Universidad de Valladolid

E.U. de Informática (Segovia)

Grado en Ingeniería Informática de Servicios y Aplicaciones

Simulador en Java de distintos sistemas de cifrado

Alumno: Pablo Pérez Corral

Tutor: José Ignacio Farrán Martín

"Si solo viajas por los caminos marcados solo verás paisajes que ya se conocen.
Si trabajas igual que los demás solo obtendrás resultados que ya existen.
Si solo haces lo que ya sabes, nunca aprenderás nada."

— Anonimo

"El aprendizaje es experiencia. Todo lo demás es información."

— Albert Einstein

Agradecimientos:

Al conjunto de profesores del universidad de informática por guiarme en mayor o menor medida a llegar hasta aquí; especialmente a Jose Ignacio Farrán, mi tutor de proyecto; y a Miguel Angel Martínez Prieto, que me apoyó en varias asignaturas y me impulsó a probar cosas diferentes en el campo de la informática y no darme por vencido.

Índice

1. Descripción del Proyecto

1.1. Identificación del Proyecto

Para ilustrar el proyecto, debemos comenzar por definir el concepto de cifrado de datos y su utilidad. El cifrado de datos consiste en alterar un mensaje antes de ser transmitido, de modo que su contenido solo sea legible para quienes posean cierta clave o método concreto para su correcta visualización o entendimiento. De ese modo, solo las personas a quienes vaya dirigido dicho mensaje podrán entender su contenido, ya que contarán de antemano con la clave necesaria para descifrarlo.

La información enviada a través de Internet no es totalmente segura, y puede ser interceptada por intrusos, el cifrado de datos es una alternativa para mantener la privacidad y seguridad de esa información. Por tanto, este concepto puede aplicarse en las distintas actividades que un usuario realiza diariamente. Al navegar por Internet, algunos sitios web ofrecen una conexión cifrada, de modo que los datos que se envían desde y hacia el sitio web se encuentran cifrados, y son ilegibles para intrusos que puedan interceptar la comunicación.

Sin embargo, no todos estos protocolos de seguridad son 100% seguros. A pesar de haberse establecido ciertos estándares, una mala implementación en un sistema que realice el cifrado y envío de datos de forma automatizada puede descubrirse insegura si un posible atacante es capaz de interceptar o incluso modificar los datos en mitad del proceso. Algunas plataformas o aplicaciones famosas que se enfrentan regularmente a este tipo de noticias son por ejemplo Facebook, Whatsapp, etc.

Además, por el hecho de que todas estas medidas son invisibles para los usuarios, existen grupos de individuos recelosos que dudan de la privacidad real de su información. Por todo ello, esta aplicación pretende añadir una medida adicional para mantener su privacidad y que satisfaga a estos usuarios, consistente en añadir un nivel adicional de cifrado personalizado por el usuario. Conviene aclarar que, para que esto último sea de utilidad, el receptor deberá estar en condiciones de descifrar el mensaje recibido, por ejemplo, como hemos mencionado anteriormente, mediante el conocimiento de una clave secreta.

El cifrado de los datos no sólo es útil para las comunicaciones, sino también para proteger información sensible. De este modo, es posible cifrar información contenida en discos, carpetas o archivos individuales,para evitar el acceso no permitido. Además del beneficio de proteger la privacidad de los usuarios, el cifrado de datos evita otro tipo de ataques como el robo de identidad, o el fraude bancario, además de brindar un mecanismo de protección ante el robo o pérdida de dispositivos con información sensible. El ejemplo más claro de éste último caso son los smartphones, que han evolucionado para contener desde gran cantidad de información personal hasta incluso información completa de nuestras cuentas bancarias

Por último, es necesario mencionar que el uso del cifrado de datos involucra un nivel adicional de complejidad y, en el caso de dispositivos cifrados, reduce la velocidad de acceso a los datos, por la necesidad de descifrarlos al momento de utilizarlos. Por lo tanto, se concluye que, para poder tomar la decisión de emplear o no este nivel adicional de seguridad de forma profesional, se debería analizar el coste/beneficio, donde el uso de este método será tolerable en los casos en que la información a proteger sea lo suficientemente importante, considerando el tiempo que se tarda en protegerla.

1.2. Objetivos

Se desea construir una aplicación que permita a los usuarios aplicar diferentes tipos de cifrado sobre mensajes de texto y archivos de forma visible para el usuario, así como descifrarlos para devolverlos de nuevo a su estado original.

El propósito principal que pretende conseguir esta aplicación es que su manejo sea lo suficientemente sencillo para permitir a usuarios con pocos conocimientos de informática realizar las operaciones provistas por el programa sin grandes complicaciones.

La aplicación proporcionará a petición del usuario una explicación breve y fácilmente entendible sobre el funcionamiento de cada tipo de cifrado, para que los usuarios tengan la posibilidad de adquirir una mayor comprensión sobre el mismo.

La aplicación permitirá realizar operaciones de criptoanálisis para ciertos tipo de cifrado no seguros, es decir, análisis para intentar descifrar un mensaje o archivo sin contar con su clave. Se pretende que esta función tenga un carácter instructivo sobre el funcionamiento y alcance de los propios límites de la aplicación. El motivo principal para no incluir esta funcionalidad para tipos de cifrados seguros se debe a que los cálculos para llevarla a cabo tardarían desde meses hasta cientos años (es precisamente por esto que se les considera seguros).

Para los métodos más avanzados, la aplicación proporcionará métodos de generación automática de claves seguras y gestión de estas claves.

El alcance de la aplicación estará limitado a ciertos ajustes predefinidos para cada tipo de cifrado, y exclusivamente para texto y archivos almacenados localmente el ordenador, aunque no se descarta la posibilidad de ampliar posteriormente las funcionalidades de la aplicación, permitiendo un almacenamiento de archivos en servidor, e incluso la realización de una plataforma de intercambio de mensajes y/o archivos orientada específicamente a la seguridad.

La consecución de todos los objetivos anteriores se materializará en una interfaz sencilla que permitirá a los usuarios realizar las operaciones a su gusto simplemente con pulsar un botón.

1.3. Árbol de Características

Este diagrama representa un modelo simple capaz de mostrar el conjunto de características que representan el alcance del proyecto desde un punto de vista de alto nivel.

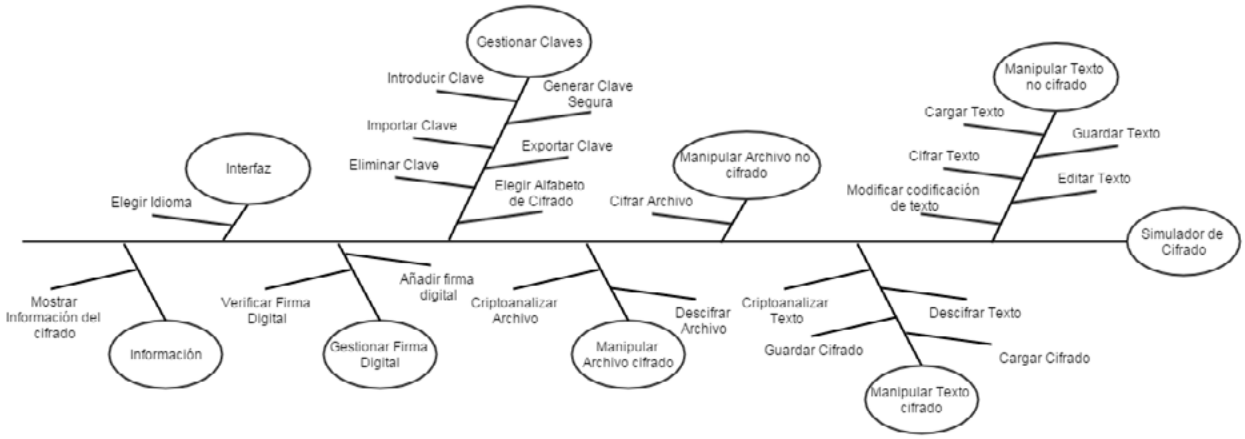


Figura 1.3: Árbol de características

A continuación explicamos brevemente las características mostradas en el diagrama anterior del árbol de características:

- **Manipular Texto no cifrado:** Esta característica se refiere a las opciones de manipulación disponibles para realizar sobre texto plano directamente introducido en la aplicación, incluida la posibilidad de modificar su contenido.
  - *Cargar Texto:* Carga un archivo con contenido de texto en la aplicación.
  - *Guardar Texto:* Guarda en un archivo de tipo ".txt" el texto no cifrado contenido por la aplicación.
  - *Cifrar Texto:* Cifra el texto según el método de cifrado y la clave elegidos y muestra el resultado en pantalla.
  - *Editar Texto:* El usuario puede modificar libremente el texto introducido en la aplicación.
  - *Modificar Codificación de Texto:* El usuario podrá modificar la codificación del texto entre diferentes tipos soportados por la aplicación (UTF–8, UTF–16, US_ASCII, etc.)

- **Manipular Texto cifrado:** Esta característica se refiere a las opciones de manipulación disponibles para realizar sobre texto previamente cifrado cargado en la aplicación.
  - *Cargar Cifrado:* Carga un archivo previamente cifrado mediante el método correspondiente y lo muestra en pantalla como texto.
  - *Guardar Cifrado:* Guarda en un archivo el contenido del texto cifrado.
  - *Descifrar Texto:* Descifra el texto según un método y clave elegidos y muestra el resultado en pantalla.
  - *Criptoanalizar Texto:* Analiza el contenido del texto cifrado y muestra en pantalla un resumen del resultado obtenido.

- **Manipular Archivo no cifrado:** Esta característica se refiere a las opciones de manipulación disponibles para realizar sobre cualquier archivo contenido por el dispositivo en el que se ejecuta la aplicación.
  - *Cifrar Archivo:* Cifra el contenido del archivo según el método de cifrado y la clave elegidos, generando un nuevo archivo cifrado como resultado.

- **Manipular Archivo cifrado:** Esta característica se refiere a las opciones de manipulación disponibles para realizar sobre cualquier archivo previamente cifrado contenido por el dispositivo en el que se ejecuta la aplicación.
  - *Desifrar Archivo:* Descifra el archivo según un método y clave previamente elegidos, generando como resultado el archivo original.
  - *Criptoanalizar Archivo:* Analiza el contenido del archivo cifrado y muestra en pantalla un resumen del resultado obtenido.

- **Gestionar Claves:** Esta característica se refiere a las opciones de gestión de claves que puedan usar los diferentes métodos de cifrado.
  - *Introducir Clave:* El usuario puede introducir una clave libremente como texto.
  - *Generar Clave Segura:* Crea un archivo que contiene una nueva clave segura, generada de acuerdo a los estándares del cifrado elegido.
  - *Importar Clave:* El usuario puede añadir a la lista de claves, una contenida por un archivo externo.
  - *Exportar Clave:* El usuario puede generar un archivo que contenga la clave elegida de acuerdo a ciertas opciones.
  - *Eliminar Clave:* El usuario puede eliminar claves de la lista a su elección.
  - *Elegir alfabeto de cifrado:* Cambia el alfabeto empleado en ciertos métodos de cifrado.

- **Gestionar Firma Digital:** Esta característica permite al usuario añadir un nivel extra de seguridad, confirmando el origen del mensaje (únicamente con los tipos de cifrado que lo permitan).
  - *Añadir Firma Digital:* El usuario puede añadir una firma digital a un mensaje o archivo que vaya a ser cifrado.
  - *Verificar Firma Digital:* El usuario puede comprobar que el mensaje procede del origen que se esperaba y no ha sido manipulado durante su envío comprobando si la firma digital es correcta.

- **Información:** Esta característica ofrece al usuario información sobre los diferentes métodos de cifrado soportados por la aplicación.
  - *Mostrar Información del Cifrado:* Se muestra por pantalla un resumen del funcionamiento del tipo de cifrado elegido.

