implementar la criptografía de clave asimétrica.

También se han de cifrar las bases de datos con el fin de maximizar la protección de su contenido.

Otro punto a mejorar consiste en implementar un sistema de trazabilidad en los servidores, tal que se puedan comprobar los accesos realizados al mismo. De este modo se facilitaría la detección de una intrusión.

Además, sería conveniente realizar un escrito el cual firmasen los responsables del tratamiento de los datos. En dicho escrito se habrían de definir sus derechos y obligaciones, así como las de los interesados. Sería también apropiada la firma de un representante de estos últimos.

Tras implementar y aplicar todas estas salvaguardas, se realizaría otra iteración del PIA. En caso de no detectar nuevos inconvenientes mayores, este se validaría y cerraría.

A continuación se muestra el PIA realizado, con un mayor nivel de detalle, tal cual es generado por la herramienta.

Preview

GENERAL INFORMATION

100%

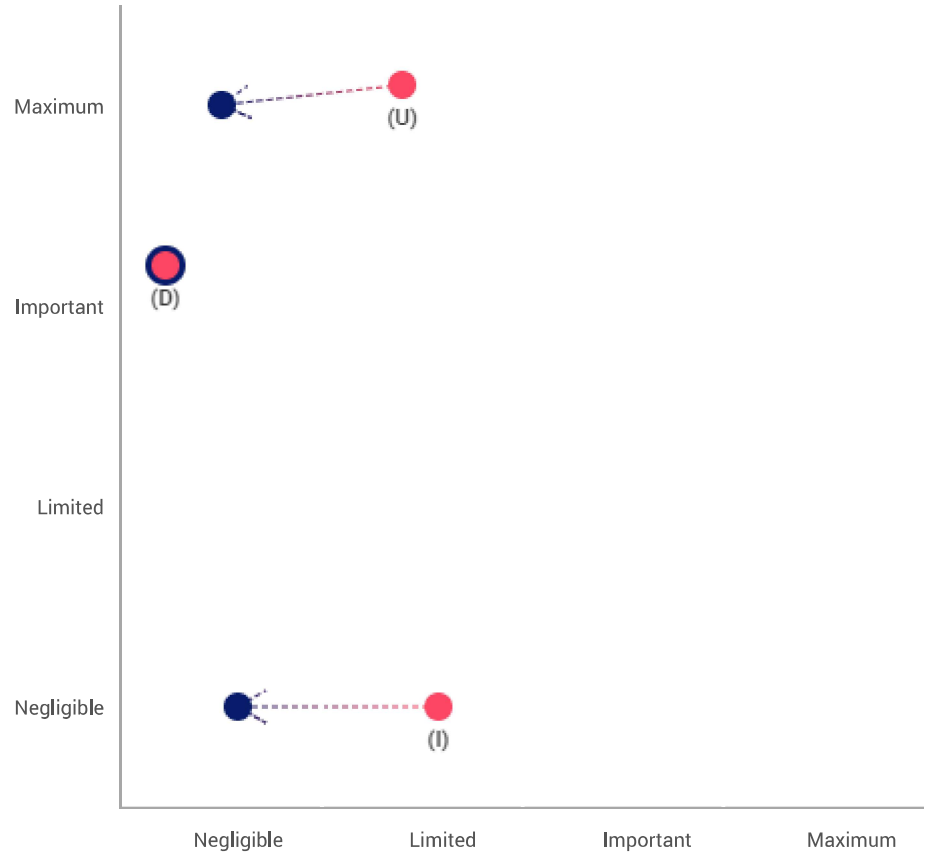
Preview

Editing :	María Ruiz Molina	**Status :**	Simple validation
Evaluation :	María Ruiz Molina		
Validation :	María Ruiz Molina		

Validation

Risk mapping

Risk seriousness



Risk likelihood

- Planned or existing measures
- With the corrective measures implemented
- (I)llegitimate access to data
- (U)nwanted modification of data
- (D)ata disappearance

16/08/2021

Validation

Action plan

Overview

Fundamental principles | Planned or existing measures

Fundamental principles

Purposes
Legal basis
Adequate data
Data accuracy
Storage duration
Information for the data subjects
Obtaining consent
Right of access and to data portability
Right to rectification and erasure
Right to restriction and to object
Subcontracting
Transfers

Planned or existing measures

Encryption
Anonymisation
Minimising the amount of personal data
Partitioning data
Logical access control
Archiving
Paper document security
Operating security
Traceability (logging)
Clamping down on malicious software
Managing workstations
Backups
Network security
Physical access control
Maintenance
Monitoring network activity
Avoiding sources of risk
Protecting against non-human sources of risks
Organisation
Policy
Managing privacy risks
Managing personal data violations

Risks

Illegitimate access to data
Unwanted modification of data
Data disappearance

Improvable Measures
Acceptable Measures

Fundamental principles

Legal basis

Action plan / corrective actions :
Convendría la implementación de una pantalla de aceptación de la política de la privacidad. Esta debiera aparecer la primera vez que el usuario inicia la aplicación.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Information for the data subjects

Action plan / corrective actions :
Como se especifica en la nota en cursiva, es necesaria la redacción e implementación de una política de privacidad en la aplicación.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Obtaining consent

Action plan / corrective actions :
Como se especifica en la nota en cursiva, es necesario implementar una pantalla donde el usuario pueda aceptar la política de privacidad la primera vez que inicie la aplicación.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Subcontracting

Action plan / corrective actions :
Sería necesaria la redacción y firmado de un contrato. Lo demás es todo correcto.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Existing or planned measures

Encryption

Action plan / corrective actions :
Como se especifica, se ha de implementar el cifrado de las bases de datos, así como el cifrado de clave asimétrica en las comunicaciones cliente a servidor.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Partitioning data

Action plan / corrective actions :
Sería conveniente descentralizar más la información almacenada en el servidor. De este modo se evita que todos los identificadores asociados a contagios estén en un mismo lugar.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Operating security

Action plan / corrective actions :
Convendría la implementación del envío de ruido al servidor para evitar la detección de tráfico dirección cliente a servidor. Este se da únicamente cuando el usuario envía un código de contagio, por lo que el no implementar el envío de ruido implica la posible escucha indeseada de identificadores contagiados.
También es necesario cifrar la base de datos, así como usar criptografía de clave privada para el cifrado de las comunicaciones con el servidor.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Traceability (logging)

Action plan / corrective actions :
Es necesario implementar un sistema de trazabilidad sobre quién ha accedido al servidor con el fin de poder detectar posibles brechas de seguridad.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Clamping down on malicious software

Action plan / corrective actions :
Se pueden añadir más medidas de seguridad, como uso de honeypots o sistemas de detección de malware.
Expected date of implementation : 01/01/2022

Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Managing personal data violations

Action plan / corrective actions :
Se puede facilitar y mejorar la detección con una implementación de trazabilidad del acceso.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García

Risks - Illegitimate access to data

Action plan / corrective actions :
Se puede disminuir la probabilidad de que estos riesgos se materialicen mediante el cifrado de las bases de datos, así como la implementación en las comunicaciones de criptografía asimétrica.
El envío de ruido al servidor también debiera ser implementado para evitar la fácil escucha de cuándo un código de contagio es enviado al servidor.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García
Taking into account the action plan, how do you re-evaluate the seriousness of this risk (Illegitimate access to data)? **Negligible**
Taking into account the action plan, how do you re-evaluate the likelihood of this risk (Illegitimate access to data)? **Negligible**

Risks - Unwanted modification of data

Action plan / corrective actions :
Se ha de implementar el cifrado de la base de datos para dificultar la modificación de su información.
Expected date of implementation : 01/01/2022
Responsible for implementation : María Ruiz Molina; Juan Velázquez García
Taking into account the action plan, how do you re-evaluate the seriousness of this risk (Unwanted modification of data)? **Maximum**
Taking into account the action plan, how do you re-evaluate the likelihood of this risk (Unwanted modification of data)? **Negligible**

Validation

TO TRANSLATE - DPO and data subjects opinion

DPO's name

María Ruiz Molina

DPO's opinion

Dado que se cumplen todas las medidas de protección de datos, así como recomendaciones para el desarrollo de aplicaciones de rastreo de contacto, el tratamiento es correcto, mínimo y exclusivamente el necesario.

Search of concerned people opinion

Concerned people opinion wasn't requested.

Reason why concerned people opinion wasn't requested

No es necesario, pues este proyecto no es tanto de carácter público sino de investigación. Por ello, los

sujetos usuarios serán controlados y voluntarios para pruebas de la aplicación.

Context
Overview

What is the processing under consideration?

El proceso aquí presentado es la **aplicación Agava COVID**, un prototipo de aplicación de **rastreo de contactos** cuya **finalidad** es la **detección de focos de contagio** y **prevención de la propagación del virus SARS-COV-2** que produce la enfermedad del COVID-19. El **contexto de uso** de esta aplicación es la **pandemia de COVID-19**, momento en el que se recurre a aplicaciones de rastreo para avisar a los usuarios de cuándo han podido establecer un contacto con una persona contagiada.
Los **beneficios** que se buscan obtener con este proyecto es frenar la expansión de la pandemia.

What are the responsibilities linked to the processing?

Los responsables del tratamiento de datos son las personas físicas **María Ruiz Molina** y **Juan Velázquez García**, desarrolladores de la aplicación y quienes han definido qué datos se recopilarán y con qué fines. También son los encargados y responsables, pues poseen acceso a la base de datos del servidor.

Are there standards applicable to the processing?

- **European Data Protection Board, Directrices 04/2020** sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19, abril de 2020.
- **Reglamento General de Protección de Datos.**
- Guías para evaluación de impacto en la privacidad de la **CNIL**.
    - Metodología.
    - Plantillas.
    - Base de conocimiento.
- Protocolo descentralizado **DP-3T.**

Evaluation : Acceptable
Evaluation comment :
La visión que se da aquí del procesamiento es correcta. Se definen todos los estándares aplicados al procesamiento y las responsabilidades de los encargados.

Context
Data, processes and supporting assets

What are the data processed?

Todos los datos recogidos refieren a los usuarios de la aplicación cliente para dispositivos móviles.

- **Estado de contagio.** Puede variar entre *sin contagio*, *con contacto contagiado* y *contagiado*. Este dato es constante en la aplicación, cambiando de valor, pero nunca saliendo del dispositivo local. Solamente su dueño puede acceder a esta información.
- **Código de contagio** proporcionado por la autoridad sanitaria a esa persona en concreto. Se envía al servidor, el cual comprueba su validez. **Una vez acabado ese proceso**, elimina el dato. Tanto la autoridad sanitaria emisora como el usuario contagiado al que está dirigido este código conocen su valor.
- **Identificadores anonimizados efímeros** intercambiados. Sus semillas generadoras se almacenan en el dispositivo del usuario receptor durante **15 días**. En el dispositivo del emisor se almacenan como activos únicamente durante 15 minutos y a los **15 días son eliminados**. Este dato se mantiene oculto. Ningún usuario puede visualizar su valor.
- **Fecha de recepción** para cada identificador anonimizado efímero recibido. Se almacenan en el dispositivo del usuario receptor durante **15 días** a modo de controlar el momento de borrado del identificador recibido. Este dato se mantiene oculto. Ningún usuario puede acceder a él.

- **Identificadores anonimizados efímeros** propios enviados al servidor **en caso de contagio.** Se almacenan en la base de datos del **servidor** durante **15 días** desde su recepción. Después de ello son eliminados. Este dato se almacena en el servidor, pudiendo ser solamente visto por administradores de la base de datos.

How does the life cycle of data and processes work?

1. El usuario instala la aplicación. En ese momento se genera y comienza a usar un **identificador efímero como activo**. Diariamente se genera una ristra de identificadores útiles a lo largo del día tal y como determina el protocolo DP-3T descrito en la memoria del proyecto.
2. El usuario permanece en contacto cercano con otro usuario y las aplicaciones, de manera automática, **intercambian los identificadores efímeros vía Bluetooth.**
3. Estos identificadores se **almacenan**, junto a la fecha de recepción, en la base de datos de cada usuario receptor.
4. El **identificador efímero propio deja de ser el identificador activo a los 15 minutos**, haya sido o no intercambiado, momento en el que otro es generado y comienza a ser usado para intercambio.
5. En caso de **contagio**, la autoridad sanitaria le proporciona un **código de contagio**, el cual introduce en la aplicación cliente de Agava COVID. Este se envía junto a las **semillas generadoras de todos los identificadores** que han estado activos en algún momento de los últimos **15 días**.
    1. En el momento del envío de un código de contagio válido, el usuario pasa al estado *contagiado*.
6. El servidor recibe el **código de contagio**, lo **comprueba** y lo **elimina**.
7. El **servidor** recibe las semillas de los identificadores y los **almacena durante 15 días** desde su recepción.
8. El **servidor** envía vía **multicast** las semillas de los identificadores contagiados a todos los **clientes**. Estos las reciben, calculan los identificadores efímeros a partir de ellas, comparan con los que poseen en su base de datos de identificadores recibidos, y **una vez acabado el proceso eliminan la información recibida en el multicast**.
    1. En caso de coincidencia, el estado de contagio pasa a ser *con un contacto contagiado*.
9. Los **identificadores recibidos por intercambio** son **eliminados** a los **15 días** contando desde su recepción.

What are the data supporting assets?

- **Sistemas operativos.**
    - Windows 10 Home, instalado en el ordenador de María.
    - Windows 8.1, instalado en el ordenador de Juan
- **Usos de negocio.**
    - Controlar y detener la expansión del virus SARS-CoV 2 mediante .
- **Sistemas de gerencia.**
    - Discord.
    - Telegram.
    - Overleaf.
    - MariaDB.
    - NetBeans
    - Android Studio.
- **Habitaciones de la oficina.**
    - Casa de María.
    - Casa de Juan.
- **Protocolos.**
    - DP-3T es el protocolo utilizado para generar los identificadores efímeros, así como administrarlos.
- **Configuraciones.**
    - Las bases de datos del servidor necesitan de un usuario y contraseña para su acceso, pudiendo únicamente acceder aquellos con permisos de administradores del sistema.
    - Las bases de datos del cliente permanecen ocultas a los usuarios de la aplicación móvil.
- **Redes.**
    - Red local de María.
    - Red local de Juan.

    - Red de Café La Passion (usada durante las pruebas Bluetooth).
- **Archivos.**
    - PDF de la memoria del proyecto.
    - PDF generado por PIA.
    - Aplicación cliente.
    - Aplicación servidor.

Evaluation : Acceptable
Evaluation comment :
Se definen los datos procesados, así como sus características y tiempo de almacenamiento. Se define correctamente el ciclo de vida para cada dato. Se describen correctamente los activos que dan soporte a los datos, incluidos aquellos activos de duración temporal.