- **Interfaz:** Esta característica permite al usuario modificar ciertos aspectos relacionados con la vista de la interfaz.
  - *Elegir idioma:* Cambia el idioma de la interfaz.

1.4. Herramientas

Las herramientas utilizadas para la creación del proyecto son:

- Windows 7: Sistema operativo sobre el que se ha empleado el entorno de desarrollo de la aplicación.

- Netbeans 8.0.2: Entorno de desarrollo para creación de aplicaciones Java.
  - *Java 1.7.0_75:* La aplicación será compilada con la última versión (actual) para usuarios de Java, y que la mayoría de sistemas operativos incluyen de forma predeterminada.
  - *Librerías java de terceros:* Para ampliar algunas funcionalidades de la aplicación se han añadido librerías de código abierto, descargables gratuitamente desde los sitios web tika.apache.org, commons.apache.org y swingx.java.net:
    - tika-core-1.11 ; tika-parsers-1.11 ; commons-codec-1.10 ; commons-io-2.4 ; swingx-all-1.6.3

- Enterprise Architect v11(licencia de prueba gratuita, no comercial): Herramienta de gestión de requisitos empleada lo largo del desarrollo de la aplicación, usada también para la creación de los diagramas de casos de uso.

- Gantt Project: Herramienta de software libre para la gestión de recursos en proyectos empleada lo largo del desarrollo de la aplicación, usada también para la creación de los diagramas de gantt.

- Gliffy Diagrams: Herramienta gratuita empleada para la creación de diagramas para diferentes propósitos.

- Adobe Indesign CC (licencia de prueba gratuita, no comercial): Herramienta de procesamiento de textos y documentos, utilizada para la realización del apartado documental.

1.5. Metodología

Para realizar el proyecto se ha elegido emplear un modelo incremental consistente en realizar el modelo en cascada hasta la fase de pruebas y repetir el ciclo hasta completar totalmente el desarrollo del software.

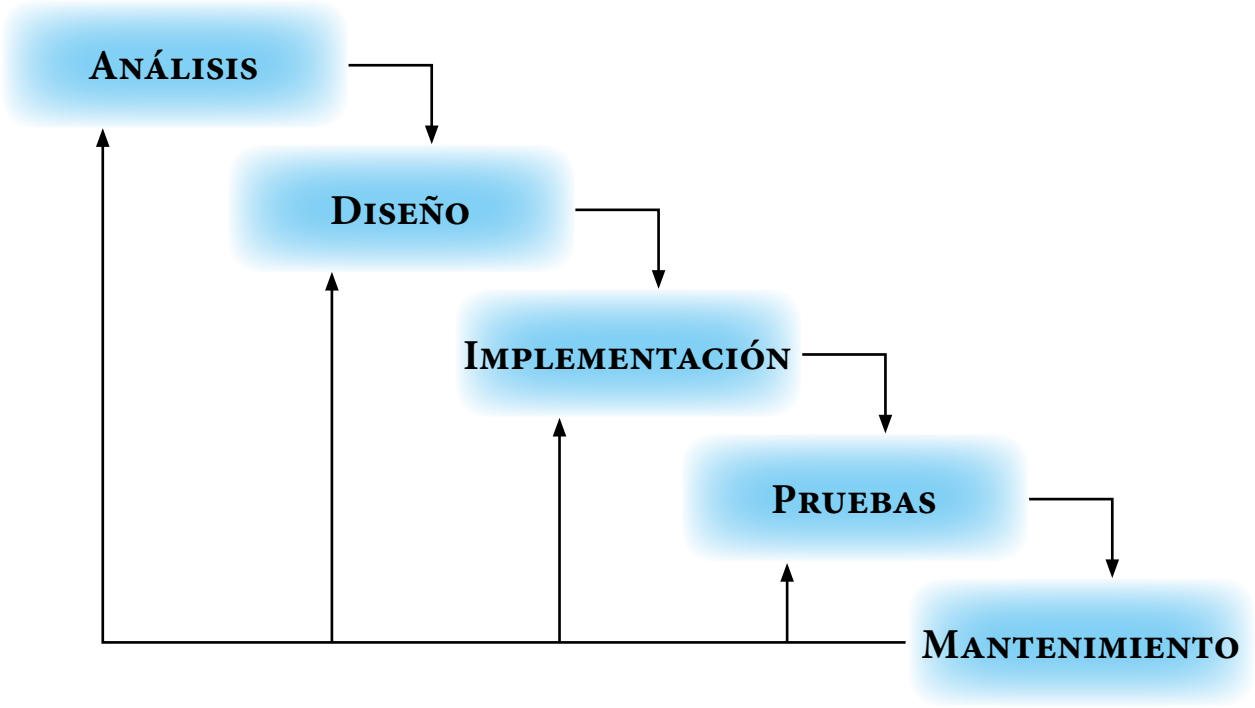


Figura 1.5.1: Metodología — Modelo en Cascada - Ciclo de vida

Como se puede apreciar en la imagen anterior, el desarrollo de este modelo consiste simplemente en avanzar a través de las diferentes fases que lo componen (tras una fase inicial de especificación de requisitos) desde el análisis hasta las fases de pruebas y mantenimiento.

Se puede considerar el modelo incremental como una repetición constante del modelo de cascada con una evolución iterativa. Al ser un modelo retroalimentado podemos adaptarnos a posibles cambios que surjan durante el desarrollo del proyecto, ya sea por detectar una equivocación en las especificaciones o por petición explicita del jefe de proyecto (en este caso, el tutor del TFG).

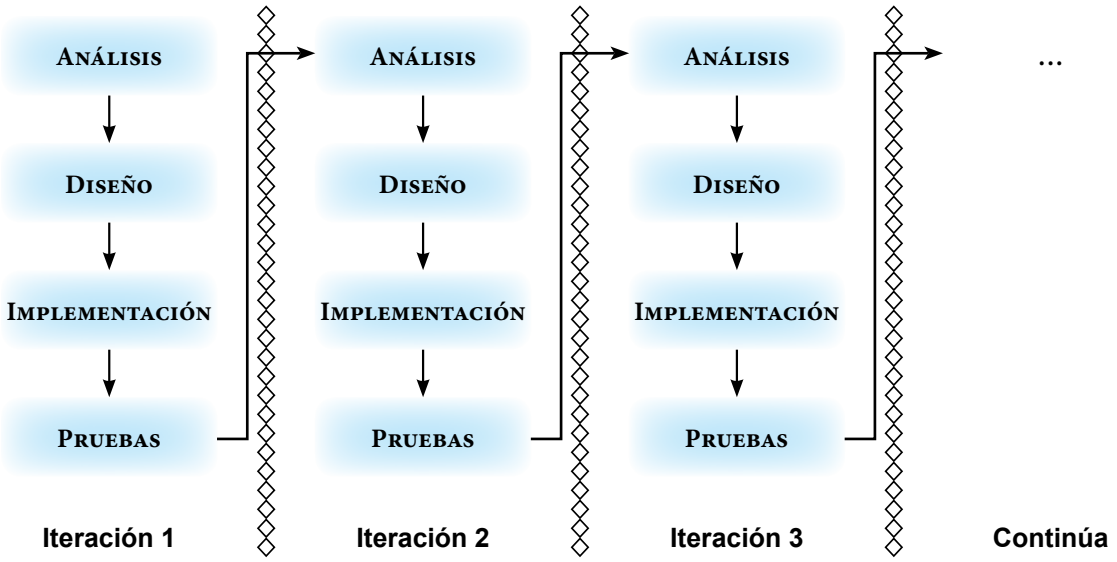


Figura 1.5.2: Metodología — Modelo Iterativo

Como aparece en la imagen anterior, se avanza a través de las fases terminando con la fase de pruebas, tras la cual se pasa a una nueva fase de análisis como un nuevo incremento, repitiendo el proceso hasta lograr completar el programa satisfaciendo las necesidades establecidas.

1.6. Organización del documento

Esta sección describe la estructura de este documento, de modo que sirva de ayuda y facilite su comprensión al lector. El documento se divide en los siguientes apartados:

- **Capítulo 1. Descripción del proyecto:** Este capítulo introduce una presentación de la aplicación y algunas posibles motivaciones para su creación. Describe sus características principales y los aspectos de su desarrollo.
- **Capítulo 2. Planificación y Presupuesto:** Este capítulo muestra en detalle el cálculo del tiempo necesario para la realización del proyecto, así como su presupuesto estimado.
- **Capítulo 3. Análisis del Sistema:** Este capítulo se centra en las funcionalidades exactas que pretende proporcionar la aplicación, así como sus requisitos y casos de uso.
- **Capítulo 4. Diseño del software:** Este capítulo muestra la arquitectura lógica y física que presentará el programa. Además, este capítulo incluye detalles de diseño de los componentes más importantes del proyecto.
- **Capítulo 5. Implementación del proyecto:** En esta parte del documento se va a describir la implementación necesaria para que el proyecto se lleve a cabo. Antes de comenzar la implementación, se va a realizar el paso de la arquitectura lógica que se ha definido en el anterior capítulo.
- **Capítulo 6. Pruebas:** En este capítulo se van a presentar las pruebas realizadas para asegurar que el proyecto funcione correctamente en base a los requisitos establecidos.

- **Capítulo 7. Recursos:** Apartado explicativo que ilustra la distribución y uso de los distintos archivos que se encuentran en el directorio raíz del programa y que son necesarios para, o complementan, su uso.

- **Capítulo 8. Manuales:** En este anexo se proporcionarán instrucciones para saber como instalar el programa y sus nociones básicas de uso.

- **Capítulo 9. Conclusiones:** En este capítulo se mostrarán las conclusiones obtenidas tras finalizar el proyecto, así como los posibles ampliaciones y trabajos futuros que se podrían hacer sobre, o derivando de, este proyecto.

- **Capítulo 10. Estructura del CD:** Apartado explicativo que ilustra la distribución de los distintos contenidos, documentos y código fuente de este programa en el CD que acompaña a este documento.

- **Glosario:** En este anexo se mostrará la terminología utilizada en este documento, así como, una breve definición para cada término reseñado.

- **Bibliografía:** En este capítulo del documento, se mostrará la información exterior consultada para poder realizar el proyecto.

- **Anexo A. Casos de uso:** En este anexo se mostrarán todos los casos de uso del sistema, así como una descripción completa de cada uno de ellos.

- **Anexo B. Requisitos Funcionales:** En este anexo se mostrarán todos los requisitos funcionales del sistema, así como una descripción completa de cada uno de ellos.

- **Anexo C. Imágenes:** Un resumen de las imágenes usadas para ilustrar distintos ejemplos del documento, ordenadas por capítulos y apartados.

- **Anexo D. Tablas:** Un resumen de las tablas usadas para representar distintos datos de utilidad a lo largo del documento, ordenadas por capítulos y apartados.

2. Planificación y Presupuesto

2.1. Estudio previo

El propósito inicial era iniciar el proyecto alrededor de Abril de 2015, sin embargo, por distintos impedimentos el proyecto inicia definitivamente a mediados de Septiembre del 2015, con previsión de finalizar alrededor de Enero del 2016. A la hora de realizar una estimación aproximada se han dado por hecho los siguientes recursos:

- **Recursos humanos:**
  - Jefe de proyecto: El jefe de proyecto asigna los recursos y prioridades, coordina la interacción con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también se cerciora de que el modo de realizar el trabajo que asegure la integridad y calidad del proyecto. Además, se encargará de supervisar la planificación y control del proyecto.
  - Analista: Captura, especifica y valida los requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Colabora en la elaboración de las pruebas funcionales y el modelo de datos.
  - Diseñador: Recibe el análisis y transforma la lista de requisitos del usuario (exenta de tecnología) en un diseño arquitectónico de alto nivel que proveerá las especificaciones a los programadores.
  - Programador: Escribe, depura y mantiene el código fuente de un programa informático.
  - Jefe de pruebas: Realiza las pruebas pertinentes para comprobar el buen funcionamiento y calidad del software.