Fundamental principles

Proportionality and necessity

Are the processing purposes specified, explicit and legitimate?

Los fines para los que los datos son recopilados son **solamente los necesarios para el correcto funcionamiento de la aplicación**.
No se venderán datos a terceros, ni saldrán de los sistemas de la propia aplicación.
Los datos, además, para evitar identificar al usuario asociado a los mismos, son **anonimizados**. Esta anonimización es debida a que los identificadores usados son identificadores generados de manera **criptográficamente segura**, **sin relación con datos del usuario**, y **efímeros** para **evitar su trazabilidad en el tiempo**, siendo su tiempo de duración activa 15 minutos.
Transcurrido ese tiempo, el identificador deja de salir del dispositivo local y otro, con valor distinto, pasa a ser el identificador efímero actualmente activo del usuario.

Ningún usuario está realmente obligado a introducir el código de contagio proporcionado por la autoridad sanitaria. Además, aquellos ciudadanos que no cuenten con la aplicación no van a verse menos beneficiados en ningún sentido ante las autoridades sanitarias o rastreadores.

La aplicación cumple únicamente el fin de ayudar a **frenar la pandemia dentro del marco de un plan sanitario mayor**. No es de uso obligatorio, aunque recomendado para detectar contactos contagiados y ayudar a contener el virus.

Los datos referentes al estado COVID de un individuo no salen de su dispositivo local, por lo que no es necesario aplicar la legislación para tratamiento de datos del RGPD.

Evaluation : Acceptable
Evaluation comment :
Los propósitos son los justos, necesarios para el funcionamiento de la aplicación y definidos.

What are the legal basis making the processing lawful?

- El usuario da el **consentimiento** para activar y hacer uso de las conexiones **Bluetooth** de la aplicación.
- El tratamiento de los datos es necesario para poder hacer funcionar la aplicación correctamente, haciendo uso de identificadores efímeros contagiados enviados desde el servidor a modo de **avisar a los usuarios de posibles contactos cercanos** con usuarios contagiados.
- El tratamiendo de datos es **parte del plan para evitar la expansión del virus** marcado por las autoridades sanitarias gubernamentales. Estos son necesarios para que funcione la aplicación Agava COVID, si bien no es obligatorio para los ciudadanos hacer uso de ella.
- El tratamiento de datos es necesario para proteger los intereses de los usuarios. Para el correcto uso de esta aplicación hace falta mantenerla activa en el dispositivo móvil cuando se salga fuera

uso de esta aplicación hace falta mantenerla activa en el dispositivo móvil cuando se salga fuera del hogar. Esto es para poder realizar intercambios de identificadores vía Bluetooth. Además, en caso de que un usuario voluntariamente envíe un código de contagio proporcionado por la autoridad sanitaria, acepta con ello la **notificación por parte de la aplicación** a los demás usuarios, pues es el **único fin de enviar dicho código** de contagio.

- El tratamiento es necesario para el desempeño de la misión de **retener la expansión del virus SARS-CoV 2**. De otro modo la aplicación no podría avisar de contactos cercanos.
- El tratamiento de datos es, en todo momento, de **datos anonimizados** con el fin de preservar la privacidad de los usuarios.

Evaluation : Improvable
Action plan / corrective actions :
Convendría la implementación de una pantalla de aceptación de la política de la privacidad. Esta debiera aparecer la primera vez que el usuario inicia la aplicación.

Are the data collected adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed ('data minimisation')?

Los datos recopilados son únicamente los necesarios para que la aplicación pueda realizar correctamente las **funciones de notificación** a los usuarios de contactos contagiados.

Es por ello que se ha optado por la tecnología **Bluetooth tradicional** en lugar de Bluetooth Low Energy, pues las guías para desarrolladores de Android determinan que en caso de usar BLE, también se han de activar los permisos de geolocalización en el móvil. Esta información no sería necesaria realmente para el funcionamiento o intereses de la aplicación y por ello se ha optado por la tecnología Bluetooth tradicional, la cual no requiere de ese permiso.

Además, con el fin de evitar la exposición de los usuarios, los datos empleados para identificarlos son **pseudónimos y efímeros** en forma de identificadores criptográficamente generados.

Evaluation : Acceptable
Evaluation comment :
Los datos recolectados son los exclusivamente necesarios para el correcto funcionamiento de la aplicación, anonimizándolos para evitar la correlación con individuos.

Are the data accurate and kept up to date?

Los **identificadores pseudónimos de los usuarios** cambian cada 15 minutos. Esto es para evitar la trazabilidad de los mismos dado su valor, pudiendo en caso de brecha, reconocerse en qué dispositivos está ese identificador y con ello con quién ha tenido contacto esa persona. Al tener una corta duración, la cantidad de dispositivos con los que ese identificador se intercambia es mínima.
Tras ello, el identificador activo pasa a tener otro valor, pero el anterior es almacenado en el dispositivo local del usuario. Esto es para que en caso de contagio, el usuario envíe al servidor todos aquellos identificadores que hayan podido intercambiarse en una ventana temporal de 15 días, tiempo que el virus está activo, y con ello avisar a aquellos usuarios con los que ha mantenido contacto estrecho en dicho periodo de tiempo.

Los **identificadores recibidos por intercambio Bluetooth** son eliminados a los 15 días de su recepción. Esto es debido a que a partir de ese momento el virus no se considera activo.

Las **semillas generadoras de identificadores almacenadas en el servidor**, las cuales están asociadas a casos de contagios, son eliminadas a los 15 días para dejar de emitirlas vía multicast a los clientes. Esto, al igual que en el punto anterior, es debido a que a partir de ese momento el virus no se considera activo.

Los **códigos de contagio** son emitidos y actualizados por una autoridad sanitaria. Cuando estos dejan de ser válidos, son borrados del servidor de la aplicación.

Evaluation : Acceptable
Evaluation comment :
Los datos poseen tiempos definidos de duración acordes al tiempo que el virus se mantiene activo. De este modo se evita mantener datos referentes a individuos que han pasado la enfermedad y ya no son contagiosos.

What are the storage duration of the data?

- **Identificadores efímeros propios.** 15 días (15 minutos activos, siendo intercambiables vía Bluetooth). Durante este tiempo se considera el virus del SARS-CoV 2 activo. Solamente está 15 minutos activo para evitar la trazabilidad, pero sus semillas generadoras se preservan durante 15 días para que, en caso de contagio, puedan ser comunicadas al servidor.
- **Identificadores efímeros recibidos.** 15 días. Durante este tiempo se considera el virus del SARS-CoV 2 activo, y por lo tanto pasado ese tiempo dejará de existir la posibilidad de recibir por multicast un aviso de contagio asociado a alguno de esos identificadores.
- **Identificadores efímeros en el servidor.** 15 días. Durante este tiempo se considera el virus del SARS-CoV 2 activo. Pasada esa fecha los identificadores dejarían de estar asociados a una persona que puede contagiar la enfermedad. Es por ello que deja de tener sentido su emisión vía multicast para avisar a los usuarios, y por lo tanto proceden a ser eliminados.
- **Códigos de contagio.** Una vez termina el proceso de comprobación (una duración de segundos). El código de contagio solo sirve para verificar que el código es uno proporcionado por la autoridad sanitaria. Una vez confirmado y recibidas las semillas asociadas a los identificadores efímeros contagiados del usuario que ha enviado la información, el código deja de ser útil. Por ello es eliminado.

Evaluation : Acceptable
Evaluation comment :
La duración de todos los datos está correctamente definida. Es acorde al tiempo que el virus se mantiene activo. Una vez los datos dejan de aportar información útil, pues refieren a casos superados, se eliminan.

Fundamental principles
Controls to protect the personal rights of data subjects

How are the data subjects informed on the processing?

La aplicación final contiene una **ventana dedicada a cuestiones informativas y política de privacidad**. En ella se encontrará toda la información relevante para el usuario, el tipo de datos recolectados y los fines y propósitos aplicados a ellos, la duración de estos datos en cada dispositivo y quién o quiénes pueden tener acceso a ellos.

Esta información no aparece en la aplicación prototipo creada con fines de demostración.

Evaluation : Improvable
Action plan / corrective actions :
Como se especifica en la nota en cursiva, es necesaria la redacción e implementación de una política de privacidad en la aplicación.

If applicable, how is the consent of data subjects obtained?

En el momento de la instalación de la aplicación, cuando se inicia por primera vez, el usuario es informado de la política de privacidad y con ello de los datos que serán recopilados. El usuario deberá confirmar en esa pantalla que ha comprendido el texto y acepta el tratamiento de los datos.

Esta pantalla no aparece en la aplicación prototipo creada con fines de demostración.

Evaluation : Improvable
Action plan / corrective actions :
Como se especifica en la nota en cursiva, es necesario implementar una pantalla donde el usuario pueda aceptar la política de privacidad la primera vez que inicie la aplicación.

How can data subjects exercise their rights of access and to data portability?

Los datos recopilados, al ser todos pseudónimos generados con fines efímeros, no debieran ser requeridos para importar a otras aplicaciones. Además, estos datos no debieran ser accesibles, siendo identificadores numéricos sin información útil fuera de la aplicación Agava COVID. Fuera de esta solo podrían usarse para asociarlos a una entidad, lo cual rompería con los fines para los cuales han sido diseñados.

Evaluation : Acceptable
Evaluation comment :
Los datos recopilados son de uso exclusivo de la aplicación, su uso no tiene sentido fuera de esta.

How can data subjects exercise their rights to rectification and erasure?

Para solicitar el borrado de los identificadores contagiados enviados al servidor junto a un código de contagio válido, los usuarios pueden ponerse en contacto con Agava en agaygava@protonmail.com

Deberán informar, de ser posible, de la fecha y hora de envío de dichos identificadores. En la aplicación final se le solicitaría un código con el que se enviarían aquellas semillas asociadas a identificadores del usuario, y estas serían eliminadas de la base de datos.

Evaluation : Acceptable
Evaluation comment :
El contacto para ejercer los derechos de rectificación y borrado es correcto y está actualizado.

How can data subjects exercise their rights to restriction and to object?

Los usuarios, en caso de que no quieran que sus identificadores sean intercambiados, tan solo deben **rechazar** en el momento de iniciar la aplicación la solicitud para activar **Bluetooth**. Han de saber que en tal caso tampoco recibirán identificadores, por lo que no podrán ser avisados en caso de contagio.

Por otro lado, en caso de contagio, **no están obligados a enviar al servidor el código de contagio** junto a las semillas generadoras de sus identificadores. Pueden elegir no introducir el código, dado que es algo voluntario. Han de saber que en caso de hacer esto la aplicación no podrá avisar a sus contactos cercanos de que han estado con un individuo contagiado.

De este modo, un usuario puede restringir qué datos envía y cuándo, así como negarse a comunicar cualquier tipo de información; aunque de este modo la aplicación no podrá avisar a otros ni avisarle a él en caso de contagio.

Evaluation : Acceptable
Evaluation comment :
El uso parcial de la aplicación es posible. Los usuarios solo deben no usar o activar aquellas características que no deseen.

Are the obligations of the processors clearly identified and governed by a contract?

Ambos encargados del tratamiento poseen el mismo nivel de acceso y tratamiento de la información. Sus responsabilidades se definen como las siguientes:

- **En cuanto a la duración.** Su responsabilidad sobre los datos tratados es de duración indefinida, pues ambos encargados son los únicos individuos con acceso completo a la aplicación por el

momento.

- **En cuanto a la naturaleza de los datos.** Los encargados tienen acceso a la base de datos del servidor, así como a la copia de seguridad del mismo. La información ahí contenida refiere a aquellos identificadores contagiados comunicados desde las aplicaciones clientes.
- **En cuanto a la finalidad del tratamiento.** Los encargados tan solo deberán manipular los datos contenidos en la base de datos en caso de necesidad de borrado de identificadores, copia de seguridad, modificación de información comunicada erróneamente o a petición de algún usuario que quiera ejercer sus derechos de rectificación o borrado.
- **En cuanto a las categorías de los interesados**. Los datos provienen de los usuarios que hayan aceptado comunicar de manera voluntaria su contagio al servidor de la aplicación Agava COVID.
- **En cuanto a las obligaciones de los responsables.** Los responsables jamás deberán otorgar accesos indebidos a la base de datos, así como hacer públicas las claves de acceso o acceder sin necesidad a la información almacenada en estas con fines no autorizados o especificados en la finalidad del tratamiento.
- **En cuanto a los derechos de los responsables.** Los responsables tienen derecho a, en caso de necesitar el acceso de terceros a la base de datos, contactar con los controladores de datos (en este caso son ellos mismos) para obtener permiso para ello. Este permiso deberá quedar por escrito para futuras referencias.

Evaluation : Improvable
Action plan / corrective actions :
Sería necesaria la redacción y firmado de un contrato. Lo demás es todo correcto.

In the case of data transfer outside the European Union, are the data adequately protected?

Los únicos datos que pueden salir fuera de la Unión Europea son aquellos **identificadores que se intercambien con individuos que posteriormente salgan de la misma**. En ese caso, las transferencias son entre dispositivos locales y únicamente de datos anonimizados que no pueden asociarse a ningún individuo. Estos datos no irán a parar en ningún momento a servidores de entidades fuera de la Unión Europea, pues los servidores donde se almacena la información de identificadores contagiados y que se encargan de realizar el multicast se encuentran dentro de la Unión Europea.

Evaluation : Acceptable
Evaluation comment :
Los datos no saldrán fuera de la Unión Europea, estos siempre irán dirigidos a servidores dentro de la misma.

Risks

Planned or existing measures

Encryption

- Los identificadores son generados de manera criptográficamente segura con el fin de proporcionar una alta entropía. De este modo, no se podrán asociar de ningún modo un grupo de identificadores como provenientes de un mismo usuario.
- La comunicación establecida cliente a servidor en el momento de comunicar un contagio se enviará de forma cifrada mediante criptografía de clave simétrica previamente establecida.
    - Esta comunicación sería conveniente actualizarla a un cifrado asimétrico empleando claves públicas y privadas, como puede ser el algoritmo RSA.
- La comunicación entre dispositivos locales vía Bluetooth, al usar Bluetooth tradicional, los dispositivos quedan enlazados y pareados. De esta manera se genera un canal de comunicación cerrado.
- Cifrado de las bases de datos. *En esta aplicación prototipo todavía no se ha implementado.*

Evaluation : Improvable
Action plan / corrective actions :
Como se especifica, se ha de implementar el cifrado de las bases de datos, así como el cifrado de clave asimétrica en las comunicaciones cliente a servidor.