- **Recursos materiales:**
  - Adobe Indesign CC (licencia de prueba gratuita, no comercial): Herramienta de procesamiento de textos y documentos, utilizada para la realización del apartado documental.
  - GanttProyect (software libre): Programa para la gestión de recursos y seguimiento del proyecto.
  - Ordenador personal (Sistema operativo: Windows 7): Soporte físico y sistema operativo usados para el desarrollo del proyecto.

2.2. Estimación Inicial

La previsión inicial fue comenzar a trabajar en el Trabajo de Fin de Grado a mediados de Abril del 2015 de modo que estuviera terminado a mediados de Agosto del 2015, evolucionando a lo largo de 3 incrementos para terminar totalmente el proyecto.

En total se pretendía que el proyecto constase de 4 fases, consistiendo la primera de ellas en la planificación del proyecto, y el resto en 3 incrementos, actualizando la documentación cuando correspondiese, a medida que el proyecto avanza.

Cada incremento está compuesto por las fases: Análisis, Diseño, Implementación y Pruebas. Se omite la fase final mantenimiento en el calendario porque no forma parte del desarrollo del proyecto propiamente dicho, sino de la continuación de su ciclo de vida tras estar terminado.

Nombre	Duración	Fecha de inicio	Fecha de fin
Proyecto	90	15/04/15	18/08/15
Planificación - Especificación	3	15/04/15	17/04/15
Calendario del proyecto	1	15/04/15	15/04/15
Estimación del Proyecto	2	16/04/15	17/04/15
Documentación	87	20/04/15	18/08/15
Incremento 1	29	20/04/15	28/05/15
Análisis	4	20/04/15	23/04/15
Diseño	7	24/04/15	4/05/15
Implementación	15	5/05/15	25/05/15
Pruebas	3	26/05/15	28/05/15
Incremento 2	29	29/05/15	8/07/15
Análisis	4	29/05/15	3/06/15
Diseño	7	4/06/15	12/06/15
Implementación	15	15/06/15	3/07/15
Pruebas	3	6/07/15	8/07/15
Incremento 3	29	9/07/15	18/08/15
Análisis	4	9/07/15	14/07/15
Diseño	7	15/07/15	23/07/15
Implementación	15	24/07/15	13/08/15
Pruebas	3	14/08/15	18/08/15

Figura 2.2.1: Estimación Inicial — Calendario

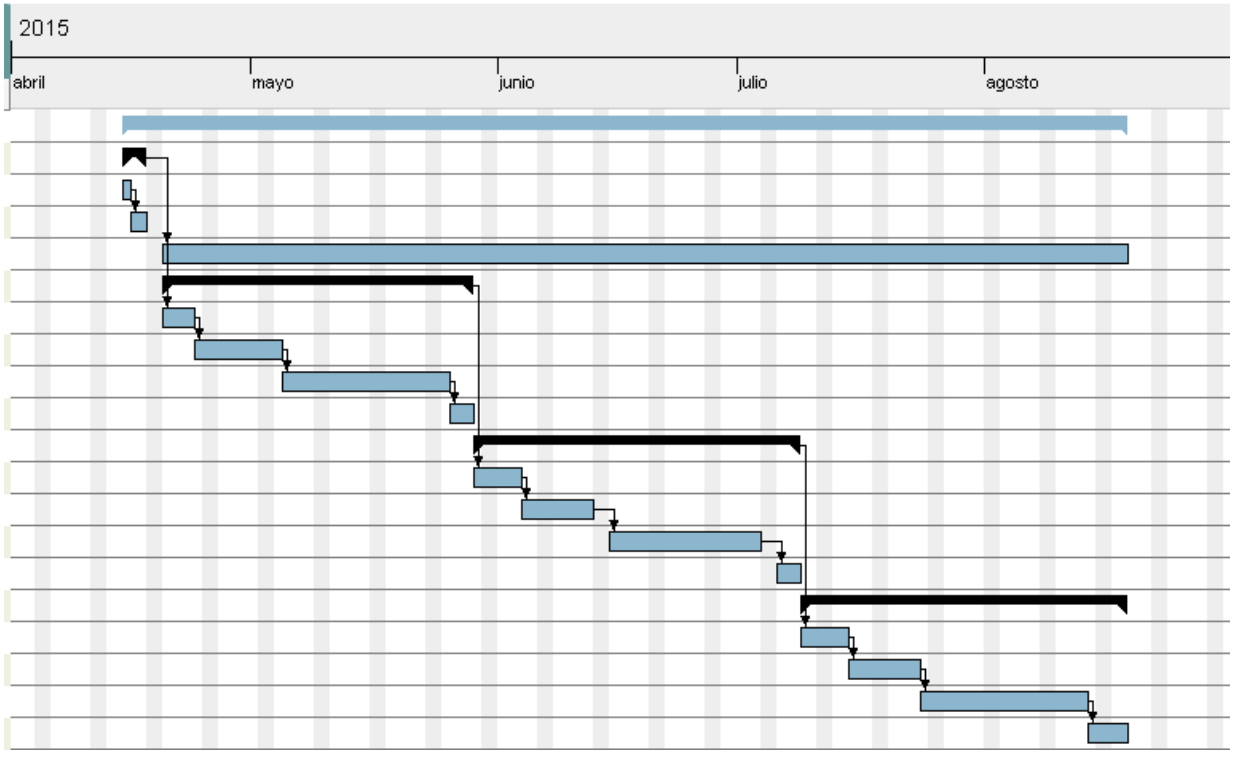


Figura 2.2.2: Estimación Inicial — Diagrama de Gantt

Según el anterior diagrama de Gantt, el proyecto habría terminado el 18 de agosto; consideramos los fines de semana fuera del calendario para calcular los días de trabajo real.

2.3. Presupuesto Inicial

El presupuesto se calculará según el método COCOMO, para lo cual primero estimaremos el número de líneas de código que constituirán la aplicación.

Entradas: 2	**Salidas: 2**	**Consultas: 10**	**Ficheros Externos: 1**	**Ficheros Internos: 1**
Archivos	Archivos	Cifrar archivo/ mensaje*[1]	Configuración	Manuales de uso
Mensajes de texto	Mensajes de Texto	Descifrar archivo/ mensaje*[1]		
		Criptoanálisis archivo/mensaje*[2]		
**En lugar de considerar cada método de cifrado individual a la hora de calcular el número de consultas, consideraremos el número de grupos distintos que engloban los métodos de cifrado implementados. Por ejemplo; el cifrado clásico (3 métodos de cifrado) solo suma 1 consulta de cada tipo.*				
**[1]Algunos métodos de cifrado (como el HASH) no implementan consultas de descifrado, por lo que no suman consultas de este tipo.*				
**[2]La mayoría de métodos de cifrado no cuentan con un método de criptoanálisis, por lo que solo se considerarán los grupos que sí lo implementan.*				

Tabla 2.3.1: Presupuesto Inicial — Cálculo de Puntos de Función no Ajustados

Puntos de Función No Ajustados (PFNA):

	Bajo	**Medio**	**Alto**	**Total**
Entradas	2 (x3)	0 (x4)	0 (x6)	6
Salidas	2 (x4)	0 (x5)	0 (x7)	8
Consultas	4 (x3)	5 (x4)	1 (x6)	38
Ficheros Internos	1 (x7)	0 (x10)	0 (x15)	7
Ficheros Externos	1 (x5)	0 (x7)	0 (x10)	5
				64

Tabla 2.3.2: Presupuesto Inicial — Cálculo de Puntos de Función no Ajustados

Ajuste de los PFNA mediante un factor de ajuste, con un valor entre 0–5, calculado sobre 14 características generales de los sistemas.

Factor de ajuste	0–5	Complejidad	0–5
Comunicación de datos	0	Funciones distribuidas	0
Rendimiento	3	Gran carga de trabajo	2
Frecuencia de transacciones	1	Entrada on-line de datos	0
Requisitos de manejo del usuario final	2	Actualizaciones on-line	0
Procesos complejos	4	Utilización de otros sistemas	0
Facilidad de mantenimiento	0	Facilidad de operación	3
Instalación en múltiples lugares	5	Facilidad de cambio	0
		Total	20

Tabla 2.3.3: Presupuesto Inicial — Ajuste de Puntos de Función

FA = 0,64+(0,01*20) = 0,64+0,20= 0,84
PF = FA · PNFA= 0,84· 64 = 53,76

Cada punto de función equivale aproximadamente a 53 líneas de código en Java. Por lo que el número total de líneas de código será: 2,85 KLDC.

Emplearemos como entorno de desarrollo típico el modelo rígido.

Con esta información, obtenemos un esfuerzo nominal de:

$$PM: 2,8 * 2,85^{(1,2)} = 9,8 \text{ personas-mes}$$

Y un tiempo de desarrollo de:

$$TD: 2,5 * 9.8^{(0,32)} = 5,2 \text{ meses}$$

Sin embargo, estos valores aún no son definitivos, y aplicaremos modificadores sobre ellos en función de varios factores (los factores considerados están resaltados en la tabla):

Atributos	Valor					
	Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
Atributos de software						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de la base de datos		*0,94*	1,00	1,08	1,16	
Complejidad	0,7	0,85	1,00	*1,15*	1,30	
Atributos de Hardware						
Restricciones del tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
Atributos de personal						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	*0,91*	0,82	
Calidad de los programadores	1,42	1,17	1,00	*0,86*	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	*0,95*		
Atributos del proyecto						
Técnicas usadas en la programación	1,24	1,10	1,00	0,91	0,82	
Uso de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones del tiempo de desarrollo	1,22	1,08	1,00	*1,04*	1,10	

Tabla 2.3.4: Presupuesto Inicial — Ajustes por personal y estructura

De este modo, los valores de esfuerzo quedan modificados de la siguiente manera:

*PM: 9,8 * (0,94 * 1,15 * 0,91 * 0,86 * 0,95 * 1,04) = 8,2 personas-mes*

*TD: 2,5 * $8,2^{(0,32)}$ = 4,9 meses*

Costes de Personal

Considerando un sueldo medio de 1200 euros, tenemos un coste total para el personal de:

*Coste: 8,2 * 1.200 = 9.840 euros*

Por otro lado, si tomásemos este coste de un modo objetivo, considerando que el personal real en la realización del TFG será de una única persona, y sustituyendo el coste por un sueldo de unos 8 euros/hora, en un tiempo aproximado de 300 horas que se corresponden con los créditos de la asignatura, en cuyo caso el coste de personal pasaría a ser:

Coste: 2.400 euros

Sin embargo, a efectos del cálculo total, se aplicará el coste calculado mediante la estimación original del método COCOMO.

Costes de Material

Únicamente mencionamos aquí las herramientas, software, etc. que han supuesto un coste de algún tipo en el desarrollo del proyecto.

Concepto	Uso	Precio	Coste
PD – Acer Aspire 5742G	10%	400 €	40 €
Windows 7 Home	10%	140 €	14 €
Conexión a Internet	470%	30 €/mes	141 €
Adobe Indesign	470%	25 €/mes	117,5 €*
**Para el proyecto se ha usado una licencia temporal gratuita, pero suponemos que de tratarse de un proyecto real, se adquiriría la licencia de pago.*			
		Total:	312,5 €

Tabla 2.3.5: Presupuesto Inicial — Coste de material

Coste Total

En base a los costes obtenidos anteriormente obtenemos el siguiente coste final:

Coste Total: (Coste de Personal + Coste de Material) = 9.840 + 312,5 = 10.152,5 €

2.4. Estimación Final

Por motivos de coordinación con otras asignaturas, prácticas en empresa, la necesidad de asistir y superar un mínimo de 1 examen extraordinario en el transcurso del curso académico de 2014–2015, etc. el proyecto se vio retrasado desde el comienzo, por lo que se decidió rehacer la planificación para comenzar finalmente al inicio del curso siguiente, en Septiembre del 2015.