Anonymisation

- Los identificadores efímeros usados para identificar a los usuarios no emplean datos de estos. El protocolo empleado para su generación y gestión permite identificar cuándo se ha tenido un contacto contagiado pero no quién es exactamente. Esto es gracias a que los identificadores se generan de manera criptográfica y no mediante datos personales del usuario.
- Los códigos emitidos por la autoridad sanitaria para comunicar un contagio al servidor son códigos numéricos. Estos no poseen ningún tipo de relación con la información del usuario contagiado.

Evaluation : Acceptable
Evaluation comment :
Tanto los identificadores como los códigos son generados de forma que no puedan relacionarse con su propietario.

Minimising the amount of personal data

- Los datos personales son minimizados mediante el uso de identificadores anónimos. Estos no asocian la información de contagio con ningún usuario.
    - El estado de contagio únicamente es tratado y mostrado de forma local, luego la legislación para tratamiento de los datos no aplicaría.

Evaluation : Acceptable
Evaluation comment :
Los datos requeridos son los exclusivamente necesarios para el correcto funcionamiento de la aplicación.

Partitioning data

- El protocolo utilizado es DP-3T, un protocolo descentralizado. Esto permite que los identificadores en lugar de estar puramente almacenados en un servidor, están distribuidos en cada dispositivo local. De esta manera, cada usuario es dueño únicamente de sus identificadores y aquellos obtenidos por intercambio.
- El servidor, por otro lado, posee aquellos identificadores contagiados pero, al ser efímeros no podrían rastrearse en dispositivos clientes de manera activa. Por otro lado, al ser información anonimizada, no aportan por sí solos datos suficientes como para asociarlos a un individuo concreto.

Evaluation : Improvable
Action plan / corrective actions :
Sería conveniente descentralizar más la información almacenada en el servidor. De este modo se evita que todos los identificadores asociados a contagios estén en un mismo lugar.

Logical access control

- Los responsables del tratamiento de los datos, así como los controladores, poseen un usuario y contraseña únicos para el acceso a la base de datos del servidor de la aplicación.

Evaluation : Acceptable
Evaluation comment :
Control mediante usuario y contraseña únicos correcto.

Archiving

- Los datos contenidos en la base de datos del servidor tan solo serán transferidos a la copia de seguridad del servidor de forma periódica diaria.
    - Solo los responsables del tratamiento de datos podrán realizar esta acción.

Evaluation : Acceptable
Evaluation comment :
Correcta restricción de acceso y transferencia de datos.

Paper document security

- La memoria del proyecto permanecerá pública, pues la información contenida en ella no es confidencial.
    - Los datos verdaderamente confidenciales son los generados por la propia aplicación, no el código o la memoria.

Evaluation : Acceptable
Evaluation comment :
La memoria y código son públicos.

Operating security

- **Cifrado de las comunicaciones** para evitar el *sniffing* de identificadores enviados.
- **Uso de identificadores efímeros para evitar el rastreo de los usuarios.** De esta forma, no se podrán relacionar sus contactos, pues cada 15 minutos el identificador cambia a otro imposible de relacionar con el anterior.
- **Descentralización de los datos.** De esta forma, en caso de brecha de seguridad en algún dispositivo, no todos los identificadores serían expuestos.
- **Uso de identificadores anonimizados.** Así, no se pueden asociar a usuarios concretos. Además, dos identificadores de un mismo usuario tampoco poseen ningún patrón, información o dato en común para evitar su asociación.
- **Uso de Bluetooth tradicional** con el fin de crear canales seguros de comunicación en el momento de intercambio de identificadores.
- **Acceso a la base de datos mediante un usuario y contraseña**, propios de los procesadores/controladores de los datos.
- **Bases de datos cifradas.** *En esta aplicación prototipo todavía no está implementado.*

Evaluation : Improvable
Action plan / corrective actions :
Convendría la implementación del envío de ruido al servidor para evitar la detección de tráfico dirección cliente a servidor. Este se da únicamente cuando el usuario envía un código de contagio, por lo que el no implementar el envío de ruido implica la posible escucha indeseada de identificadores contagiados.
También es necesario cifrar la base de datos, así como usar criptografía de clave privada para el cifrado de las comunicaciones con el servidor.

Traceability (logging)

- Trazabilidad del acceso a la base de datos del servidor. *En esta aplicación prorotipo todavía no se ha implementado.*

Evaluation : Improvable
Action plan / corrective actions :
Es necesario implementar un sistema de trazabilidad sobre quién ha accedido al servidor con el fin de poder detectar posibles brechas de seguridad.

Clamping down on malicious software

- Control de que únicamente los puertos que usa la aplicación para comunicarse estén abiertos. Estos además estarán protegidos mediante firewall en el servidor.

Evaluation : Improvable
Action plan / corrective actions :
Se pueden añadir más medidas de seguridad, como uso de honeypots o sistemas de detección de malware.

Managing workstations

- Los Sistemas Operativos se actualizarán de manera periódica con cada nueva actualización.
- El gestor de base de datos MariaDB se actualizará de manera periódica para evitar brechas en el software.

Evaluation : Acceptable
Evaluation comment :
La gestión es correcta.

Backups

- El servidor cuenta con una copia de seguridad. El servidor principal está desplegado en el ordenador de María Ruiz Molina. La copia en el de Juan Velázquez García.
    - Con ello se incluye la base de datos.
- El código de todo el proyecto (cliente y servidor) cuenta con dos copias en local, una en cada ordenador de los autores; y otra en Github.
- La memoria del proyecto cuenta con dos copias en local, una en cada ordenador de los autores; y otra en Overleaf.

Evaluation : Acceptable
Evaluation comment :
Se poseen varias copias de seguridad, tanto locales como en la nube.

Network security

- Control de qué puertos están abiertos para evitar tener abiertos puertos no usados.
- Firewall de Kaspersky Total Security para el servidor principal.
- Firewall del sistema operativo Windows 8.1 para el servidor copia de seguridad.
- Conexiones cifradas con el servidor.
- Uso de multicast en lugar de broadcast para que la información sea enviada a un único grupo de multicast, específico de la aplicación, en lugar de a todos los dispositivos.

Evaluation : Acceptable
Evaluation comment :
Las redes se mantienen lo más seguras posibles.

Physical access control

- Al tratarse de los ordenadores personales de los autores del proyecto, estos se encuentran en sus respectivas casas con la protección propia de una vivienda.
- La red empleada en su momento durante el periodo de pruebas de Bluetooth, de La Passion Cafe, es pública al igual que las instalaciones. De todos modos, fue un lugar empleado de manera esporádica y solo para dichos fines de pruebas.

Evaluation : Acceptable
Evaluation comment :
A pesar de haber hecho uso de una red pública durante un periodo de pruebas, no podía ser de otro modo debido a las medidas de contingencia de la pandemia. Durante ese periodo solo se probaron funciones Bluetooth, luego las comunicaciones con el servidor no estuvieron en peligro.

Maintenance

- En caso de que la aplicación servidor lo necesite, se realizará el mantenimiento requerido. Cuando las aplicaciones clientes requieran de algún tipo de supervisión se hará mediante actualizaciones.
- Los dispositivos que actúan como servidores son ordenadores particulares por ahora. En caso de moverse a servidores grandes, estos requerirían de refrigeración, limpieza de polvo, entre otros, para evitar problemas producidos por sobrecalentamiento o semejante.

Evaluation : Acceptable
Evaluation comment :
El mantenimiento se lleva de forma controlada, mediante control de versiones en un repositorio público.

Monitoring network activity

- Para monitorizar las conexiones dentro de la red empleada para pruebas se hace uso de WNetWatcher, con el fin de detectar dispositivos no deseados pudiendo establecer algún tipo de conexión malintencionada.

Evaluation : Acceptable
Evaluation comment :
Las medidas, dadas las circunstancias de la red, son por ahora aceptables y suficientes.

Avoiding sources of risk

La zona donde se encuentran los servidores, tanto el principal como la copia de seguridad, está libre de inundaciones, de erupciones volcánicas, la posibilidad de terremoto es ínfima y la de incendio también pues no hay cableado especialmente sensible. Tampoco hay proximidad con industrias químicas que pudieran dañar a los equipos.

Evaluation : Acceptable
Evaluation comment :
Las zonas donde los servidores se encuentran son correctas.

Protecting against non-human sources of risks

La fuente eléctrica empleada para el servidor es la fuente del propio piso. Esta no es un generador potente que pueda suponer un riesgo grave.
Para prevención de incendios a lo largo del edificio se encuentran extintores.

Evaluation : Acceptable
Evaluation comment :

Las prevenciones son correctas.

Organisation

Ambos autores de la aplicación son responsables de la privacidad de la misma. En caso de necesidad de refuerzo de la privacidad de los datos, estos serán los encargados. Deberán incorporar mediante actualizaciones dichas mejoras tanto al servidor como a la aplicación cliente.

Ellos mismos se encargan de monitorizar que la privacidad de la aplicación sea la debida. En caso de que alguna mejora sea detectada, se deberá implementar.

Evaluation : Acceptable
Evaluation comment :
Dadas las dimensiones del proyecto, la organización es adecuada.

Policy

La siguiente política refiere a los usuarios de la aplicación cliente.

- **Riesgos.**
  - En caso de brecha de seguridad los identificadores revelados pueden ser usados para generar focos de contagio irreales. En caso de que así suceda, se avisará a los clientes a través de la aplicación de que si reciben una notificación de contagio esta pudiera ser real o no.
  - En caso de brecha de seguridad, una entidad malintencionada podría ubicar su propio servidor y recibir los códigos de contagio que los usuarios envíen. En caso de detección de un caso así, la aplicación informará a sus usuarios y tomará medidas a la mayor brevedad.
- **Principios claves a seguir.**
  - Para el correcto funcionamiento de la aplicación, el usuario debe mantenerla activa, así como comunicar al servidor el código de contagio en caso de obtener uno. En caso de no hacerlo, pues es algo voluntario, el usuario entiende que no podrá recibir notificaciones de contactos cercanos ni comunicar su contagio a sus contactos a través de Agava COVID.
- **Destinatarios.**
  - Todos los usuarios que descarguen la aplicación cliente.
- **Reglas que se aplicarán.**
  - El tratamiento de los datos de los usuarios cumple la legislación del Reglamento General de Protección de Datos, así como las guías y recomendaciones de la Commission Nationale de l'Informatique et des Libertés y del European Data Protection Board.

La siguiente política refiere a los autores y controladores de la aplicación.

- **Principios claves a seguir.**
  - Al tener acceso a la base de datos de la aplicación, los controladores y procesadores de datos no deberán tratar la información de modos idebidos y que no hayan sido especificados previamente en esta documentación.
- **Destinatarios.**
  - Los autores de la aplicación, que actúan como controladores de datos, procesadores de datos, administradores.
    - María Ruiz Molina.
    - Juan Velázquez García.
- **Reglas que se aplicarán.**
  - El tratamiento de datos deberá cumplir en todo momento la legislación del Reglamento General de Protección de Datos, así como las guías y recomendaciones de la Commission Nationale de l'Informatique et des Libertés y del European Data Protection Board.
  - En caso de brecha de seguridad o necesidad de refuerzo de la privacidad, los autores deberán actuar implementando las medidas correspondientes lo antes posible. De ser necesario, avisarán a las entidades

Evaluation : Acceptable
Evaluation comment :
La política especificada es aceptable acorde a los sujetos y controladores de los datos.

Managing privacy risks

- Cumplimentación con el Reglamento General de Protección de Datos.
- Cumplimentación con las guías y recomendaciones de la Commission Nationale de l'Informatique et des Libertés.
- Cumplimentación con las guías del European Data Protection Board para aplicaciones de rastreo de contacto.
- Realización de análisis de impacto en la privacidad mediante:
  - La herramienta PIA de la CNIL (por María Ruiz Molina).
  - La herramienta PILAR del CCN-CERT (por Juan Velázquez García).
- Apartado en la memoria con conclusiones sobre la evaluación de riesgos en la privacidad que pudieran ocasionarse.

Evaluation : Acceptable
Evaluation comment :
Se cumple todo el reglamento y recomendaciones, así como se han elaborado análisis de privacidad y seguridad del proyecto.

Managing personal data violations

La detección sobre si se ha producido un incidente que afecte a los datos de los usuarios es posible en caso de que:

- Se detecte un dispositivo extraño conectado a la red donde se encuentra el servidor.
- Se detecten modificaciones en el código del servidor.
- Se detecten alteraciones, que ninguno de los dos administradores de la base de datos ha realizado, en la información de la base de datos.
- Se detecten versiones de la aplicación cliente no oficiales.
- Se detecten servidores no oficiales activos.

Evaluation : Improvable
Action plan / corrective actions :
Se puede facilitar y mejorar la detección con una implementación de trazabilidad del acceso.

Risks

Illegitimate access to data

What could be the main impacts on the data subjects if the risk were to occur?

Detección de focos de contagio en un corto plazo, Escucha de los identificadores contagiados, Escucha de los intercambios Bluetooth

What are the main threats that could lead to the risk?

Acceso indebido a la base de datos del servidor, Acceso indebido a las bases de datos de los clientes, Escucha de la comunicación tras descifrado, Creación de servidor malintencionado

What are the risk sources?

Ciberdelincuente haciendo objetivo a un usuario, Ciberdelincuente haciendo objetivo a toda la base de datos, Ciberdelincuente modificando focos de contagio, Error de administrador de base de datos

Which of the identified planned controls contribute to addressing the risk?

Encryption, Anonymisation, Minimising the amount of personal data, Partitioning data, Logical access control, Traceability (logging), Clamping down on malicious software, Operating security, Network

security, Physical access control, Monitoring network activity

How do you estimate the risk severity, especially according to potential impacts and planned controls?

Negligible, Las medidas tomadas, sobre todo la anonimización de los datos y el hecho de que los identificadores sean efímeros, así como el uso de un protocolo descentralizado, dificulta a los atacantes obtener información útil de los usuarios.

El riesgo es limitado, pues la información que pudiera salir a la luz por ello serían focos de contagio o contactos, aunque de manera muy limitada. Dado que no hay forma de asociar un identificador a individuos concretos, el impacto no es tan grande.

How do you estimate the likelihood of the risk, especially in respect of threats, sources of risk and planned controls?

Limited, A pesar de las medidas ya implantadas, hay algunos aspectos mejorables en la aplicación, como son:

- Cifrado de las bases de datos.
- Uso de criptografía de clave pública y privada en las comunicaciones con el servidor en lugar de clave simétrica fija.

Para poder ocasionar estos riesgos, es necesario un filtrado de la clave de cifrado usada en el cliente y servidor para las comunicaciones o bien el acceso a la base de datos del servidor.
Dado que la primera no es pública, y la segunda está protegida por usuario y contraseña de los administradores, no es algo que se espere ocurra de forma frecuente.