En primer lugar se modificó la fecha inicial del proyecto en el calendario planificado anteriormente, y teniendo en cuenta la previsión de 4,7 meses obtenida en la previsión inicial también se modificó ligeramente la duración de las fases y subfases de trabajo con objeto de finalizar el proyecto a finales de Enero.

Por último, a inicios del mes de Enero se añadió una fase adicional basada en una petición del tutor del proyecto para ampliar el potencial del proyecto, precedida por una nueva estimación de costes y planificación global. Al mismo tiempo, el personal de trabajo (el alumno) entró en un nuevo proyecto que limitaba el tiempo disponible para la realización de la práctica (practicas extracurriculares a jornada completa), por lo que esa ultima fase se extiende en un periodo de tiempo mucho mayor en comparación a las demás, extendiendo la previsión temporal hasta finales del mes de Marzo.

GANTT project

Nombre	Duración	Fecha de inicio	Fecha de fin
Proyecto	138	16/09/15	25/03/16
Planificación - Especificación	3	16/09/15	18/09/15
Calendario del proyecto	1	16/09/15	16/09/15
Estimación del Proyecto	2	17/09/15	18/09/15
Documentación	135	21/09/15	25/03/16
Incremento 1	28	21/09/15	28/10/15
Análisis	3	21/09/15	23/09/15
Diseño	7	24/09/15	2/10/15
Implementación	14	5/10/15	22/10/15
Pruebas	4	23/10/15	28/10/15
Incremento 2	28	29/10/15	7/12/15
Análisis	3	29/10/15	2/11/15
Diseño	7	3/11/15	11/11/15
Implementación	14	12/11/15	1/12/15
Pruebas	4	2/12/15	7/12/15
Incremento 3	28	8/12/15	14/01/16
Análisis	3	8/12/15	10/12/15
Diseño	7	11/12/15	21/12/15
Implementación	14	22/12/15	8/01/16
Pruebas	4	11/01/16	14/01/16
Planificación - Especificación 2	3	15/01/16	19/01/16
Calendario del proyecto	1	15/01/16	15/01/16
Estimación del Proyecto	2	18/01/16	19/01/16
Incremento 4	48	20/01/16	25/03/16
Análisis	6	20/01/16	27/01/16
Diseño	14	28/01/16	16/02/16
Implementación	20	17/02/16	15/03/16
Pruebas	8	16/03/16	25/03/16

Figura 2.4.1: Estimación Final — Calendario

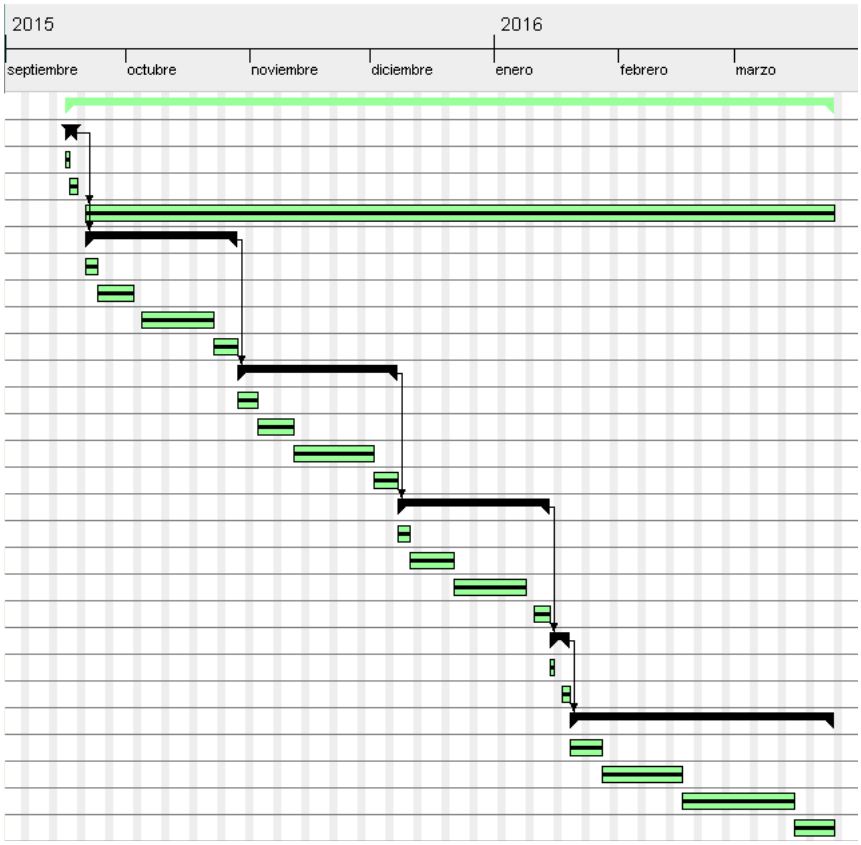


Figura 2.4.2: Estimación Final — Diagrama de Gantt

2.5. Presupuesto Final

Al modificar la duración y algunos aspectos del proyecto durante la realización del mismo, el presupuesto también ha variado, por lo que es necesario calcularlo de nuevo.

Entradas: 2	**Salidas: 2**	**Consultas: 12**	**Ficheros Externos: 1**	**Ficheros Internos: 2**
Archivos	Archivos	Cifrar archivo/ mensaje*[1]	Módulos de cifrado^	Manuales de uso
Mensajes de texto	Mensajes de Texto	Descifrar archivo/ mensaje*[1]		Módulos predeterminados[§]
		Criptoanálisis archivo/mensaje*[2]		
		Firma Digital*[3]		
**En lugar de considerar cada método de cifrado individual a la hora de calcular el número de consultas, consideraremos el número de grupos distintos que engloban los métodos de cifrado implementados. Por ejemplo; el cifrado clásico (3 métodos de cifrado) solo suma 1 consulta de cada tipo.*				
**[1]Algunos métodos de cifrado (como el HASH) no implementan consultas de descifrado, por lo que no suman consultas de este tipo.*				
**[2]La mayoría de métodos de cifrado no cuentan con un método de criptoanálisis, por lo que solo se considerarán los grupos que sí lo implementan.*				
**3Esta consulta engloba 2 operaciones principales, añadir la firma digital y verificar la firma digital, por lo que se considerará como 2 consultas distintas.*				
^Al construirse los distintos módulos de cifrado a partir de una plantilla base para ser leídos de forma externa por el programa, solo se considerará 1 archivo externo para el cómputo final (la plantilla base).				
§El conjunto de módulos se extraen a un directorio predeterminado como un único conjunto, por lo que solo se considerará como 1 archivo interno para el computo final.				

Tabla 2.5.1: Presupuesto Final — Cálculo de Puntos de Función no Ajustados

Puntos de Función No Ajustados (PFNA):

	Bajo	**Medio**	**Alto**	**Total**
Entradas	2 (x3)	0 (x4)	0 (x6)	6
Salidas	2 (x4)	0 (x5)	0 (x7)	8
Consultas	4 (x3)	7 (x4)	1 (x6)	46
Ficheros Internos	2 (x7)	0 (x10)	0 (x15)	14
Ficheros Externos	0 (x5)	1 (x7)	0 (x10)	7
				81

Tabla 2.5.2: Presupuesto Inicial — Cálculo de Puntos de Función no Ajustados

Ajuste de los PFNA mediante un factor de ajuste, con un valor entre 0–5, calculado sobre 14 características generales de los sistemas.

Factor de ajuste	0–5	Complejidad	0–5
Comunicación de datos	0	Funciones distribuidas	0
Rendimiento	3	Gran carga de trabajo	2
Frecuencia de transacciones	1	Entrada on-line de datos	0
Requisitos de manejo del usuario final	2	Actualizaciones on-line	0
Procesos complejos	4	Utilización de otros sistemas	0
Facilidad de mantenimiento	1	Facilidad de operación	4
Instalación en múltiples lugares	5	Facilidad de cambio	4
		Total	27

Tabla 2.5.3: Presupuesto Final — Ajuste de Puntos de Función

FA = 0,81+(0,01*27) = 0,81+0,27= 1,08
PF = FA · PNFA= 1,08· 81 = 87,5

Cada punto de función equivale aproximadamente a 53 líneas de código en Java. Por lo que el número total de líneas de código será: 4,6 KLDC.

Emplearemos como entorno de desarrollo típico el modelo rígido.

Con esta información, obtenemos un esfuerzo nominal de:

$$PM: 2,8 * 4,6^{(1,2)} = 17,4 \text{ personas-mes}$$

Y un tiempo de desarrollo de:

$$TD: 2,5 * 17,4^{(0,32)} = 6,2 \text{ meses}$$

Sin embargo, estos valores aún no son definitivos, y aplicaremos modificadores sobre ellos en función de varios factores (los factores considerados están resaltados en la tabla):

Atributos	Valor					
	Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
Atributos de software						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de la base de datos		*0,94*	1,00	1,08	1,16	
Complejidad	0,7	0,85	1,00	*1,15*	1,30	
Atributos de Hardware						
Restricciones del tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
Atributos de personal						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	*0,91*	0,82	
Calidad de los programadores	1,42	1,17	1,00	*0,86*	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	*0,95*		
Atributos del proyecto						
Técnicas usadas en la programación	1,24	1,10	1,00	*0,91*	0,82	
Uso de herramientas de software	1,24	1,10	1,00	*0,91*	0,83	
Restricciones del tiempo de desarrollo	1,22	1,08	1,00	*1,04*	1,10	

Tabla 2.5.4: Presupuesto Final — Ajustes por personal y estructura

De este modo, los valores de esfuerzo quedan modificados de la siguiente manera:

*PM: 17,4 * (0,94 * 1,15 * 0,91 * 0,86 * 0,95 * 0,91 * 0,91 * 1,04) = 12 personas-mes*

*TD: 2,5 * $12^{(0,32)}$ = 5,5 meses*

Costes de Personal

Considerando un sueldo medio de 1200 euros, tenemos un coste total para el personal de:

*Coste: 12 * 1.200 = 14.400 euros*

Costes de Material

Los costes de material no han variado en absoluto, por lo que se mantienen en los *312,5 €* calculados en el presupuesto inicial.

Coste Total

En base a los costes obtenidos anteriormente obtenemos el siguiente coste final:

Coste Total: (Coste de Personal + Coste de Material) = 14.400 + 312,5 = 14.712,5 €

3. Análisis del Sistema

3.1. Introducción

Esta sección del documento se dedicará a especificar los requisitos que deberán cumplimentarse, junto con los diagramas de casos de uso de la aplicación, además de la relación de cada caso de uso con el requisito al que hacen referencia.

3.2. Identificación de los actores

Al ser una aplicación de escritorio que no cuenta con la intervención de figuras externas, primordialmente contará tan solo con un tipo de actor, al que llamaremos simplemente Usuario.

ACT–01	Usuario
Descripción	Cualquier persona que utilice la aplicación
Comentario	

Tabla 3.2: Actores del sistema

3.3. Requisitos funcionales

RF–1: Elegir cifrado: La aplicación permitirá al usuario elegir un tipo de cifrado entre una variedad de tipos distintos.

RF–2: Cargar alfabetos: La aplicación cargará automáticamente al iniciarse sus alfabetos predeterminados, además de cualquier otro alfabeto que haya podido añadir el usuario.

RF–3: Elegir alfabeto: La aplicación permitirá al usuario elegir uno de entre los alfabetos disponibles para la aplicación.

RF–4: Cargar tipos de codificación de texto: La aplicación cargará automáticamente al iniciarse la lista de tipos de codificación de texto permitidas por Java.

RF–5: Cargar codificación de un archivo de texto: La aplicación detectará automáticamente la codificación de un archivo de texto al cargarlo.

RF–6: Elegir codificación de texto: La aplicación permitirá al usuario cambiar la codificación del texto no cifrado previamente cargado en el programa.

RF–7: Consultar información: La aplicación mostrará una explicación simple del funcionamiento del cifrado elegido.

RF–8: Introducir clave: La aplicación permitirá al usuario introducir una clave alfanumérica de su elección para ser utilizada en los procesos de cifrado/descifrado.

RF–9: Cargar texto: La aplicación permitirá al usuario cargar un archivo de texto para ser cifrado posteriormente.

RF–10: Guardar texto: La aplicación permitirá al usuario guardar un texto no cifrado como un archivo.

RF–11: Cargar texto cifrado: La aplicación permitirá al usuario cargar un archivo cifrado como texto para ser descifrado posteriormente como texto.

RF–12: Guardar texto cifrado: La aplicación permitirá al usuario guardar un texto cifrado como un archivo cifrado de tipo determinado en función del método de cifrado utilizado.

RF–13: Cargar cifrado hexadecimal: La aplicación permitirá al usuario cargar un archivo cifrado como una cadena de texto hexadecimal para ser descifrado posteriormente como texto.

RF–14: Guardar cifrado hexadecimal: La aplicación permitirá al usuario guardar una cadena de texto hexadecimal como un archivo cifrado de tipo determinado en función del método de cifrado utilizado.

RF–15: Cifrar texto: La aplicación permitirá al usuario cifrar mediante el método de cifrado elegido previamente un texto cualquiera, de modo que el usuario vea el resultado inmediatamente en pantalla.

RF–16: Descifrar texto: La aplicación permitirá al usuario descifrar mediante el método de cifrado elegido previamente un texto cifrado cualquiera, de modo que el usuario vea el resultado inmediatamente en pantalla.

RF–17: Cifrar archivo: La aplicación permitirá al usuario elegir un archivo de su PC para cifrarlo mediante el método de cifrado elegido previamente. El resultado será guardado como otro archivo con la extensión adecuada.

RF–18: Descifrar archivo: La aplicación permitirá al usuario elegir un archivo previamente cifrado de su PC para descifrarlo mediante el método de cifrado elegido previamente. El resultado será guardado como otro archivo con la extensión adecuada.

RF–19: Criptoanalizar texto: La aplicación permitirá al usuario analizar un texto previamente cifrado por el método mediante el método de cifrado elegido previamente de modo que proporcione al usuario la clave más probablemente usada para cifrar ese texto.

RF–20: Criptoanalizar archivo: La aplicación permitirá al usuario analizar un archivo previamente cifrado por el método de cifrado mediante el método de cifrado elegido previamente de modo que proporcione al usuario la clave más probablemente usada para cifrar ese archivo.

RF–21: Cargar claves: La aplicación cargará automáticamente las claves correspondientes para tipos de cifrado que no admitan claves alfanuméricas, previamente guardadas por la aplicación.

RF–22: Generar clave segura: La aplicación permitirá al usuario generar a su elección una clave segura para su uso en tipos de cifrado que no admitan claves alfanuméricas.

RF–23: Eliminar clave: La aplicación permitirá al usuario borrar una clave previamente cargada y guardada por la aplicación.

RF–24: Importar clave: La aplicación permitirá al usuario importar un archivo que contenga una clave para el tipo de cifrado elegido previamente que no admita claves alfanuméricas.

RF–25: Exportar clave: La aplicación permitirá al usuario exportar una clave adecuada para el tipo de cifrado elegido previamente que no admita claves alfanuméricas, generando como resultado un archivo que contenga la clave correspondiente.

RF–26: Añadir Firma Digital: La aplicación permitirá al usuario decidir si quiere añadir un nivel de seguridad al mensaje/archivo que se va a cifrar, solicitando una nueva clave que actuará para la generación de la firma.

RF–27: Verificar Firma Digital: La aplicación permitirá al usuario solicitar la comprobación de la firma digital de un mensaje/archivo al tiempo que este se descifra, mostrando el mensaje descifrado únicamente si la firma resulta correcta.

RF–28: Elegir idioma: La aplicación permitirá al usuario elegir uno de los idiomas disponibles para la aplicación.

RF–29: Cargar métodos de cifrado: La aplicación cargará automáticamente al iniciarse los método de cifrado comapatibles existentes en el subdirectorio reservado.

RF–30: Cargar componentes portables: La aplicación extraerá automáticamente y antes de iniciar la interfaz sus componentes mínimos necesarios para su correcto funcionamiento en un árbol de subdirectorios en la carpeta raíz donde fue ejecutado.

3.4. Requisitos no funcionales

RnF–1: Multiplataforma: El uso de la aplicación será compatible con la mayoría de sistemas operativos, incluyendo Windows, Linux y Mac OS entre otros.

RnF–2: Interfaz de usuario simple: Se reducirán al mínimo los ajustes y selecciones disponibles sobre los tipos de cifrado disponibles, de modo que el usuario puedan realizar las operaciones proporcionadas por la aplicación con solo pulsar un botón.

RnF–3: Ajustes de cifrado seguros: Los ajustes aplicados sobre los métodos de cifrado empleados por la aplicación siempre mantendrán unos niveles de seguridad aceptables.

RnF–4: Escalabilidad: La aplicación proporcionará los medios, o será construida de acuerdo a ciertos patrones que permitan el aumento de sus funcionalidades a corto y largo plazo.

RnF–5: Portabilidad: La aplicación será de pequeño tamaño, capaz de iniciarse desde un único archivo ejecutable, con independencia de otros ficheros.

RnF–6: Internacionalización: La interfaz de la aplicación contará, como mínimo con los idiomas Español e Inglés.

3.4.1. Requisitos de información

INF–1: Registro de claves: En el caso de las claves que sean generadas automáticamente por la aplicación, serán guardadas como archivos para su uso posterior por la aplicación.

INF–2: Registro de alfabetos: La aplicación generará unos alfabetos predeterminados al iniciarse, además de guardar en el mismo directorio reservado cualquier alfabeto que pueda ser añadido por la aplicación.

INF–3: Registro de criptoanálisis: En el caso de realizar una función de criptoanálisis, el sistema guardará en un archivo del menor tamaño posible una copia con el resumen de los resultados de esa función de criptoanálisis.

INF–4: Manuales integrados: Los usuarios podrán acceder a un manual sencillo que explique el funcionamiento del tipo cifrado que están utilizando.

3.5. Diagramas y especificaciones de casos de uso

Los siguientes diagramas de casos de uso están divididos de acuerdo a las diferentes estructuras diferenciadas de la aplicación, de modo que su visualización sea simple y también su comprensión lo más sencilla posible.

Este conjunto de diagramas muestran las distintas funcionalidades que proporciona la aplicación, así como relación entre el usuario y todas la acciones que puede realizar mediante ella como herramienta.

Casos de uso — Interfaz general

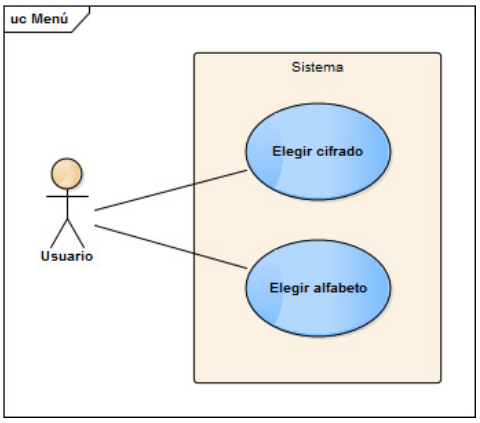


Figura 3.5.1: Diagrama de Casos de uso — Menú

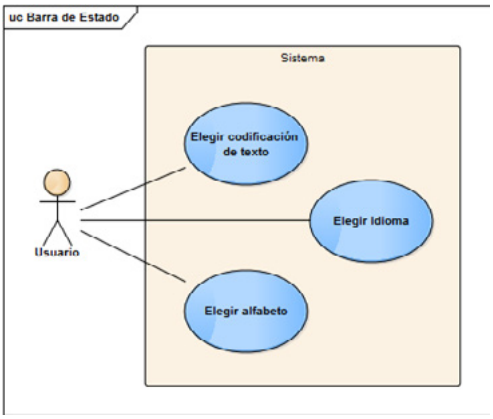


Figura 3.5.2: Diagrama de Casos de uso — Barra de estado

ID y Nombre del caso de uso:	**CU–1 — Elegir cifrado**
Dependencias:	Depende del requisito RF–1: Elegir cifrado.
Disparador:	Seleccionar elemento de una lista.
Descripción:	La aplicación muestra una lista con los métodos de cifrado disponibles, organizados por tipo, para que el usuario seleccione el que pretende usar.
Precondiciones:	
Postcondiciones:	POST–1 — La interfaz cambia de acuerdo a las necesidades el método de cifrado elegido.
Flujo normal:	**1.0 Se muestra la lista de métodos de cifrado disponibles** 1. El usuario elige uno de los métodos de cifrado disponibles.
Flujos alternativos:	
Excepciones:	
Prioridad:	Alta
Frecuencia de uso:	1-2 veces por cada uso de la aplicación.

Tabla 3.5.1: CU–01 — Elegir cifrado

ID y Nombre del caso de uso:	**CU–2 — Elegir alfabeto**
Dependencias:	Depende del requisito RF–3: Elegir alfabeto.
Disparador:	Seleccionar elemento de una lista.
Descripción:	La aplicación muestra una lista con los alfabetos previamente cargados en el sistema para que el usuario seleccione el que pretende usar.
Precondiciones:	PRE–1 — Se debe haber seleccionado previamente un tipo de cifrado que utilice alfabetos para poder seleccionar y cambiar el alfabeto.
Postcondiciones:	POST–1 — Se comprueba la clave introducida actualmente, para que se adecue a las necesidades del alfabeto actual.
Flujo normal:	**2.0 Se muestra la lista de alfabetos disponibles** 1. El usuario elige uno de los alfabetos disponibles. 2. La aplicación lee el archivo del sistema que contiene el alfabeto elegido y comprueba que el archivo exista y tenga un formato correcto. 3. La aplicación comprueba que la clave está compuesta solo por caracteres alfanuméricos existentes en el alfabeto elegido, y borra el resto de caracteres de la clave.
Flujos alternativos:	
Excepciones:	**2.0.E1 El alfabeto elegido no existe o tiene un formato incorrecto** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación. 3. La aplicación elimina el archivo (si existía) 4. La aplicación recarga la lista de alfabetos y repite la operación con el primero de la lista.
Prioridad:	Media
Frecuencia de uso:	1-2 veces por cada uso de la aplicación.

Tabla 3.5.2: CU–02 — Elegir alfabeto

ID y Nombre del caso de uso:	**CU–3 — Elegir codificación de texto**
Dependencias:	Depende del requisito RF–6: Elegir codificación de texto.
Disparador:	Seleccionar elemento de una lista.
Descripción:	La aplicación muestra una lista con los tipos de codificación de texto soportados por java en el sistema para que el usuario seleccione el que pretende usar.
Precondiciones:	
Postcondiciones:	POST–1 — Se modifica el texto no cifrado de la aplicación, de modo que la codificación de ese texto pasa a ser la nueva codificación elegida.
Flujo normal:	**3.0 Se muestra la lista de tipos de codificación disponibles** 1. El usuario elige uno de los tipos de codificación disponibles.
Flujos alternativos:	
Excepciones:	**3.0.E1 La codificación de texto no es soportada** 1. La aplicación cancela la operación. 2. La aplicación repite la operación, cambiando la codificación automáticamente por “UTF–8”.
Prioridad:	Media
Frecuencia de uso:	1-2 veces por cada uso de la aplicación.