Evaluation : Improvable
Action plan / corrective actions :
Se puede disminuir la probabilidad de que estos riesgos se materialicen mediante el cifrado de las bases de datos, así como la implementación en las comunicaciones de criptografía asimétrica.
El envío de ruido al servidor también debiera ser implementado para evitar la fácil escucha de cuándo un código de contagio es enviado al servidor.
Taking into account the action plan, how do you re-evaluate the seriousness of this risk (Illegitimate access to data)? **Negligible**
Taking into account the action plan, how do you re-evaluate the likelihood of this risk (Illegitimate access to data)? **Negligible**

Risks

Unwanted modification of data

What could be the main impacts on the data subjects if the risk were to occur?

Creación de contactos irreales, Creación de focos de contagio irreales

What are the main threats that could lead to the risk?

Envío indebido de identificadores "sanos" al servidor, Creación de servidor malintencionado, Aplicación cliente falsa malintencionada

What are the risk sources?

Ciberdelincuente haciendo objetivo a un usuario, Ciberdelincuente modificando focos de contagio, Error de administrador de base de datos, Ciberdelincuente haciendo objetivo a toda la base de datos

Which of the identified controls contribute to addressing the risk?

Logical access control, Operating security, Clamping down on malicious software, Backups, Managing personal data violations, Managing privacy risks, Monitoring network activity, Encryption, Partitioning data, Traceability (logging), Network security, Physical access control

How do you estimate the risk severity, especially according to potential impacts and planned controls?

planned controls?

Maximum, La generación de avisos a personas que realmente no han tenido contacto con contagiados crea focos de contagio falsos. Estos focos de contagio falsos pueden ocasionar que se desconozca quién realmente ha podido establecer un contacto, ocasionando cuarentenas y consultas médicas realmente innecesarias.

Esto en el peor caso supondría que todos los usuarios fuesen avisados de un posible contacto, lo que ocasionaría la completa inutilización de la aplicación y una pérdida de los casos reales.

How do you estimate the likelihood of the risk, especially in respect of threats, sources of risk and planned controls?

Limited, Dado que para acceder a la base de datos es necesario un usuario y contraseña únicos de los administradores, así como sobrepasar las barreras del firewall.
Además se planea el cifrado de las bases de datos (característica no implementada en el prototipo).

Evaluation : Improvable
Action plan / corrective actions :
Se ha de implementar el cifrado de la base de datos para dificultar la modificación de su información.
Taking into account the action plan, how do you re-evaluate the seriousness of this risk (Unwanted modification of data)? **Maximum**
Taking into account the action plan, how do you re-evaluate the likelihood of this risk (Unwanted modification of data)? **Negligible**

Risks

Data disappearance

What could be the main impacts on the data subjects if the risk were to occur?

Borrado de base de datos, Evasión de focos de contagio

What are the main threats that could lead to the risk?

Accidentes físicos en servidor y copia de seguridad

What are the risk sources?

Fuego, Agua u otros líquidos, Errores en el sistema operativo, Error de administrador de base de datos, Individuo dañando físicamente los dispositivos, Accidente físico en ambos servidores simultáneamente

Which of the identified controls contribute to addressing the risk?

Partitioning data, Operating security, Traceability (logging), Clamping down on malicious software, Physical access control, Backups, Avoiding sources of risk, Monitoring network activity, Network security, Protecting against non-human sources of risks, Logical access control

How do you estimate the risk severity, especially according to potential impacts and planned controls?

Important, El borrado de información significaría la pérdida de identificadores contagiados y de contactos, por lo que no se podría avisar a algunos individuos de la posibilidad de un contacto cercano con un usuario contagiado. Tampoco se podría realizar un adecuado intercambio de identificadores en caso de afectar a ciertas aplicaciones cliente.

De suceder esto, el virus se propagaría más fácilmente, pues los usuarios no podrían hacer cuarentena preventiva en caso de haber tenido contacto estrecho con un usuario contagiado.

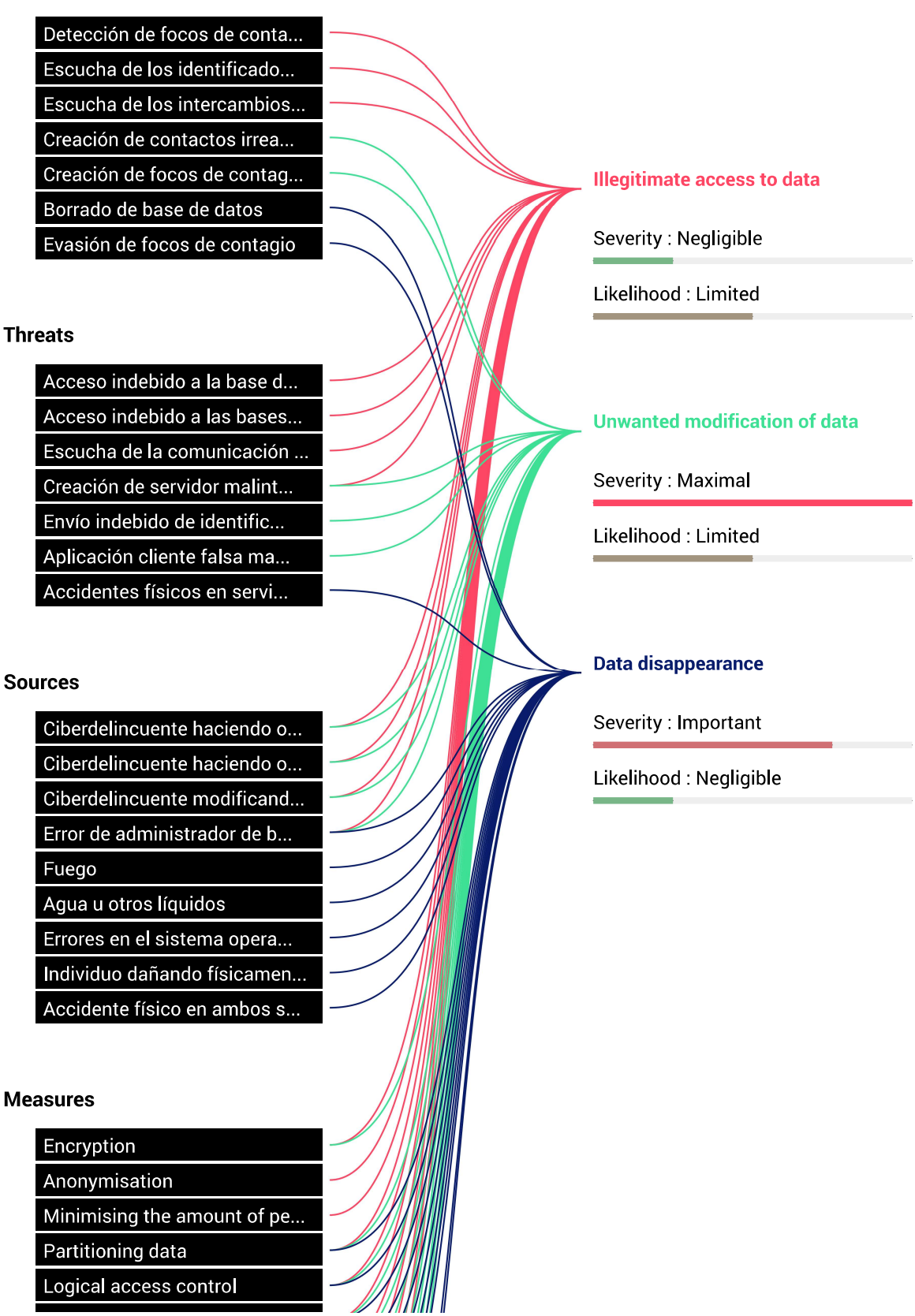
How do you estimate the likelihood of the risk, especially in respect of threats, sources of risk and planned controls?

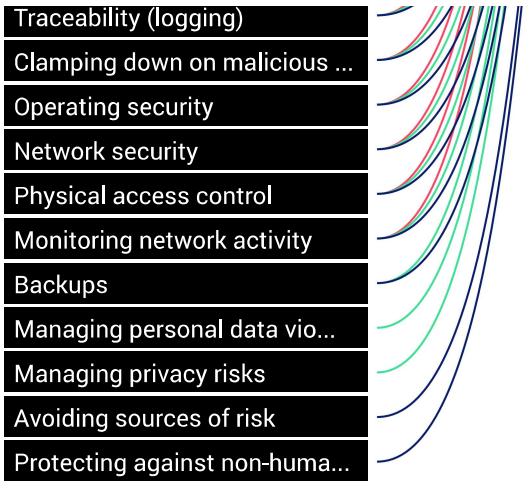
Negligible, Debido a la existencia de copias de seguridad, la reincorporación de esta información sería rápida. Además, aquellos identificadores que no se hubieran emitido por multicast para avisar a los clientes, lo harían en cuanto la base de datos se hubiese reparado, pues dicha emisión es periódica y constante.

Evaluation : Acceptable
Evaluation comment :
Dada la existencia de copias de seguridad, así como el uso de un protocolo descentralizado, el borrado de información se dificulta.

Risks

Risks overview




Traceability (logging)
Clamping down on malicious ...
Operating security
Network security
Physical access control
Monitoring network activity
Backups
Managing personal data vio...
Managing privacy risks
Avoiding sources of risk
Protecting against non-huma...

6.3.5. Conclusiones de la evaluación

Como puede verse, el PIA tiene puntos a mejorar. Se califica como aceptable para concluir este proyecto, pues dado el alcance de este es suficiente.

Si la aplicación fuese de carácter público con unos sujetos no voluntarios para las pruebas, como es el caso de cualquier aplicación de rastreo de contactos desarrollada durante la pandemia, el PIA sería mejorable.

Esto llevaría a realizar una iteración más del análisis. Previamente se habrían de incorporar aquellos aspectos más importantes para la seguridad de los datos. Del análisis se extraen los siguientes.

- **Cifrado de las bases de datos.** Para que en caso de acceso indebido, los datos no quedaran expuestos o modificados.
- **Cifrado asimétrico en las comunicaciones con el servidor.** Implementación de criptografía de clave asimétrica. El servidor otorgaría su clave pública al cliente para que este cifrase con ella la clave simétrica para futuras comunicaciones. Tras ello se la mandaría al servidor, el cual podría descifrarla usando su clave privada. Además, el servidor podría tener su clave pública validada por una Autoridad Certificadora con el fin de ser más confiable.
- **Envío de ruido al servidor.** Dado que el único flujo de información cliente a servidor es cuando se comunica un contagio, mediante un ataque de escucha pueden detectarse. Para evitar esto, se generaría ruido emitido de cliente a servidor, así no sería posible mediante el *sniffing* diferenciar el ruido de datos reales.
- **Trazabilidad en el servidor.** Con el fin de detectar accesos indebidos al servidor, debiera hacerse un registro de los mismos. De este modo sería más fácil detectar si ha ocurrido una brecha de seguridad o frenarlo a tiempo bloqueando al atacante de algún modo.
- **Redacción de una política de privacidad en la aplicación.** Con el fin de que los usuarios queden correctamente informados y puedan aceptarla nada más iniciar la aplicación.
- **Elaboración de un contrato.** Este sería firmado por los encargados del procesamiento de los datos con el fin de llevar un control regulado de los mismos, de sus deberes y de sus derechos.

Con estas mejoras el PIA quedaría mejorado. De esta forma probablemente se pudiera validar tras la nueva iteración, siempre revisando de nuevo punto por punto para que no se escapasen nuevos o anteriores detalles.

6.4. Comparativa de resultados con otras alternativas europeas

Como puede verse el prototipo necesita todavía un par de actualizaciones que permitan asegurar la privacidad de los datos más aún. Esto es implementar el cifrado de las bases de datos y el uso de la criptografía de clave pública, así como el envío de ruido de forma periódica al servidor.

El prototipo, por lo tanto, aún tiene aspectos que son fácilmente mejorables. Suponiendo su implementación, el análisis PIA quedaría validado y con ello una protección de los datos maximizada.

La aplicación sin embargo posee ciertas debilidades por diseño. La principal es el hecho de estar operativa constantemente en segundo plano haciendo uso de comunicaciones Bluetooth. Esta característica no solo mantiene una puerta abierta a la escucha de tráfico de datos vía Bluetooth, sino que además no ha inspirado mucha confianza entre los usuarios.

El uso de una aplicación de rastreo de contactos puede ser útil para frenar el número de casos, pero de nada sirve si la acogida entre los ciudadanos no es mayoritaria.
Las primeras opciones que se propusieron, de protocolos centralizados, fueron rápidamente descartadas, dando lugar a los protocolos descentralizados como el usado en Agava COVID.

Actualmente, la Unión Europea ante las nuevas necesidades de la pandemia y de un control más globalizado, propone el sistema de Self-Sovereign Identity.

Ambas opciones se encargan de controlar el estado COVID de los ciudadanos, si bien en distintos momentos de la pandemia.

Una aplicación de rastreo de contactos que hace uso de la tecnología Bluetooth y es descentralizada, almacena de forma local únicamente la información sobre el propio contacto. Sin embargo, sigue habiendo presente un servidor en cierto modo centralizado, el cual se encarga de recibir todos aquellos identificadores contagiados. Aunque estos no poseen correlación directa con sus propietarios, el hecho de que haya un único punto de almacenamiento de los identificadores contagiados, implica una mayor atracción al mismo en caso de ataque.

Si el servidor no estuviera correctamente configurado o surgiera algún tipo de problema, accediendo al mismo una persona no autorizada con fines malintencionados, podría introducir identificadores realmente no contagiados o borrar los contagiados. De este modo podría manipular los focos de contagio, creando nuevos y eliminando los realmente existentes.

Aunque se tomasen todas las medidas posibles para evitar el riesgo, este nunca sería nulo y el hecho de que esta parte de los datos se almacene de modo centralizado, puede suponer peligros para la población, ya no solo usuaria de la aplicación sino también aquella en contacto con los usuarios, provocando un efecto en cadena.

Dadas las circunstancias actuales en las que además, el interés sobre el estado COVID ya no son tanto los focos de contagio sino quién está ya inmunizado (sea mediante vacuna o por haber pasado la enfermedad), los intereses tecnológicos también varían en parte.

El modelo de identidad autosoberana (Self-Sovereign Identity) permite el almacenamiento de la información de modo exclusivamente local. De esta manera los usuarios son completamente dueños de sus datos sanitarios y pueden elegir a quién se los presentan.

Los usuarios podrían obtener estas Credenciales Verificables de autoridades sanitarias, las cuales las crearían con toda la información necesaria, como tipo de vacuna y número de dosis.

Así, cuando una entidad proveedora de servicios que necesita comprobar si un usuario ha podido pasar la enfermedad o ha recibido una vacuna. Además, dependiendo de las circunstancias, el usuario podría mostrar más o menos información.

En caso de que una página que organiza viajes solicite información sobre si el ciudadano está inmunizado,

el usuario podría elegir presentar únicamente una Credencial Verificable con el dato de *Sí* o *No*. Este dato al llevar la firma digital de una entidad sanitaria podría ser comprobado por el proveedor de servicios y con ello dado por válido y confiable. Además, la transacción quedaría registrada en la blockchain para garantizar el no repudio, almacenando solo los datos referentes a que esa interacción se ha realizado, sin guardar datos privados como el valor de la Credencial Verificable.

Por otro lado, si el proveedor de servicios solicita el tipo de vacuna administrada, siendo por ejemplo un centro médico, la Credencial Verificable esta vez no sería únicamente un predicado afirmativo o negativo, sino que contendría el tipo de vacuna recibida. De nuevo esa información iría firmada por la entidad sanitaria emisora que actuó como *Issuer*, dando por válido el dato.

Está claro que este modelo presenta una estructura mucho más segura, puesto que no debe estar constantemente activo y hace al usuario dueño de sus datos.

Las aplicaciones de rastreo de contacto tuvieron muchas dificultades, siendo una tecnología desarrollada rápido, bajo mucha presión y críticas constantes. El paso a unas nuevas necesidades las deja atrás, por ahora, pero no quita que deba ser un campo a todavía investigar en caso de otra pandemia que necesite de control de focos de contagio.

Quizá una extensión del modelo Self-Sovereign Identity para estos casos o nuevos protocolos que hagan uso de otras tecnologías y tipos de conexiones, ayuden a, desde la calma, crear aplicaciones verdaderamente confiables y con un alto grado de privacidad y seguridad.

7. Conclusiones y Trabajo Futuro

Finalmente, se exponen las conclusiones obtenidas de este trabajo. Primeramente, se revisa qué objetivos se han logrado, así como en qué cantidad se han podido llegar a satisfacer. De estos objetivos y del trabajo mejorable surge el apartado de *A futuro*, donde se exponen aquellos puntos mejorables o nuevos apartados que podrían complementar este proyecto.

Finalmente, se cierra con una conclusión general sobre el proyecto y lo aprendido durante su desarrollo.

Objetivos cumplidos

Los objetivos planteados inicialmente pueden encontrarse en la sección *Objetivos*.

Una vez se ha terminado de elaborar el proyecto se hace retrospectiva para ver qué objetivos se han completado

- **Objetivo 1.** Se ha podido llevar a cabo una profunda investigación tanto sobre los protocolos de rastreo de contactos, como sobre el posicionamiento de la Unión Europea ante estos, la evolución del estado COVID, y también sobre materias de seguridad y privacidad en aplicaciones móviles y conexión Bluetooth.

  Se ha podido observar cómo ha evolucionado a lo largo del año los distintos planteamientos tecnológicos ante la pandemia. Desde protocolos de rastreo de contactos en proceso de desarrollo, sus diversas perspectivas y enfoques en la privacidad, hasta nuevas propuestas como el modelo de Self-Sovereign Identity.

  En cuanto a seguridad de los datos y privacidad, se ha podido indagar sobre la legislación europea entorno a este tema. Tanto recomendaciones para elaborar aplicaciones de rastreo de contactos, como guías para hacer análisis de impacto en la privacidad o diversos puntos del Reglamento General de Protección de Datos.

- **Objetivo 2.** Respecto a la realización de una aplicación prototipo de rastreo de contactos se puede considerar cumplido. Del desarrollo de este trabajo ha nacido Agava COVID, una aplicación prototipo con mucho potencial de desarrollo.

- **Objetivo 3.** Se ha elaborado un análisis de impacto en la privacidad mediante el uso de la herramienta *PIA*, herramienta de la CNIL. En este se ha podido llevar a cabo el proceso para corroborar que se cumple con el RGPD en todo el proyecto, sacar a la luz las amenazas más importantes para los datos personales, su nivel de impacto y probabilidad; y medidas para contrarrestarlo.

- **Objetivo 4.** De este trabajo se han obtenido diversas conclusiones. El desarrollar una aplicación de rastreo de contactos implica muchos retos para la privacidad. Los datos sanitarios de los usuarios son de especial sensibilidad y es por ello que las soluciones tecnológicas para controlar la pandemia deben ser especialmente seguras y privadas. También se ha podido ver cómo la UE se va adaptando a las necesidades que emergen con el transcurso de la pandemia, elaborando protocolos y soluciones cada vez más seguros y privados.

A futuro

En esta sección se exponen los puntos a tratar en el futuro del proyecto, pues bien no se han podido llevar acabo por falta de tiempo o porque se escapan del alcance de este trabajo.

- **Mejoras en el uso de la tecnología Bluetooth.** Hasta ahora el prototipo necesita de interacción activa por parte del usuario para el envío de identificadores por Bluetooth. Esta funcionalidad debería ser implementada tal que funcionase en segundo plano y de manera automática. Además, el alcance de la señal de Bluetooth debería restringirse a 2 metros en lugar de a la distancia predeterminada.
- **Borrado automático de identificadores caducados en la base de datos del servidor.** Este proceso debería ser automatizado para mejorar el rendimiento de la aplicación servidor, evitar el almacenamiento de datos superfluos y cumplir fielmente con la normativa del RGPD y la no conservación indefinida de los datos.
- **Mejor tratamiento de los identificadores.** Hasta ahora, con el fin de controlar las pruebas, los identificadores se rotaban de manera manual. Estos deberían ir rotando de forma automática cada 15 minutos. También, un punto a tener muy en cuenta, es la necesidad de envío de ruido al servidor, pues de otro modo mediante la escucha del canal puede averiguarse quién envía códigos de contagio, pues es la única comunicación que hay dirección al servidor. Para ello se implementaría el envío de identificadores falsos, los cuales llegarían igualmente a la base de datos, solo que serían descartados al momento, pues tendrían una fecha superior a 14 días. Con ello, la base de datos los eliminaría al comprobar la existencia de identificadores caducados.
- **Pulir el funcionamiento de la aplicación cliente.** Hay varias mejoras que podrían implementarse en la aplicación cliente con el fin de perfeccionar su interacción con el usuario. Estos aspectos a pulir son automatizar el cambio de estado de contagiado a sin contactos, un aviso como notificación móvil en el momento en el que el estado de contagio cambie, y la retroalimentación por parte del servidor en el momento de validar si un código de contagio es o no aceptado.
- **Despliegue de la aplicación servidor en Internet.** Para conseguir que la aplicación funcione a nivel del gran público habría que desplegar la aplicación servidor en la red. Por supuesto, se debería revisar su correcto funcionamiento y, si fuese necesario adaptar el código para ello, así como revisar los apartados pertinentes en el análisis de impacto sobre la privacidad debido a su mayor exposición al público.
- **Cifrado usando pares de clave pública y privada.** Para la aplicación prototipo se ha utilizado el cifrado simétrico AES-256, pero para una mayor protección de los datos enviados por red, se habría de implementar un cifrado de clave asimétrica, como podría ser RSA. De este modo, al inicio de la conexión TCP, el servidor envía al cliente su clave pública, con la que el cliente cifra la clave simétrica para posteriores comunicaciones. El servidor podría obtener esa clave simétrica descifrando el mensaje con su clave privada. Tras ello usaría la clave simétrica para comunicarse con el cliente.
- **Redacción de política de privacidad.** Para cumplir con la normativa del RGPD se debería redactar una accesible desde la aplicación, al igual que implementar una pantalla de aceptación de la política al momento de iniciar por primera vez la aplicación.
- **Implementar las contramedidas especificadas en el análisis.** Para que este proyecto sea viable se deberían minimizar los riesgos que surgidos en el análisis. Entre ellos los más destacables refieren al cifrado de la base de datos y el uso de criptografía de clave pública para las comunicaciones cliente con servidor.

Como puede verse, todavía hay muchas líneas de trabajo que pueden ser desarrolladas y mejoradas. Muchas de ellas refieren a la implementación, así como al tipo de conexiones empleadas por los protocolos de rastreo de contacto más populares, incluido DP-3T.

Es por ello que en este tipo de aplicaciones de control sobre el estado COVID, todavía hay ciertos riesgos desde el punto de vista de diseño del protocolo.

Además, debido a su funcionamiento, el cual obliga al usuario a mantener la aplicación activa en segundo plano con conexión Bluetooth constante, no ha inspirado confianza en gran parte de la población.

Es por ello que, la búsqueda de tecnologías alternativas que se adapten mejor a la situación post-pandemia y a la vez sean altamente privadas y seguras, es un movimiento que ya se está dando.

Modelos como Self-Sovereign Identity, descentralizados y basados en tecnologías blockchain bajo la protección de la Unión Europea, como la EBSI, pretenden convertirse en el nuevo Internet, uno más privado y seguro, donde la información del usuario sea realmente suya, y donde bien pudiera incluirse el estado COVID de los ciudadanos.[60]

Valoración personal

El desarrollo de una aplicación de rastreo de contactos es un proyecto que sirve como punto de partida para indagar en cómo la tecnología, a lo largo de este año, ha ido evolucionando y adaptándose rápidamente a las necesidades de la pandemia.

Desde protocolos iniciales que fueron rechazados por cientos de expertos en ámbitos de privacidad, pasando por protocolos descentralizados y con un enfoque centrado en la seguridad de los datos del usuario, hasta un sistema completamente rompedor con el concepto actual de identidad digital y la información asociada a cada ciudadano. Todos estos ámbitos han ido de la mano y han podido ser explorados a lo largo de este año, un año en el que la evolución tecnológica ha sido enorme y centro de atención.

El hecho de haber desarrollado una aplicación que maneja un tipo de datos tan sensible como lo son los sanitarios es un enfoque completamente nuevo. El planteamiento necesario desde la privacidad y la seguridad de los datos, la capacidad para que el ciudadano sienta que su información está siendo tratada y almacenada debidamente, que pueda confiar en la aplicación. Todos estos planteamientos surgen a lo largo del proyecto.

De igual modo, el hecho de haber podido investigar en profundidad un tema con tantas novedades cada mes, tan cambiante y poder hacer evolucionar el proyecto a lo largo del año a la par que todas las innovaciones tecnológicas iban apareciendo, da una visión muy clara de lo rápido que hay que adaptarse a la tecnología.

Está claro que la pandemia ha sido un antes y un después a nivel tecnológico. Han surgido muchas necesidades en todo el año y la informática ha sido clave en muchísimos aspectos. Frenar la expansión del virus, permitir compartir informes entre centros sanitarios e investigadores, monitorizar su avance... Y no solo eso, sino también acercar a las personas mediante redes sociales y videojuegos de tipo *party* o permitir a las personas seguir formándose y teletrabajar gracias a las aplicaciones de videollamadas.

También ha sido un año donde los ciberataques han llegado a duplicarse debido al incremento de tránsito de datos, sobre todo valiosos, muchos procedentes de laboratorios médicos y redes locales de casas. Es por todo ello que, en momentos donde la tecnología ha sido tan importante para todos, el enfoque en la privacidad y seguridad de sus usuarios ha de ser mayor.

Por otro lado, la experiencia de realizar parte del trabajo en equipo ha sido muy enriquecedora. He tenido la suerte de que esta haya sido junto a mi amigo Juan Velázquez García. Hemos tenido que afrontar juntos situaciones complicadas, cuadrando horarios, complementándonos en conocimientos y explicándonoslos mutuamente. El hecho de conocernos desde hace tiempo y el carácter afable de Juan, ha facilitado mucho la comunicación y ayudado a hacer más llevaderos los momentos más complicados del proyecto.

Anexo I: Contenido adjunto

El contenido adjunto a este documento incluye:

- **DatosTFGMariaRuizMolina.zip**. Que contiene los siguientes ficheros.
  - **agavaserver.zip**. Código aplicación servidor.
  - **agavaclient.zip**. Código aplicación cliente.

Anexo II: Manual de instalación

Requisitos de instalación

Se necesitan los siguientes elementos previamente a realizar la instalación de todo el proyecto.

- Dispositivo móvil con Android versión 10 y Bluetooth.
- Dispositivo diferente al anterior para despliegue del servidor.
- Acceso a una red para realizar comunicaciones por Internet.

Será necesaria la instalación de los siguientes componentes en el dispositivo donde se desplegará el servidor.

- Java JDK-8.
- NetBeans IDE 8.2 para la ejecución del servidor.
- Configuración del código en UTF-8.
- Gestor de base de datos MariaDB versión 10.5.9.
- MySQL Connector/J para realizar la conexión de la base de datos con NetBeans.
- Android Studio. Solo en caso de querer editar algún componente del código de la aplicación cliente.

Será necesaria la instalación de los siguientes componentes en el dispositivo donde se desplegará el cliente.

- Ejecutable de la aplicación.

Instalación del servidor

Se descargará e instalará NetBeans 8.2 desde https://www.oracle.com/technetwork/java/javase/downloads/jdk-netbeans-jsp-3413139-esa.html.
Esta versión incluye JDK-8, pero de querer instalar este por separado puede hacerse desde https://www.oracle.com/es/java/technologies/javase/javase-jdk8-downloads.html.
Tras ello, se descargará el gestor de base de datos MariaDB desde https://mariadb.org/download/, en concreto la versión 10.5.9.
Una vez hecho esto, es necesario instalar el conector MySQL Connector/J (descargable desde https://dev.mysql.com/downloads/connector/j/8.0.html).
Se descomprime el contenido de lo descargado. El archivo *mysql-connector-java-8.0.12.jar* se debe situar en el mismo directorio que mantiene los archivos comunes de las librerías de JAVA.
Se hace click derecho encima del nombre del proyecto y se selecciona *Properties*.
Tras ello se pulsa en *Libraries - Compile - Add Jar/Folder*. Aquí se selecciona el archivo *mysql-connector-java-8.0.12.jar* y se pulsa *Open* y tras ello *OK*.

Es importante cambiar en el código los datos referentes a usuario y contraseña por los propios.

```
conn =  DriverManager.getConnection("jdbc:mysql://192.168.0.40/employees?" +
"user=TU_USUARIO&password=TU_CONTRASEÑA");
```

Instalación del cliente

Se descargará el ejecutable de la aplicación. Tras ello se instalará y con ello la aplicación podría usarse desde el dispositivo.