Tabla 3.5.3: CU–03 — Elegir codificación de texto

ID y Nombre del caso de uso:	**CU–4 — Elegir idioma**
Dependencias:	Depende del requisito RF–28: Elegir idioma.
Disparador:	Seleccionar elemento de una lista.
Descripción:	La aplicación muestra una lista con los idiomas existentes en el sistema para que el usuario seleccione el que desea usar.
Precondiciones:	
Postcondiciones:	POST–1 — Se modifican los elementos mostrados en la interfaz para cambiar su texto al idioma seleccionado.
Flujo normal:	**4.0 Se muestra la lista de idiomas disponibles** 1. El usuario elige uno de los idiomas disponibles. 2. La aplicación modifica el texto de cada elemento de la interfaz.
Flujos alternativos:	
Excepciones:	**4.0.E1 No se encuentra el archivo interno con los textos del idioma correspondiente** 1. La aplicación muestra un mensaje interno. 2. La aplicación cancela la operación. 3. La aplicación se cierra.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.4: CU–04 — Elegir idioma

Casos de uso — Cifrado clásico

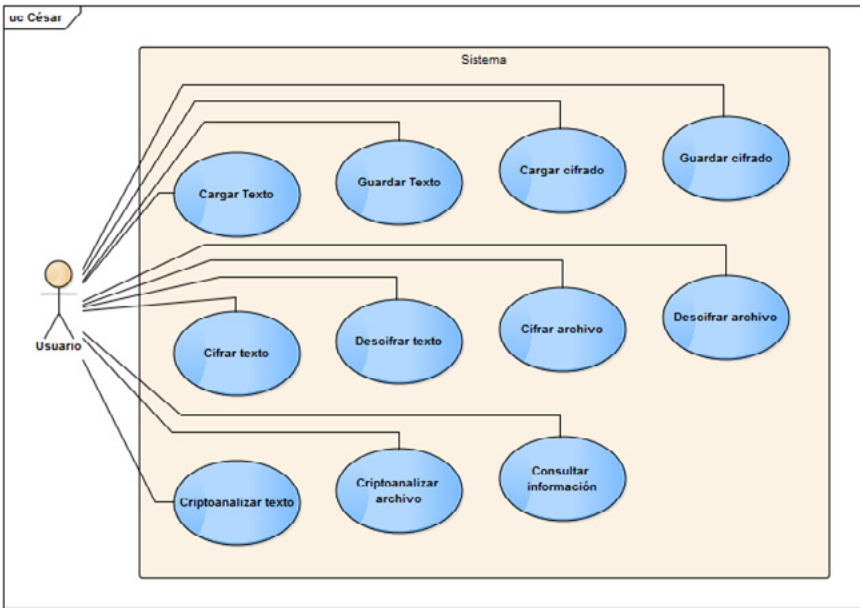


Figura 3.5.3: Diagrama de Casos de uso — Cifrado clásico: César

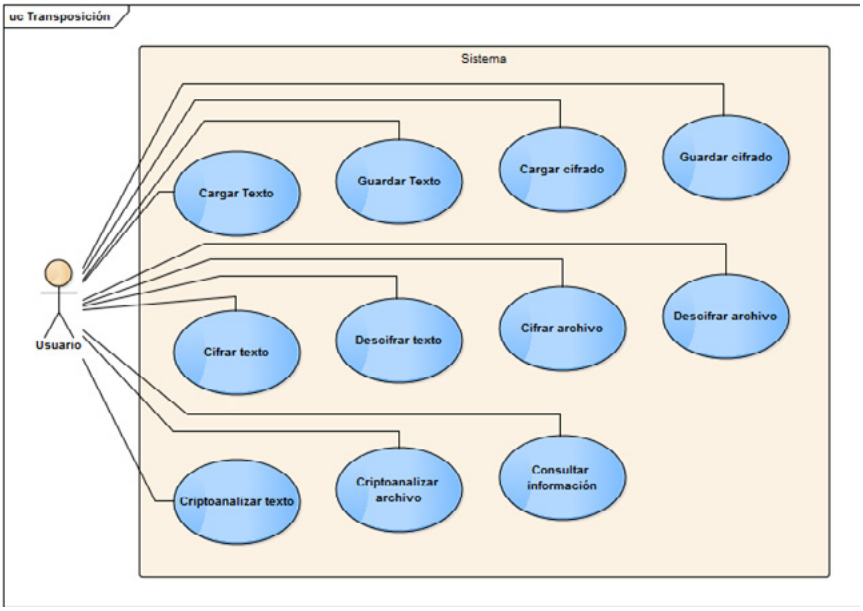


Figura 3.5.4: Diagrama de Casos de uso — Cifrado clásico: Transposición

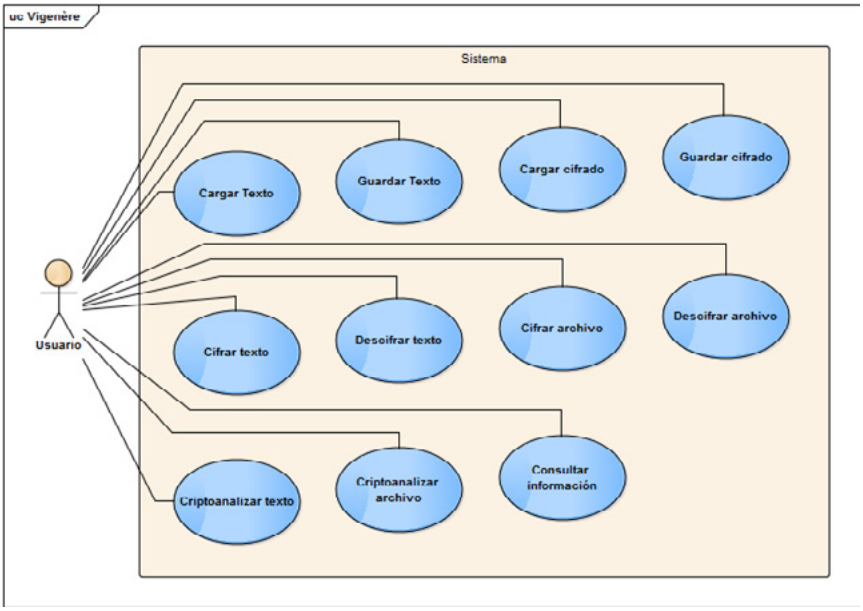


Figura 3.5.5: Diagrama de Casos de uso — Cifrado clásico: Vigenere

ID y Nombre del caso de uso:	CU–5 — Cargar texto
Dependencias:	Depende del requisito RF–9: Cargar texto.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un archivo de tipo ".txt" de su PC, y el contenido se carga en el cuadro de texto de la aplicación.
Precondiciones:	PRE–1 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se muestra en un cuadro de texto editable el contenido del archivo elegido.
Flujo normal:	**5.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación detecta automáticamente el tipo de codificación del texto que contiene el archivo. 3. Se modifica la codificación de texto actual a la detectada por el archivo. 4. Se carga el texto contenido en el archivo en un cuadro de texto editable.
Flujos alternativos:	**5.1 No se puede detectar la codificación archivo, o no está soportada por la aplicación** 1. La aplicación trata la operación como si la codificación detectada hubiera sido "UTF-8".
Excepciones:	**5.0.E1 El archivo elegido no existe.** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación.
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.5: CU–05 — Cargar texto

ID y Nombre del caso de uso:	CU–6 — Guardar texto
Dependencias:	Depende del requisito RF–9: Guardar texto.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un nuevo archivo de su PC para guardar el texto contenido en el cuadro de texto correspondiente de la aplicación.
Precondiciones:	
Postcondiciones:	POST–1 — Se crea un archivo con el contenido del cuadro de texto correspondiente con codificación elegida.
Flujo normal:	**6.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un nuevo archivo de su PC. 2. Se guarda el contenido del cuadro de texto correspondiente en el archivo elegido.
Flujos alternativos:	**6.1 El archivo elegido ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o elegir uno nuevo.
Excepciones:	**6.0.E1 La codificación elegida no está soportada por la aplicación** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación. 3. La aplicación reestablece la codificación "UTF-8" en el texto introducido en la aplicación.
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.6: CU–06 — Guardar texto

ID y Nombre del caso de uso:	**CU–7 — Cargar texto cifrado**
Dependencias:	Depende del requisito RF–11: Cargar texto cifrado.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un archivo de su PC con la extensión adecuada en función del método de cifrado empleado, y el contenido se carga en el cuadro de texto cifrado de la aplicación.
Precondiciones:	PRE–1 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se muestra en un cuadro de texto el contenido del archivo elegido.
Flujo normal:	**7.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación lee el contenido del archivo como un texto con codificación “UTF-8”. 3. Se carga el contenido en un cuadro de texto.
Flujos alternativos:	
Excepciones:	**7.0.E1 El archivo elegido no existe.** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación.
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.7: CU–07 — Cargar texto cifrado

ID y Nombre del caso de uso:	**CU–8 — Guardar texto cifrado**
Dependencias:	Depende del requisito RF–12: Guardar texto cifrado.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un nuevo archivo de su PC para guardar el texto contenido en el cuadro de texto correspondiente de la aplicación.
Precondiciones:	
Postcondiciones:	POST–1 — Se crea un archivo con el contenido del cuadro de texto correspondiente con codificación “UTF-8”.
Flujo normal:	**8.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un nuevo archivo de su PC. 2. Se guarda el contenido del cuadro de texto correspondiente en el archivo elegido.
Flujos alternativos:	**8.1 El archivo elegido ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o elegir uno nuevo.
Excepciones:	
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.8: CU–08 — Guardar texto cifrado

ID y Nombre del caso de uso:	CU–9 — Cifrar texto
Dependencias:	Depende del requisito RF–15: Cifrar texto.
Disparador:	Pulsar un botón.
Descripción:	La aplicación cifra el texto en función del método de cifrado y la clave elegidos y lo muestra en el cuadro de texto cifrado de la aplicación.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe haberse introducido una clave válida. PRE–3 — Debe proporcionarse el texto a cifrar.
Postcondiciones:	POST–1 — Se mostrará en un cuadro de texto el resultado de la operación.
Flujo normal:	**9.0 El usuario pulsa el botón que inicia la operación** 1. La aplicación realiza la operación de cifrado de forma inadvertida al usuario. 2. La aplicación carga en el cuadro de texto cifrado el resultado de la operación.
Flujos alternativos:	
Excepciones:	**9.0.E1 No se ha proporcionado texto o una clave válida** 1. La aplicación cancela la operación.
Prioridad:	Alta
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.9: CU–09 — Cifrar texto

ID y Nombre del caso de uso:	CU–10 — Descifrar texto
Dependencias:	Depende del requisito RF–16: Guardar texto.
Disparador:	Pulsar un botón.
Descripción:	La aplicación descifra el texto en función del método de cifrado y la clave elegidos y lo muestra en el cuadro de texto de la aplicación.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe haberse introducido una clave válida. PRE–3 — Debe proporcionarse el texto a descifrar.
Postcondiciones:	POST–1 — Se mostrará en un cuadro de texto el contenido del archivo elegido.
Flujo normal:	**10.0 El usuario pulsa el botón que inicia la operación** 1. La aplicación realiza la operación de descifrado de forma inadvertida al usuario. 2. La aplicación carga en el cuadro de texto el resultado de la operación.
Flujos alternativos:	
Excepciones:	**10.0.E1 No se ha proporcionado texto o una clave válida** 1. La aplicación cancela la operación.
Prioridad:	Alta
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.10: CU–10 — Descifrar texto