En caso de querer realizar la conexión con un servidor distinto, habrá que editar en el código de la aplicación la dirección IP del servidor, así como, de ser necesario, los puertos con los que se comunicará.
Para ello sería necesario instalar un editor de Android, como por ejemplo, Android Studio. Tras realizar las pertinentes modificaciones, se selecciona la siguiente opción en el menú superior con el fin de generar un nuevo ejecutable de la aplicación.

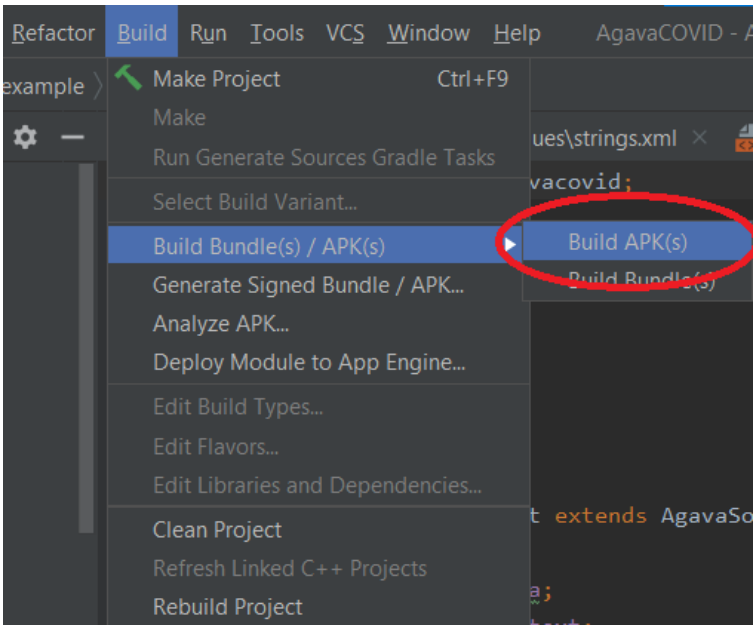


Figura 64: Build APK

Anexo III: Manual de usuario

Aplicación servidor

Para el uso de esta aplicación solo debemos tener en cuenta ciertos detalles:

- **Comprobar el estado del firewall.** En el equipo donde vayamos a ejecutar la aplicación debemos comprobar que el firewall no filtre los puertos que utiliza el programa. Esto se puede comprobar fácilmente al ejecutar el proyecto, pues a la aplicación móvil no le llegarán mensajes si efectivamente se realiza este filtrado. La solución pasa por desactivarlo mientras se lleva a cabo la ejecución. Algunos antivirus, gestionan este filtrado y permitir o no el paso según su configuración. En este caso debemos explorar nuestro antivirus y configurarlo para que permita las comunicaciones.
- **Comprobar el uso de los puertos.** También puede ocurrir que algún otro software tenga en uso los puertos utilizados por la aplicación. Recomendamos revisar si los puertos 3327, 3384 y 4445 están uso.

Para ejecutar la aplicación únicamente debemos abrir Netbeans y ejecutar el archivo *AgavaCovidServer.java* localizado en el paquete *agavacovidserver*.

Aplicación Cliente.

Bienvenid@ a AgavaCovid, tu nueva aplicación de rastreo de contactos. Antes de empezar, nos gustaría darte las gracias por confiar en nosotros para la protección de tu salud y la de tus conocidos.

¿Qué es AgavaCovid?

Como ya sabrás, AgavaCovid es una aplicación de rastreo de contactos pero, ¿eso qué quiere decir?

AgavaCovid permite conocer si has tenido contacto con otro usuario de la aplicación contagiado gracias a la tecnología Bluetooth. Al cruzarte con esa persona vuestras aplicaciones registran el uno al otro anónimamente. En caso de que uno de los dos se contagie, mediante el uso de un código proporcionado por la autoridad sanitaria el infectado podrá comunicar su contagio y la aplicación automáticamente te avisará en caso de haber estado en contacto.

Primeros pasos.

Para abrir la aplicación solo debemos pulsar en el icono con el título AgavaCovid.

Una vez abierta nos aparecerá un pequeño diálogo donde nos preguntará si queremos dar nuestro permiso para utilizar Bluetooth. Para que la aplicación funcione correctamente debemos dar a permitir. Acto seguido, si nos fijamos en la parte superior de nuestro teléfono veremos que nos aparecerá el icono de Bluetooth encendido.

Pantalla principal.

En la pantalla principal nos encontramos con cuatro elementos. En la parte superior encontramos una imagen con las mascotas de la aplicación Aga y Gava. En esta versión prototipo, al pulsar este botón realizamos el intercambio de información entre dispositivos (en la versión final esto se haría de forma automática).

Justo debajo encontramos un recuadro con nuestro estado de contagio. En caso de no estar contagiado y sin contactos contagiados, aparecerá en verde; en caso de un contacto contagiado, en naranja; y en caso de

estar contagiado, en rojo. Si pulsamos en él nos aparecerá una pantalla con unos consejos que cambiarán dependiendo de nuestro estado de contagio.

Si volvemos a la pantalla principal vemos un botón azul que dice *Comunica tu positivo*. Este botón nos da a acceso a un formulario para comunicar nuestro positivo en caso de estar contagiados.

Finalmente, en la parte de abajo encontramos tres botones con los títulos *Principal*, *Información* y *Ajustes de Idioma*. Si pulsamos sobre ellos cambiaremos de pantalla.

Pantalla de información

En la pantalla de información encontramos la política de privacidad y el compromiso con el usuario.

Pantalla de Ajustes de Idioma

En primer lugar encontramos una lista con los diferentes idiomas para los que la aplicación está disponible. Si pulsamos sobre uno de ellos veremos que se oscurecerá, significando que está seleccionado.

En la parte de abajo encontramos un botón de confirmación que al pulsar cambiará el idioma al seleccionado en la lista anterior.

¿Cómo comunico que estoy contagiado?

1. Nos colocamos en la **Pantalla Principal** y pulsamos el botón *Comunica tu positivo*.
2. En la pantalla del formulario encontramos dos campos la fecha y el código de contagio.
3. En caso de conocer alguna, introduciremos o bien la fecha de inicio de síntomas o bien la fecha de toma de muestra para diagnóstico. Para ello simplemente tendremos que pulsar sobre el campo y nos aparecerá un calendario. Ahí nos saldrán en resaltado las fechas de los últimos 14 días para poder seleccionar una de ellas. Pinchamos en una de ellas y pulsamos en *Aceptar*. Una vez seleccionemos la fecha, nos aparecerá escrita en formato *año-mes-día*.
4. Tras esto pulsamos sobre el campo *Introduzca el código*. Introducimos el número proporcionado por la autoridad sanitaria. En este caso al ser un prototipo a modo de simulación hay disponibles únicamente estos códigos válidos 123456789012, 273384273384 y 133713371337.
5. Si el código es correcto al pulsar en *Aceptar* nos saldrán dos nuevos botones para confirmar el envío. En caso de querer rectificar pulsaremos *Cancelar* y modificaremos los datos que sean necesarios. Si está todo bien pulsamos en *Aceptar*.
6. Podremos apreciar en la pantalla principal que nuestro estado de contagio habrá cambiado a *Contagiado*.

Anexo IV: Diccionario

RGPD

«El Reglamento General de Protección de Datos (RGPD) es el reglamento europeo relativo a la protección de las personas físicas en lo que respecta al tratamiento de sus datos personales y a la libre circulación de estos datos. Entró en vigor el 24 de mayo de 2016 y fue de aplicación el 25 de mayo de 2018, dos años durante los cuales las empresas, las organizaciones, los organismos y las instituciones se fueron adaptando para su cumplimiento. Es una normativa a nivel de la Unión Europea, por lo que cualquier empresa de la unión, o aquellas empresas que tengan negocios en la Unión Europea, que manejen información personal de cualquier tipo, deberán acogerse a ella. Las multas por el no cumplimiento del RGPD pueden llegar a los 20 millones de euros.»(Wikipedia. *Reglamento General de Protección de Datos*, 2021)

ENS

«En el ámbito de la Administración Electrónica española, el Esquema Nacional de Seguridad (ENS) tiene por objeto establecer la política de seguridad en la utilización de medios electrónicos y está constituido por principios básicos y requisitos mínimos que permitan una protección adecuada de la información. Dicho esquema se regula en Real Decreto 3/2010, de 8 de enero, y fue establecido anteriormente en el artículo 42 de la Ley 11/2007, de 22 de junio, de acceso electrónico de los ciudadanos a los Servicio Públicos, que fue modificado por el Real Decreto 951/2015 para actualizarlo a la luz de la experiencia obtenida en su implantación, de la evolución de la tecnología y las ciberamenazas y del contexto regulatorio internacional y europeo.» (Wikipedia. *Esquema Nacional de Seguridad*, 2021)

COVID-19

«La enfermedad por coronavirus de 2019, más conocida como COVID-19 es una enfermedad infecciosa causada por el virus SARS-CoV-2. Produce síntomas similares a los de la gripe o catarro, entre los que se incluyen fiebre, tos, disnea, mialgia y fatiga. En casos graves se caracteriza por producir neumonía, síndrome de dificultad respiratoria aguda, sepsis y choque séptico que conduce a cerca de 3,75 % de los infectados a la muerte según la OMS.18» (Wikipedia. *COVID-19*, 2021)

Identificador efímero

Elemento que se emplea para identificar al usuario unívocamente. Son únicos con el fin de evitar colisiones. Son generados de manera aleatoria a partir de unas semillas.

Su duración está delimitada por un determinado periodo de tiempo y es sucedido por otro. Esto es así porque en el contexto de esta aplicación, es necesario determinar el periodo temporal en el cual se ha mantenido el contacto entre dos usuarios. De esta manera se dificulta el seguimiento de un usuario ya que el identificador cambia transcurrido dicho tiempo.

En el tipo de aplicaciones como la desarrollada en este trabajo, es fundamental que los identificadores no revelen información personal y/o privada de los usuarios.

SDK

«Un kit de desarrollo de software (en inglés, software development kit o SDK) es generalmente un conjunto de herramientas de desarrollo de software que permite a un desarrollador de software crear una aplicación

informática para un sistema concreto, por ejemplo ciertos paquetes de software, entornos de trabajo, plataformas de hardware, computadoras, videoconsolas, sistemas operativos, etcétera.» (Wikipedia. *Kit de desarrollo de software*, 2020)

IMEI

IMEI significa International Mobile Equipment Identity, y es un identificador único que tiene cada teléfono móvil. El código consta de cuatro partes: TAC o Type Allocation Code (los primeros dos indican el RBI o Reporting Body Identifier, es decir, la organización encargada de regular el teléfono), FAC o Final Assembly Code (indica el fabricante), Número de serie, Código verificador (verifica que el código sea correcto y no haya habido errores).

Dirección MAC

«En las redes de computadoras, la dirección MAC (siglas en inglés de Media Access Control) es un identificador de 48 bits (6 bloques de dos caracteres hexadecimales [8 bits]) que corresponde de forma única a una tarjeta o dispositivo de red. Se la conoce también como dirección física, y es única para cada dispositivo. Está determinada y configurada por el IEEE (los últimos 24 bits) y el fabricante (primeros 24 bits)». (Wikipedia. *Dirección MAC*, 2021)

Dirección MAC de Bluetooth

Se trata de la dirección identificadora de cada dispositivo para entablar conexiones BlueTooth. Estas están conformadas por 12 caracteres hexadecimales.

Bluetooth

«Bluetooth es una especificación industrial para redes inalámbricas de área personal (WPAN) creado por Bluetooth Special Interest Group, Inc. que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2.4 GHz. Los principales objetivos que se pretenden conseguir con esta norma son:

- Facilitar las comunicaciones entre equipos móviles.
- Eliminar los cables y conectores entre estos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre equipos personales.

Los dispositivos que con mayor frecuencia utilizan esta tecnología pertenecen a sectores de las telecomunicaciones y la informática personal, como teléfonos móviles, computadoras portátiles [...] o cámaras digitales». (Wikipedia. *Bluetooth*, 2021)

WiFi

«El wifi (escrito también wi fi) es una tecnología que permite la interconexión inalámbrica de dispositivos electrónicos. Los dispositivos habilitados con wifi (tales como ordenadores personales, teléfonos, televisores, videoconsolas, reproductores de música, etcétera) pueden conectarse entre sí o a Internet a través de un punto de acceso de red inalámbrica». (Wikipedia. *Wifi*, 2021)

IP

«La dirección IP es un conjunto de números que identifica, de manera lógica y jerárquica, a una interfaz en la red (elemento de comunicación/conexión) de un dispositivo (computadora, laptop, teléfono inteligente) que utilice el protocolo (Internet Protocol) o, que corresponde al nivel de red del modelo TCP/IP. La dirección IP no debe confundirse con la dirección MAC». (Wikipedia. *Dirección IP*, 2021)

BLE/BlueTooth Low Energy

«Bluetooth Low Energy (Bluetooth LE, coloquialmente BLE) es una tecnología de red de área personal [...] destinada a aplicaciones novedosas en el cuidado de la salud, fitness y beacons, seguridad y las industrias de entretenimiento en el hogar. Comparado con el Bluetooth clásico, Bluetooth Low Energy está diseñado para proporcionar un bajo consumo de energía, manteniendo un rango de alcance de comunicación similar». (Wikipedia. *Bluetooth de baja energía*, 2020)

Semillas generadoras

Estas consisten en dos elementos, la clave generadora, y la fecha generadora.

Claves generadoras

Consiste en una clave generada a partir de una secuencia binaria, la cual es subdividida en n claves generadoras que se emplean a lo largo del día. Dicha secuencia es obtenida de manera pseudoaleatoria a partir de metodologías criptográficas. Estas claves se emplean para generar cada uno de los identificadores efímeros.

Fechas generadoras

Es la fecha que indica el fragmento a seleccionar de la clave generadora. Estas fechas siempre van de 15 en 15 minutos desde las 00:00:00. Cada 15 minutos indica se selecciona el fragmento siguiente a utilizar, el cual determina el identificador efímero de ese periodo de tiempo.

Paradigma

«Para la Ingeniería de Software el paradigma es una agrupación de métodos, herramientas y procedimientos con el fin de describir un modelo.» (Heli Sulbaran Sistemas. *Paradigmas en el desarrollo de software*, 2014)

Android

«Android es un sistema operativo móvil basado en núcleo Linux y otros software de código abierto. Fue diseñado para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tabletas, relojes inteligentes (Wear OS), automóviles (Android Auto) y televisores (Android TV)». (Wikipedia. *Android*, 2021)

iOS

«iOS es un sistema operativo móvil de la multinacional Apple Inc. Originalmente desarrollado para el iPhone (iPhone OS), después se ha usado en dispositivos como el iPod touch y el iPad. Apple no permite la

instalación de iOS en hardware de terceros». (Wikipedia. *iOS*, 2021)

Unix Epoch Time

Unix Epoch Time es el nombre que recibe las 00:00:00 UTC del 1 de enero de 1970, que es la fecha que la mayoría de dispositivos informáticos toman como referencia para empezar a contar el tiempo. Para ello se utiliza el número de milisegundos que han transcurrido desde esta fecha.

AES

«Advanced Encryption Standard (AES), también conocido como Rijndael (pronunciado Rain Doll.[en] inglés), es un esquema de cifrado por bloques adoptado como un estándar de cifrado por el gobierno de los Estados Unidos, creado en Bélgica. [...] El cifrado fue desarrollado por dos criptólogos belgas, Joan Daemen y Vincent Rijmen. [...] AES tiene un tamaño de bloque fijo de 128 bits y tamaños de llave de 128, 192 o 256 bits. [...] La mayoría de los cálculos del algoritmo AES se hacen en un campo finito determinado. AES opera en una matriz de 4×4 bytes, llamada state». (Wikipedia. *Advanced Encryption Standard*, 2021)

Clave AES

Se tratan de claves simétricas empleadas en el algoritmo criptográfico por bloques AES.

Dependiendo del algoritmo AES utilizado, AES-128, AES-192 o AES-256, la longitud de esta será de 128, 192 o 256 bits respectivamente (como indica el nombre de la variante AES).

Durante el cifrado, cuando el texto original es más largo que la clave y cifrados los bloques que abarcaba la longitud de la clave, esta se expande empleando una serie de operaciones con el fin de cifrar los bloques restantes del texto original. Esta expansión se realiza mediante operaciones de rotación, sustitución y XOR.

PUID

Identificador de 128 bits generado de manera pseudoaleatoria en el protocolo PEPP-PT/PEPP. El servidor proporciona uno a cada nuevo usuario registrado con el fin de identificarlos de manera única.

EBID

Un EBID o Ephemeral Bluetooth ID es la implementación de identificador efímero que utiliza el protocolo PEPP-PT/PEPP.

Se trata de identificadores que se intercambian entre usuarios vía BlueTooth con el fin de registrar con quién se ha mantenido contacto.

A diferencia de los PUIDs, como puede verse, los EBIDs sirven para identificar a los usuarios entre sí, mientras que los PUIDs lo hacen frente al servidor.

HMAC-SHA256

Se trata de un código de autentificación de mensajes en clave hash que utiliza para calcular dichos resúmenes hash SHA-256. Puede servir para comprobar la integridad de los datos (estos no han sido modificados), la

autenticación del mensaje (el emisor es quien dice ser) o la generación pseudoaleatoria de cadenas de bits (al generar difusión y confusión en la transformación de la entrada).

En el caso de DP-3T, este algoritmo se emplea para generar de manera pseudoaleatoria parte del identificador secreto S_EphID(BK).

Broadcast

«En Informática, la difusión amplia, difusión ancha o broadcast, es una forma de transmisión de información donde un nodo emisor envía información a una multitud de nodos receptores de manera simultánea, sin necesidad de reproducir la misma transmisión nodo por nodo». (Wikipedia. *Difusión amplia*, 2020)

IV

«En criptografía, un vector de inicialización (conocido por sus siglas en inglés IV) es un bloque de bits que es requerido para permitir un cifrado en flujo o un cifrado por bloques, en uno de los modos de cifrado, con un resultado independiente de otros cifrados producidos por la misma clave. El tamaño del IV dependen del algoritmo de cifrado y del protocolo criptográfico y a menudo es tan largo como el tamaño de bloque o como el tamaño de la clave.» (Wikipedia. *Vector de inicialización*, 2019)

Upload What You Observed

Paradigma utilizado en protocolos de rastreo de contactos donde los datos que se envía al servidor son los identificadores recibidos mediante intercambios. De este modo el servidor envía a los clientes una lista de identificadores que han estado en contacto con un contagiado. Así, cada cliente compara la lista recibida con sus propios identificadores.

Upload What You Sent

Paradigma empleado en protocolos de rastreo de contactos donde los datos enviados al servidor son los identificadores del propio usuario. De este modo, el servidor envía a los clientes una lista de identificadores contagiados, la cual cada cliente compara con aquellos que ha recibido.

PSI-CA

La PSI-CA o Private Set Intersection Cardinality es una técnica criptográfica de cálculo multiparte segura (o protocolo de preservación de privacidad) que permite a un emisor y a un receptor computar la cardinalidad de la intersección entre sus conjuntos sin revelar más información al otro. De esta forma se preserva la privacidad del resto de información relacionada, pues solo se revelan los elementos contenidos en la intersección. [58]

Firmas Digitales Ciegas

«La firma digital ciega es un protocolo de firma digital que permite a una persona obtener un mensaje con una firma o sello otorgados por otra entidad para que pueda ser presentada ante terceros, sin necesidad de revelarle a esta información del contenido específico del mensaje.

La principal motivación que tuvo su creador David Chaum fue que cada vez que se llama por teléfono, se compra un producto usando una tarjeta de crédito, se suscribe a una revista o paga algún impuesto, esa

información va a parar a una base de datos en algún lugar, lo que trasgrede nuestro derecho a privacidad». (Wikipedia. *Firma digital ciega*, 2020)

Diffie-Hellman Algorithm

Se trata del primer algoritmo de clave pública, creado en 1976 por W. Diffie y M. Hellman. Se emplea para la distribución de claves y no de mensajes largos debido a que su coste computacional aumenta con el tamaño del mensaje a cifrar. Se aprovecha de la dificultad para calcular logaritmos discretos en un campo finito y emplea funciones matemáticas de la forma

$$g^a mod(p)$$

donde p es un número primo grande y a un entero.

Handshake

«El establecimiento de comunicación (del inglés handshake, literalmente apretón de manos) es utilizado en tecnologías informáticas, telecomunicaciones, y otras conexiones para establecer automáticamente una negociación entre pares que establece de forma dinámica los parámetros de un canal de comunicación entre ellos antes de que comience la comunicación normal por el canal. De ello se desprende la creación física del canal y precede a la transferencia de información normal.» (Wikipedia. *Establecimiento de comunicación*, 2021)

Entropía de Shannon

La entropía de Shannon mide la incertidumbre de una fuente de información. Considera la cantidad de información promedio que contienen los caracteres empleados. Los símbolos con menor probabilidad de aparición son los que aportan mayor información, mientras que aquellos más frecuentes aportan menor información. Cuando todos los símbolos poseen la misma frecuencia de aparición, todos aportan información relevante y por lo tanto la entropía es máxima.

En este proyecto se ha utilizado para detectar cuándo esos datos son más o menos estructurados. El máximo es 8, representando datos no estructurados "aleatorios". Datos encriptados o que corresponden a un resumen hash deben poseer una entropía superior a 7.5.

TOR

«Tor (sigla de The Onion Router [...]) Es un proyecto cuyo objetivo principal es el desarrollo de una red de comunicaciones distribuida de baja latencia y superpuesta sobre internet, en la que el encaminamiento de los mensajes intercambiados entre los usuarios no revela su identidad, es decir, su dirección IP (anonimato a nivel de red) y que, además, mantiene la integridad y el secreto de la información que viaja por ella». (Wikipedia. *Tor (red de anonimato)*, 2021)

Nerd-attack

Aunque las aplicaciones DP-3T están hechas para recolectar la mínima cantidad de los datos, el código es abierto. Un *Nerd-attack* consiste en que un atacante puede elaborar sus propios clientes DP-3T donde se recolecten más datos como la geolocalización o información del mensaje BlueTooth.

Militia-attack

Se trata de un *Nerd-attack* con más fases, consistentes en vender los datos recolectados, sobre todo los que identifiquen a los infectados, a milicias organizadas.

Ataque de Paparazzi

Ataque consistente en la recolección de información que viaja de la aplicación al servidor y viceversa con el fin de encontrar identificadores infectados de personajes públicos. En definitiva es un ataque de escucha algo más sofisticado, donde el objetivo está predefinido.

Ataque inyección de código

Consiste en la inserción de código malicioso que modifica el software para otros fines no intencionados. Dicha inserción se aprovecha de vulnerabilidades en entradas de datos, o accesos a puertos mal protegidos del servidor, para provocar un desbordamiento de pila, lo que permite acceder a zonas indebidas del almacenamiento o código del sistema.

Ataque de enlace/Phishing

«Consiste en la emisión masiva de correos electrónicos a usuarios. Estos correos suplantan a entidades de confianza (ejemplo bancos) y persiguen el engaño del usuario y la consecución de información. Por ejemplo en el mensaje se incluyen enlaces a dominios maliciosos. Para camuflar estos enlaces es habitual que el texto del enlace sea la URL correcta, pero el enlace en sí apunte al sitio malicioso». (Wikipedia. *Phishing*, 2020)

En el tipo de aplicaciones como la desarrollada en este trabajo, el objetivo es evitar que dichos enlaces maliciosos sean insertados en la aplicación o dispuestos al usuario mediante otras vías haciéndose pasar por entidades sanitarias.

Objeto

«En el paradigma de programación orientada a objetos (POO, o bien OOP en inglés), un objeto es un ente orientado a objetos (programa de computadoras) que consta de un estado y de un comportamiento, que a su vez constan respectivamente de datos almacenados y de tareas realizables durante el tiempo de ejecución. Un objeto puede ser creado instanciando una clase, como ocurre en la programación orientada a objetos, o mediante escritura directa de código y la replicación de otros objetos, como ocurre en la programación basada en prototipos». (Wikipedia. *Objeto (programación)*, 2021)

HTTP

«HTTP, de sus siglas en inglés: "Hypertext Transfer Protocol", es el nombre de un protocolo el cual nos permite realizar una petición de datos y recursos, como pueden ser documentos HTML. Es la base de cualquier intercambio de datos en la Web, y un protocolo de estructura cliente-servidor, esto quiere decir que una petición de datos es iniciada por el elemento que recibirá los datos (el cliente), normalmente un navegador Web.

Clientes y servidores se comunican intercambiando mensajes individuales. [...] Los mensajes que envía el cliente, normalmente un navegador Web, se llaman peticiones, y los mensajes enviados por el servidor se llaman respuestas». (Mozilla. *Generalidades del protocolo HTTP*, 2020)

Protanopia

«La protanopia es la carencia de sensibilidad al color rojo, una disfunción visual relacionada con la percepción del color. Se denomina también dicromacia roja. [...] Por tanto, los individuos que sufren protanopia padecen una pérdida clara de sensibilidad a la luminosidad del extremo rojo del espectro cromático». (Wikipedia. *Protanopia*, 2020)

Deuteranopia

«La deuteranopia o deuteranopsia es una disfunción visual consistente en alteración para la percepción del color.

Los conos de la retina responsables de la recepción de luz con longitud de onda correspondiente al color verde están ausentes o no son funcionales. Por tanto existe una deficiencia a la hora de discriminar entre verde y rojo». (Wikipedia. *Deuteranopia*, 2019)

Tritanopia

«La tritanopia es una disfunción visual relacionada con la percepción del color.

Consiste en la carencia de sensibilidad al color azul, denominada también dicromacia azul.

Se trata de una de las alteraciones de la visión cromática menos frecuentes». (Wikipedia. *Tritanopia*, 2020)

Bibliografía

[1] DP-3T. DP-3T/dp3t-sdk-android. URL: https://github.com/DP-3T/documents/blob/master/Security%5C%20analysis/PEPP-PT_%5C%20Data%5C%20Protection%5C%20Architechture%5C%20-%5C%20Security%5C%20and%5C%20privacy%5C%20analysis.pdf (visitado 16-04-2021).

[2] DP-3T. DP-3T/dp3t-sdk-android. URL: https://github.com/DP-3T/dp3t-sdk-android/blob/master/dp3t-sdk/sdk/src/main/java/org/dpppt/android/sdk/InfectionStatus.java (visitado 16-04-2021).

[3] DP-3T. DP-3T/dp3t-sdk-android. URL: https://github.com/DP-3T/documents/blob/master/DP3T%5C%20-%5C%20Exposure%5C%20Score%5C%20Calculation.pdf (visitado 26-04-2021).

[4] DP-3T. DP-3T/dp3t-sdk-android. URL: https://github.com/DP-3T/dp3t-sdk-android/tree/feaf563eb39d1d8c9416f718a81a574b25fa8384 (visitado 16-04-2021).

[5] DP-3T. DP3T-Backend-SDK. URL: https://github.com/DP-3T/dp3t-sdk-backend (visitado 16-04-2021).

[6] Albert Agustinoy. Francia alinea su normativa nacional con el RGPD. Feb. de 2018. URL: https://blog.cuatrecasas.com/propiedad-intelectual/francia-alinea-normativa-nacional-rgpd/ (visitado 10-08-2021).

[7] Nadeem Ahmed y col. "A Survey of COVID-19 Contact Tracing Apps". En: IEEE Access 8 (2020), págs. 134577-134601. DOI: 10.1109/ACCESS.2020.3010226. URL: https://doi.org/10.1109/ACCESS.2020.3010226 (visitado 29-03-2021).

[8] Gennaro Avitabile y col. "Towards Defeating Mass Surveillance and SARS-CoV-2: The Pronto-C2 Fully Decentralized Automatic Contact Tracing System". En: IACR Cryptol. ePrint Arch. 2020 (2020), pág. 493. URL: https://eprint.iacr.org/2020/493 (visitado 23-04-2021).

[9] Wasilij Beskorovajnov y col. "ConTra Corona: Contact Tracing against the Coronavirus by Bridging the Centralized - Decentralized Divide for Stronger Privacy". En: IACR Cryptol. ePrint Arch. 2020 (2020), pág. 505. URL: https://eprint.iacr.org/2020/505 (visitado 28-01-2021).

[10] Bluetooth overview nbsp;: nbsp; Android Developers. URL: https://developer.android.com/guide/topics/connectivity/bluetooth.html (visitado 17-05-2021).

[11] BlueTrace. [Online]. Feb. de 2021. URL: https://en.wikipedia.org/wiki/BlueTrace (visitado 16-04-2021).

[12] BlueTrace Controversy. [Online]. Feb. de 2021. URL: https://en.wikipedia.org/wiki/BlueTrace#Controversy (visitado 16-04-2021).

[13] European Data Protection Board. "Directrices 04/2020 sobre el uso de datos de localización y herramientas de rastreo de contactos en el contexto de la pandemia de COVID-19". En: (dic. de 2020). URL: https://edpb.europa.eu/sites/edpb/files/files/file1/edpb_guidelines_20200420_contact_tracing_covid_with_annex_es.pdf (visitado 02-01-2021).