ID y Nombre del caso de uso:	**CU–11 — Cifrar archivo**
Dependencias:	Depende del requisito RF–17: Cifrar archivo.
Disparador:	Pulsar un botón.
Descripción:	El usuario elige un archivo de su PC, y la aplicación generará un nuevo archivo como resultado de cifrar su contenido con el método de cifrado y la clave elegidos.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe haberse introducido una clave válida. PRE–3 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se creará un nuevo archivo con la extensión adecuada en función del método de cifrado elegido
Flujo normal:	**11.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación realiza la operación de cifrado de forma inadvertida al usuario. 3. La aplicación guarda en un nuevo archivo el resultado de la operación.
Flujos alternativos:	**11.1 El archivo generado por la aplicación ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o cancelar la operación.
	11.2 Se utiliza un método de cifrado clásico (solo cifra texto) 1. La aplicación detecta automáticamente la codificación del texto contenido por el archivo. 2. Se lee en contenido del archivo como un texto con la codificación detectada. 3. El texto resultante de la operación de cifrado se guarda con la codificación “UTF-8”.
Excepciones:	**11.0.E1 El archivo elegido no existe, o la clave no es válida** 1. La aplicación cancela la operación.
Prioridad:	Alta
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.11: CU–11 — Cifrar archivo

ID y Nombre del caso de uso:	CU–12 — Descifrar archivo
Dependencias:	Depende del requisito RF–18: Descifrar archivo.
Disparador:	Pulsar un botón.
Descripción:	El usuario elige un archivo de su PC, y la aplicación generará un nuevo archivo como resultado de descifrar su contenido con el método de cifrado y la clave elegidos.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe haberse introducido una clave válida. PRE–3 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se creará un nuevo archivo con el nombre y extensión del archivo original (previo al cifrado).
Flujo normal:	**12.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación realiza la operación de descifrado de forma inadvertida al usuario. 3. La aplicación guarda en un nuevo archivo el resultado de la operación.
Flujos alternativos:	**12.1 El archivo generado por la aplicación ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o cancelar la operación.
	12.2 Se utiliza un método de cifrado clásico (solo cifra texto) 1. El texto resultante de la operación de descifrado se guarda con la codificación "UTF-8".
Excepciones:	**12.0.E1 El archivo elegido no existe, o la clave no es válida** 1. La aplicación cancela la operación.
Prioridad:	Alta
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.12: CU–12 — Descifrar archivo

ID y Nombre del caso de uso:	CU–13 — Criptoanalizar texto
Dependencias:	Depende del requisito RF–19: Criptoanalizar texto.
Disparador:	Pulsar un botón.
Descripción:	La aplicación examina el texto cifrado de la aplicación, genera un archivo resumen con los resultados de las operaciones de análisis y lo muestra en pantalla.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe proporcionarse el texto a analizar.
Postcondiciones:	POST–1 — Se creará un nuevo archivo con información sobre el resultado de las operaciones realizadas. POST–2 — Se mostrará el archivo creado por pantalla.
Flujo normal:	**13.0 El usuario pulsa el botón que inicia la operación** 1. La aplicación realiza la operación de criptoanálisis del texto de forma inadvertida al usuario. 2. La aplicación crea el archivo resultante con la información recopilada. 3. La aplicación muestra el archivo creado por pantalla.
Flujos alternativos:	
Excepciones:	**13.0.E1 No se ha proporcionado texto.** 1. La aplicación cancela la operación.
Prioridad:	Baja
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.13: CU–13 — Criptoanalizar texto

ID y Nombre del caso de uso:	**CU–14 — Criptoanalizar archivo**
Dependencias:	Depende del requisito RF–20: Criptoanalizar archivo.
Disparador:	Pulsar un botón.
Descripción:	El usuario elige un archivo de su PC, la aplicación generará archivo resumen como resultado de analizar su contenido, y mostrará ese archivo por pantalla.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se creará un nuevo archivo con información sobre el resultado de las operaciones realizadas. POST–2 — Se mostrará el archivo creado por pantalla.
Flujo normal:	**14.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación realiza la operación de análisis de forma inadvertida al usuario. 3. La aplicación crea el archivo resultante con la información recopilada. 4. La aplicación muestra el archivo creado por pantalla.
Flujos alternativos:	
Excepciones:	**14.0.E1 El archivo elegido no existe** 1. La aplicación cancela la operación.
Prioridad:	Baja
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.14: CU–14 — Criptoanalizar archivo

ID y Nombre del caso de uso:	**CU–15 — Consultar información**
Dependencias:	Depende del requisito RF–7: Consultar información.
Disparador:	Pulsar un botón.
Descripción:	La aplicación abre un archivo ".pdf" predefinido que explica brevemente el funcionamiento del tipo de cifrado correspondiente
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — El archivo predefinido debe existir.
Postcondiciones:	POST–1 — Se mostrará por pantalla el archivo predefinido.
Flujo normal:	**15.0 El usuario pulsa el botón que inicia la operación** 1. La aplicación muestra el archivo predefinido por pantalla.
Flujos alternativos:	
Excepciones:	**15.0.E1 El archivo elegido no existe** 1. La aplicación muestra el mensaje de error correspondiente. 1. La aplicación cancela la operación.
Prioridad:	Baja
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.15: CU–15 — Consultar Información

Casos de uso — Cifrado de flujo

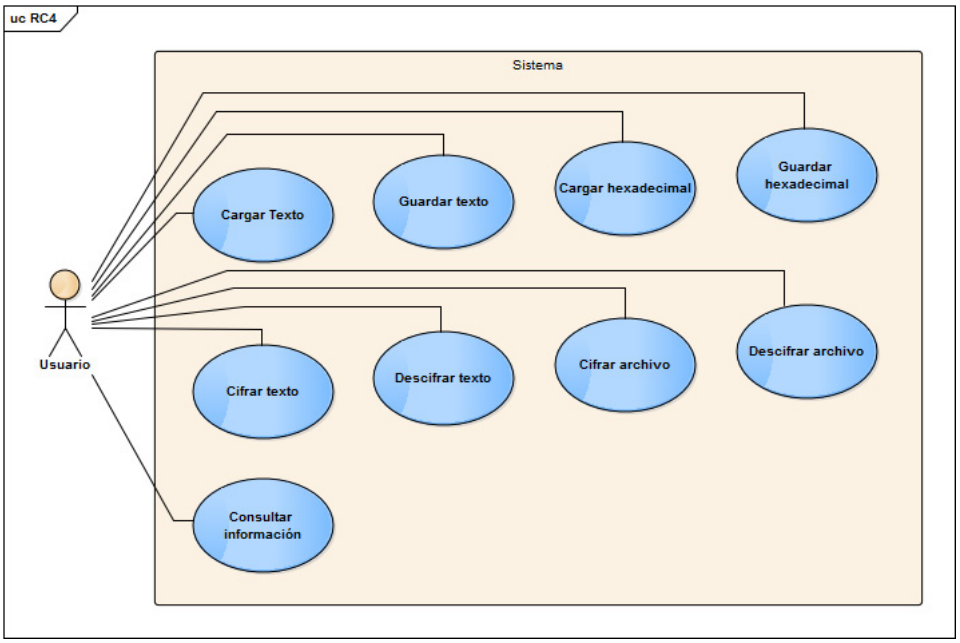


Figura 3.5.6: Diagrama de Casos de uso — Cifrado de flujo: RC4

ID y Nombre del caso de uso:	**CU–16 — Cargar cifrado hexadecimal**
Dependencias:	Depende del requisito RF–13: Cargar cifrado hexadecimal.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un archivo de su PC con la extensión adecuada en función del método de cifrado empleado, y el contenido se carga como texto hexadecimal en el cuadro de texto cifrado de la aplicación.
Precondiciones:	PRE–1 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se muestra en un cuadro de texto el contenido del archivo elegido.
Flujo normal:	**16.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación lee el contenido del archivo como una cadena de bytes. 3. Se carga el contenido como texto hexadecimal en un cuadro de texto.
Flujos alternativos:	
Excepciones:	**16.0.E1 El archivo elegido no existe.** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación.
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.16: CU–16 — Cargar cifrado hexadecimal

ID y Nombre del caso de uso:	**CU–17 — Guardar cifrado hexadecimal**
Dependencias:	Depende del requisito RF–14: Guardar cifrado hexadecimal.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un nuevo archivo de su PC para guardar el texto contenido en el cuadro de texto correspondiente de la aplicación.
Precondiciones:	
Postcondiciones:	POST–1 — Se crea un archivo con una cadena de bytes como el contenido del cuadro de texto correspondiente .
Flujo normal:	**17.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un nuevo archivo de su PC. 2. Se transforma el contenido del cuadro de texto correspondiente en la cadena de bytes que representa la secuencia de texto hexadecimal. 3. Se guarda en el archivo elegido la cadena de bytes resultante.
Flujos alternativos:	**17.1 El archivo elegido ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o elegir uno nuevo.
Excepciones:	
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.17: CU–17 — Guardar cifrado hexadecimal