[14] CNIL. Applications mobiles en santé et protection des données personnelles: Les questions à se poser. [Online]. Ago. de 2018. URL: https://www.cnil.fr/fr/applications-mobiles-en-sante-et-protection-des-donnees-personnelles-les-questions-se-poser (visitado 11-08-2021).

[15] CNIL. Commission Nationale de l'Informatique et des Libertés. [Online]. URL: https://www.cnil.fr/ (visitado 26-08-2021).

[16] CNIL. Privacy Impact Assessment (PIA). [Online]. Feb. de 2018. URL: https://www.cnil.fr/en/PIA-privacy-impact-assessment-en (visitado 24-08-2021).

[17] CNIL. Privacy Impact Assessment (PIA) - Methodology. [Online]. Feb. de 2018. URL: https://www.cnil.fr/sites/default/files/atoms/files/cnil-pia-1-en-methodology.pdf (visitado 24-08-2021).

[18] CNIL. Publication of the CNIL's opinion on the StopCovid mobile application project. [Online]. Abr. de 2020. URL: https://www.cnil.fr/en/publication-cnils-opinion-stopcovid-mobile-application-project (visitado 12-08-2021).

[19] CEF Digital - European Commission. Experience the future with the European Blockchain Services Infrastructure. [Online]. 2020. URL: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/EBSI (visitado 08-08-2021).

[20] Conversor de salario. [Online]. 2021. URL: https://es.talent.com/convert?salary=9&start=hour&end=year&hw=37.5 (visitado 03-08-2021).

[21] COVID Credentials Initiative. [Online]. 2020. URL: https://www.covidcreds.org/ (visitado 13-08-2021).

[22] Cuándo, para qué y por qué utilizar MariaDB. [Online]. Ene. de 2018. URL: https://www.arsys.es/blog/programacion/mariadb/ (visitado 07-04-2021).

[23] Paul Dalg y col. "Das gefährliche Chaos um die Corona-App". En: Tagesspiegel (abr. de 2020). URL: https://www.tagesspiegel.de/wissen/welche-technologie-soll-es-sein-das-gefaehrliche-chaos-um-die-corona-app/25755338.html (visitado 23-03-2021).

[24] Decentralized Privacy-Preserving Proximity Tracing. [Online]. Dic. de 2020. URL: https://en.wikipedia.org/wiki/Decentralized_Privacy-Preserving_Proximity_Tracing (visitado 16-04-2021).

[25] Definition of IP multicast. URL: https://www.pcmag.com/encyclopedia/term/ip-multicast (visitado 19-04-2021).

[26] Paul-Olivier Dehaye y Joel Reardon. "SwissCovid: a critical analysis of risk assessment by Swiss authorities". En: CoRR abs/2006.10719 (2020). arXiv: 2006.10719. URL: https://arxiv.org/abs/2006.10719 (visitado 09-11-2020).

[27] Desconocido. "Den Tracing-App-Entwicklern laufen die Partner weg". En: Spiegel Netzwelt (abr. de 2020). URL: https://www.spiegel.de/netzwelt/apps/pepp-pt-in-corona-krise-den-tracing-app-entwicklern-laufen-die-partner-weg-a-017f50eb-c1e2-4097-8182-53708ca6db59 (visitado 23-03-2021).

[28] Difference with Apple/Google solution. [Online]. Nov. de 2020. URL: https://github.com/DP-3T/documents/issues/128 (visitado 19-04-2021).

[29] European Commission - DIGIT. About SSI eIDAS Bridge. [Online]. Mar. de 2021. URL: https://joinup.ec.europa.eu/collection/ssi-eidas-bridge/about (visitado 07-08-2021).

[30] DP3T - Decentralized Privacy-Preserving Proximity Tracing. [Online]. Dic. de 2020. URL: https://github.com/DP-3T/documents (visitado 16-04-2021).

[31] Exposure Notification. [Online]. Nov. de 2020. URL: https://en.wikipedia.org/wiki/Exposure_Notification (visitado 15-04-2021).

[32] Gouvernement Français. Application TousAntiCovid. [Online]. Mayo de 2021. URL: https://www.gouvernement.fr/info-coronavirus/tousanticovid (visitado 15-08-2021).

[33] France24. Así funciona StopCovid, la app para rastrear contagios que causa polémica en Francia. [Online]. Mayo de 2020. URL: https://www.france24.com/es/20200527-stopcovid-polemica-francia-app-rastreo-contagios (visitado 15-08-2021).

[34] GDPR Compliance for Apps. Ene de 2021. URL: https://www.privacypolicies.com/blog/gdpr-compliance-apps/ (visitado 09-08-2021).

[35] Carlos Alonso González. Introducción a la Minería de Datos. https://aulas.inf.uva.es/pluginfile.php/41079/mod_resource/content/6/01IntroduccionMD.pdf. Accessed: 2021–03-03. 2020.

[36] Lisa Hegemann. "Wissenschaftler warnen vor "beispielloser Überwachung"". En: Zeit (abr. de 2020). URL: https://www.zeit.de/digital/datenschutz/2020-04/corona-app-initiative-pepp-pt-datenschutz-warnung-forscher (visitado 23-03-2021).

[37] Alex Hern. "Digital contact tracing will fail unless privacy is respected, experts warn". En: The Guardian (abr. de 2020). URL: https://www.theguardian.com/world/2020/apr/20/coronavirus-digital-contact-tracing-will-fail-unless-privacy-is-respected-experts-warn (visitado 17-11-2020).

[38] Wim Hoogenraad. Beacon Apps GDPR, privacy is paramount. [Online]. Oct. de 2020. URL: https://en.itpedia.nl/2018/06/13/beacon-apps-gdpr-privacy-staat-bovenaan/ (visitado 12-08-2021).

[39] How COVID-19 Challenged Our Views on Privacy. Jun. de 2020. URL: https://www.eticasconsulting.com/how-covid-19-challenged-our-views-on-privacy/ (visitado 18-08-2021).

[40] Mike Cotterell . Bob Hughes. ""Software Project Management", Fifth Edition, Tata McGraw Hill, 2004." En: 2015.

[41] Simon Hurtz. "Der Anti-Corona-App droht ein Glaubenskrieg unter Forschern". En: Süddeutsche Zeitung (abr. de 2020). URL: https://www.sueddeutsche.de/digital/coronavirus-pepp-pt-dp-3t-smartphone-app-streit-1.4882612 (visitado 23-03-2021).

[42] Introducción general a Bluetooth. [Online]. Nov. de 2020. URL: https://developer.android.com/guide/topics/connectivity/bluetooth?hl=es-419 (visitado 30-05-2021).

[43] A. Tan; C. Sheng Hau; L. Yongquan; J. Tan J. Bay; J. Kek y T. Anh Quy. "BlueTrace: A privacy-preserving protocol forcommunity-driven contact tracing across borders". En: (dic. de 2020). URL: https://bluetrace.io/static/bluetrace_whitepaper-938063656596c104632def383eb33b3c.pdf (visitado 25-01-2021).

[44] Bobbie Johnson. "Some prominent exposure apps are slowly rolling back freedoms". En: (nov. de 2020). URL: https://2020.internethealthreport.org/slideshow-internet-health/#16 (visitado 13-11-2020).

[45] Sebastian Klöckner. Joint Statement on Contact Tracing. Abr. de 2020. URL: https://cispa.de/en/news-and-events/news-archive/articles/2020/joint-statement-on-contact-tracing (visitado 24-03-2021).

[46] Légifrance. Article L1110-4-1 (abrogé). [Online]. Mayo de 2021. URL: https://www.legifrance.gouv.fr/codes/article_lc/LEGIARTI000038886960/ (visitado 11-08-2021).

[47] Légifrance. Article L1111-24. [Online]. Jul. de 2010. URL: https://www.legifrance.gouv.fr/codes/article_lc/LEGIARTI000020892842/ (visitado 11-08-2021).

[48] Ricardo Oliva León. Monedero digital de identidad europea y propuesta de Reglamento eIDAS 2. Jul. de 2021. URL: https://www.algoritmolegal.com/entorno-juridico-internet/reglamento-eidas-2/ (visitado 07-08-2021).

[49] N. Lomas. "Norway pulls it's coronavirus contacts tracing app after privacy watchdogs warning". En: (dic. de 2020). URL: https://techcrunch.com/2020/06/15/norway-pulls-its-coronavirus-contacts-tracing-app-after-privacy-watchdogs-warning/?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAANqd1ugQ71yUiD22OjcVdJtVmSUPfuVEqULuAwgeXtLYz3ij5XYlOX8ZseqYHiE1Ct4Af9h2mm061Tar2KJKokRTfuejJAwNkdt8-1LMvTajRzId8N4ptyTw4X1Aa-O7nN7KrQIiC3v6 (visitado 29-12-2020).

[50] MariaDB. [Online]. Mar. de 2021. URL: https://es.wikipedia.org/wiki/MariaDB (visitado 07-04-2021).

[51] Rafael Marín. Los gestores de bases de datos más usados en la actualidad. URL: https://revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/ (visitado 07-04-2021).

[52] Mobile databases: SQLite and SQLite alternatives for Android and iOS. [Online]. Dic. de 2020. URL: https://greenrobot.org/news/mobile-databases-sqlite-alternatives-and-nosql-for-android-and-ios/ (visitado 04-04-2021).

[53] MySQL. [Online]. Ene. de 2021. URL: https://es.wikipedia.org/wiki/MySQL (visitado 04-04-2021).

[54] MySQL, MariaDB y PostgreSQL: ¿Cuál elegimos? [Online]. Jun. de 2020. URL: https://www.arsys.es/blog/mysql-mariadb-postgresql/ (visitado 04-04-2021).

[55] Patrick Howell O'Neill. Norway halts coronavirus app over privacy concerns. Nov. de 2020. URL: https://www.technologyreview.com/2020/06/15/1003562/norway-halts-coronavirus-app-over-privacy-concerns/ (visitado 15-12-2020).

[56] Pan-European Privacy-Preserving Proximity Tracing. [Online]. Dic. de 2020. URL: https://en.wikipedia.org/wiki/Pan-European_Privacy-Preserving_Proximity_Tracing (visitado 20-04-2021).

[57] President von der Leyen's speech at the high-level opening session of the 2021 Digital Assembly. [Online]. Jun. de 2021. URL: https://ec.europa.eu/commission/presscorner/detail/en/speech_21_2804 (visitado 11-08-2021).

[58] Protocolo seguro. [Online]. Mar. de 2021. URL: https://es.wikipedia.org/wiki/Protocolo_seguro (visitado 20-03-2021).

[59] Reglamento General de Protección de Datos. [Online]. Abr. de 2021. URL: https://es.wikipedia.org/wiki/Reglamento_General_de_Protecci%5C%C3%5C%B3n%5C_de%5C_Datos (visitado 16-04-2021).

[60] Timothy Ruff. The EU Announcement Is the Biggest Ever in SSI. [Online]. Jun. de 2021. URL: https://credentialmaster.com/the-eu-announcement-is-the-biggest-ever-in-ssi/ (visitado 11-08-2021).

[61] José Luis Sevillano y col. "Soft real-time communications over Bluetooth under interferences from ISM devices". En: Int. J. Commun. Syst. 19.10 (2006), págs. 1103-1116. DOI: 10.1002/dac.796. URL: https://doi.org/10.1002/dac.796 (visitado 22-05-2021).

[62] Mark Surman. Privacy Norms and the Pandemic. Abr. de 2020. URL: https://blog.mozilla.org/blog/2020/04/22/privacy-norms-and-the-pandemic/ (visitado 27-04-2021).

[63] Ni Trieu y col. "Epione: Lightweight Contact Tracing with Strong Privacy". En: IEEE Data Eng. Bull. 43.2 (2020), págs. 95-107. URL: http://sites.computer.org/debull/A20june/p95.pdf (visitado 21-04-2021).

[64] Ni Trieu y col. "Epione: Lightweight Contact Tracing with Strong Privacy". En: ArXiv abs/2004.13293 (2020). (Visitado 25-04-2021).

[65] Ni Trieu y col. Epione: Lightweight Contact Tracing with Strong Privacy. URL: https://sunblaze-ucb.github.io/privacy/projects/epione.html (visitado 25-04-2021).

[66] Diario Oficial de la Unión Europea. Derecho a la limitación del tratamiento. [Online]. Abr. de 2016. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=ES#d1e2763-1-1 (visitado 20-08-2021).

[67] Diario Oficial de la Unión Europea. Derecho de acceso del interesado. [Online]. Abr. de 2016. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=ES#d1e2576-1-1 (visitado 19-08-2021).

[68] Diario Oficial de la Unión Europea. Derecho de oposición. [Online]. Abr. de 2016. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=ES#d1e2866-1-1 (visitado 20-08-2021).

[69] Diario Oficial de la Unión Europea. Derecho de rectificación. [Online]. Abr. de 2016. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=ES#d1e2662-1-1 (visitado 20-08-2021).

[70] Diario Oficial de la Unión Europea. Evaluación de impacto relativa a la protección de datos. [Online]. Abr. de 2016. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=ES#d1e3606-1-1 (visitado 20-08-2021).

[71] Diario Oficial de la Unión Europea. Principio de transparencia. [Online]. Abr. de 2016. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=ES#d1e40-1-1 (visitado 20-08-2021).

[72] Diario Oficial de la Unión Europea. Pruebas y requisitos para el consentimiento. [Online]. Abr. de 2016. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=ES#d1e40-1-1 (visitado 19-08-2021).

[73] Diario Oficial de la Unión Europea. REGLAMENTO (UE) 2016/679 DEL PARLAMENTO EUROPEO Y DEL CON [Online]. Abr. de 2016. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=EN#d1e40-1-1 (visitado 19-08-2021).

[74] Diario Oficial de la Unión Europea. Seguridad del tratamiento. [Online]. Abr. de 2016. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=ES#d1e3443-1-1 (visitado 20-08-2021).

[75] Serge Vaudenay. "Analysis of DP3T". En: IACR Cryptol. ePrint Arch. 2020 (2020), pág. 399. URL: https://eprint.iacr.org/2020/399 (visitado 27-04-2021).

[76] Serge Vaudenay. "Centralized or Decentralized? The Contact Tracing Dilemma". En: IACR Cryptol. ePrint Arch. 2020 (2020), pág. 531. URL: https://eprint.iacr.org/2020/531 (visitado 27-04-2021).

[77] W3C. Verifiable Credentials Data Model 1.0. [Online]. Nov. de 2019. URL: https://www.w3.org/TR/vc-data-model/ (visitado 06-08-2021).

[78] Matthew Wickline y the Human-Computer Interaction Resource Network. "Color Blind Simulation". En: (- de 2001). URL: https://www.color-blindness.com/coblis-color-blindness-simulator/ (visitado 06-07-2021).