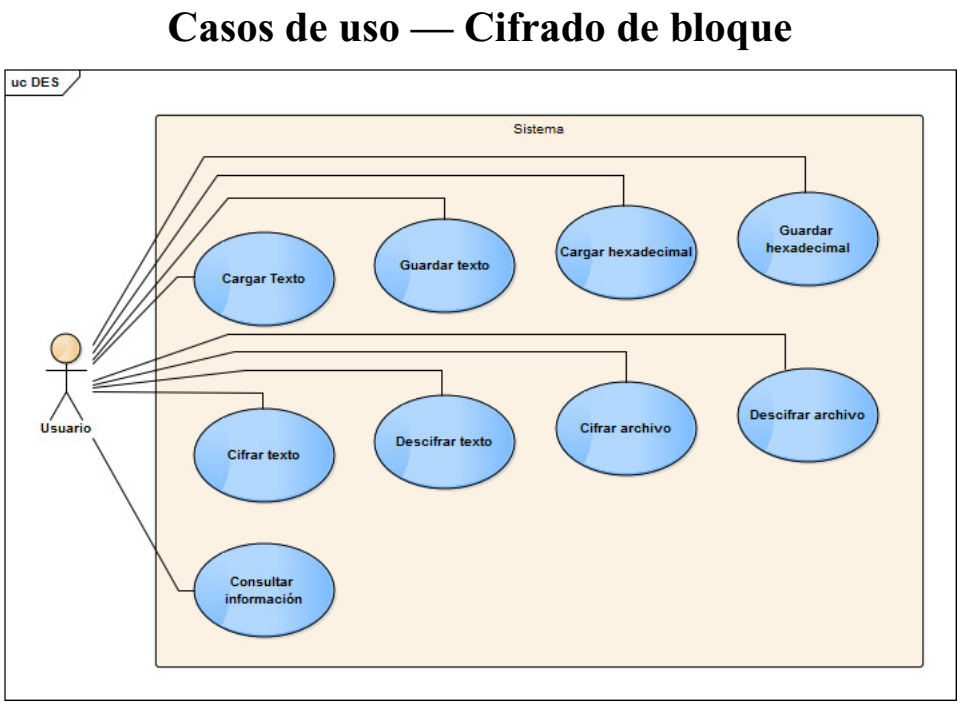


Figura 3.5.7: Diagrama de Casos de uso — Cifrado de bloque: DES

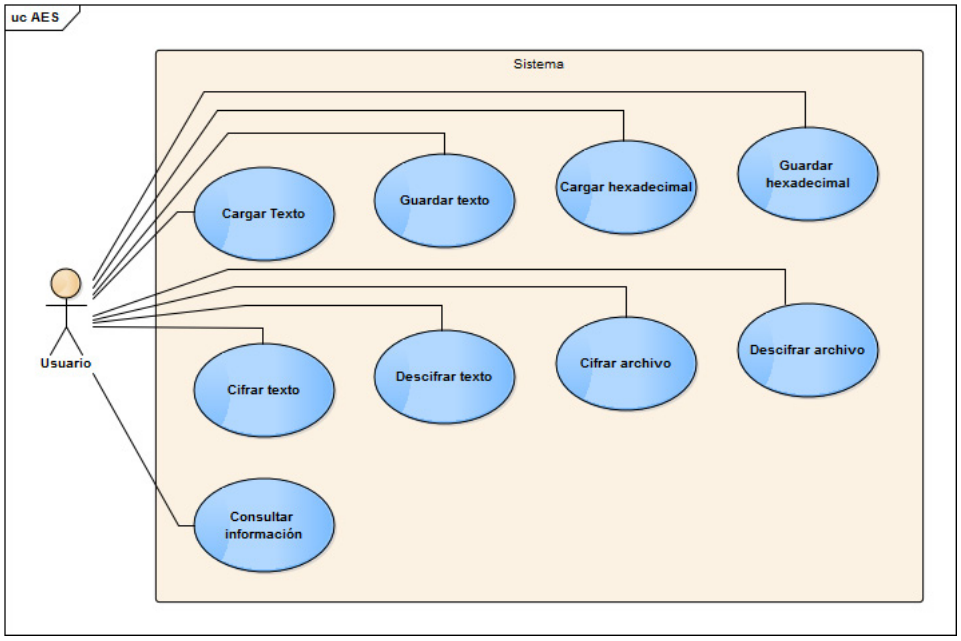


Figura 3.5.8: Diagrama de Casos de uso — Cifrado de bloque: AES

Casos de uso — Cifrado por HASH

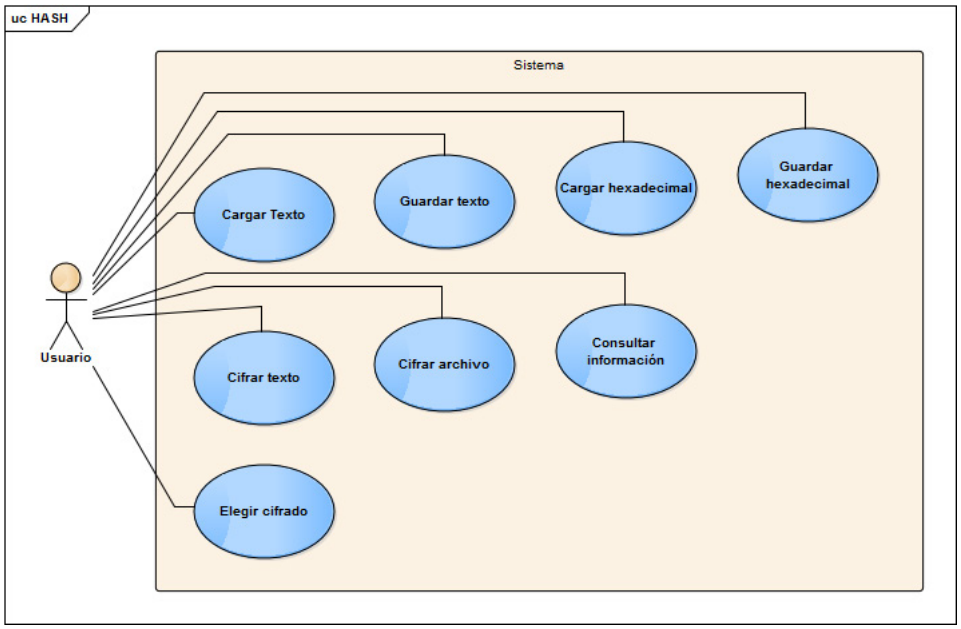


Figura 3.5.9: Diagrama de Casos de uso — Cifrado por HASH

Casos de uso — Cifrado por clave pública

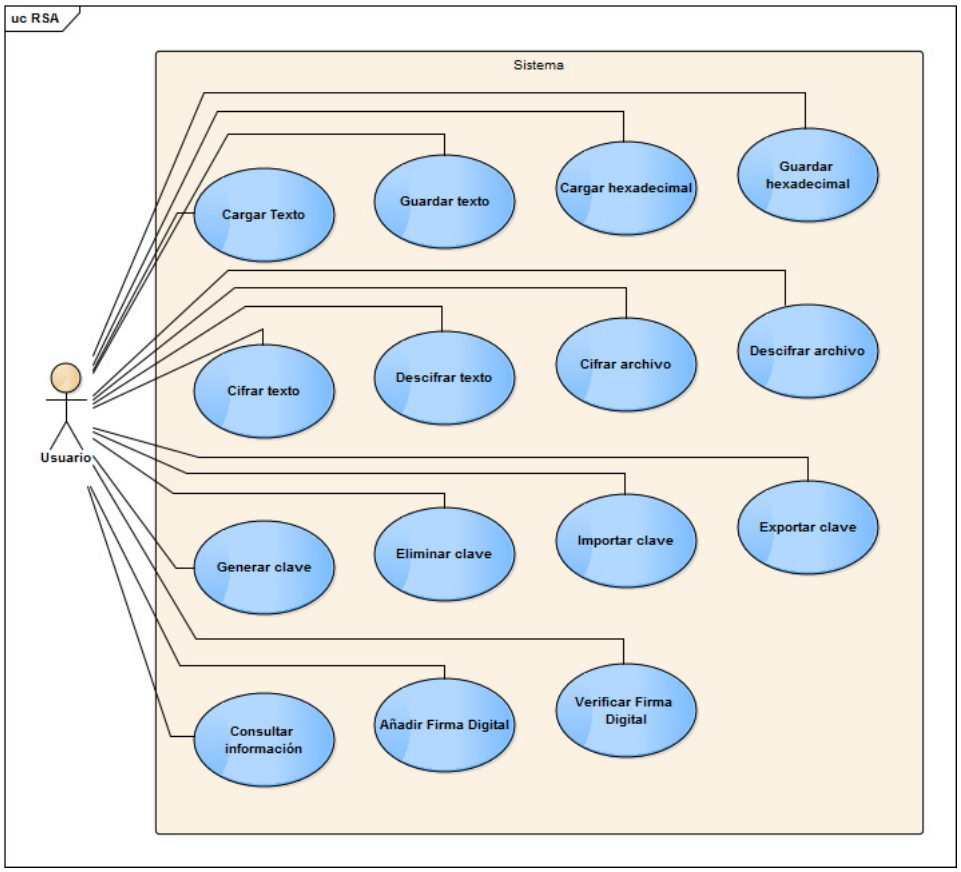


Figura 3.5.10: Diagrama de Casos de uso — Cifrado por clave pública: RSA

ID y Nombre del caso de uso:	**CU–18 — Generar clave segura**
Dependencias:	Depende del requisito RF–22: Generar clave segura.
Disparador:	Pulsar un botón.
Descripción:	La aplicación genera automáticamente una clave segura aleatoria, válida para el tipo de cifrado elegido, y la guarda en el sistema como un archivo del formato adecuado.
Precondiciones:	
Postcondiciones:	POST–1 — Se creará un nuevo archivo con la nueva clave segura aleatoria. POST–2 — Se recargarán las claves de la lista.
Flujo normal:	**18.0 Se muestra una ventana de tipo formulario** 1. El usuario elige un nombre para su clave. 2. La aplicación genera el nuevo archivo clave. 3. La aplicación recarga la lista de claves.
Flujos alternativos:	**18.1 La clave elegida ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar la clave o cancelar la operación.
Excepciones:	
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.18: CU–18 — Generar clave segura

ID y Nombre del caso de uso:	CU–19 — Eliminar clave
Dependencias:	Depende del requisito RF–23: Eliminar clave.
Disparador:	Pulsar un botón.
Descripción:	La aplicación muestra una ventana de confirmación para que el usuario verifique que realmente quiere eliminar la clave seleccionada.
Precondiciones:	PRE–1 — La clave elegida debe existir
Postcondiciones:	POST–1 — Se eliminará el archivo que guarda la clave elegida. POST–2 — Se recargarán las claves de la lista.
Flujo normal:	**19.0 Se muestra una ventana de confirmación** 1. El usuario elige la opción "Sí" o "No" en la ventana de confirmación
Flujos alternativos:	**19.1 Se confirma la eliminación de la clave** 1. La aplicación elimina el archivo correspondiente a la clave elegida. 2. La aplicación recarga la lista de claves.
	19.2 Se cancela la eliminación de la clave 1. La aplicación anula la operación.
Excepciones:	**19.0.E1 La clave elegida no existe.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.19: CU–19 — Eliminar clave

ID y Nombre del caso de uso:	CU–20 — Exportar clave
Dependencias:	Depende del requisito RF–25: Exportar clave.
Disparador:	Pulsar un botón.
Descripción:	El usuario elegirá la carpeta en la que extraer el la clave que desea exportar, y la aplicación copiará la clave en esa carpeta como un nuevo archivo.
Precondiciones:	PRE–1 — La clave elegida debe existir. PRE–2 — La carpeta elegida debe existir.
Postcondiciones:	POST–1 — Se copiará el archivo que guarda la clave elegida a la carpeta elegida.
Flujo normal:	**20.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige la carpeta en la que exportar la clave. 2. La aplicación realiza una copia del archivo que guarda la clave seleccionada, a la carpeta destino elegida por el usuario.
Flujos alternativos:	
Excepciones:	**20.0.E1 La clave o carpeta elegidas no existen.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.20: CU–20 — Exportar clave

ID y Nombre del caso de uso:	CU–21 — Importar clave
Dependencias:	Depende del requisito RF–24: Importar clave.
Disparador:	Pulsar un botón.
Descripción:	El usuario elegirá la el archivo clave que desea importar, y la aplicación copiará esa clave en el sistema.
Precondiciones:	PRE–1 — El archivo elegido debe existir PRE–2 — El archivo elegido debe tener un formato compatible.
Postcondiciones:	POST–1 — Se copiará el archivo clave elegido al la carpeta de claves predeterminada del sistema. POST–2 — La aplicación recarga la lista de claves
Flujo normal:	**21.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige el archivo clave que desea importar. 2. La aplicación realiza una copia del archivo que guarda la clave seleccionada, a la carpeta de claves predeterminada del sistema.
Flujos alternativos:	
Excepciones:	**21.0.E1 El archivo no existe, o tiene un formato no compatible.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.21: CU–21 — Importar clave

ID y Nombre del caso de uso:	CU–22 — Añadir Firma Digital
Dependencias:	Depende del requisito RF–26: Añadir Firma Digital.
Disparadores:	Activar “Cifrar Archivo” o “Cifrar Texto”.
Descripción:	El usuario determinará si pretende añadir una firma digital al archivo o mensaje, y en caso afirmativo, elegir la clave que usará como firma.
Precondiciones:	PRE–1 — La clave elegida como firma debe existir
Postcondiciones:	POST–1 — Se añadirá al inicio del archivo o texto resultante un conjunto de datos que se corresponden con la firma digital del archivo.
Flujo normal:	**22.0 Se muestra una ventana de confirmación** 1. El usuario elige si desea añadir la firma digital. 2. En caso afirmativo el usuario vuelve elige una clave adicional en una nueva ventana, que será la usada para realizar la firma digital 3. El proceso de encriptado comienza con la adición de la firma digital al inicio del archivo o texto cifrado. 4. El proceso de encriptado continúa normalmente.
Flujos alternativos:	**22.1 Se cancela la operación de firma digital** 1. El proceso de encriptado continúa normalmente sin la firma digital.
Excepciones:	**22.0.E1 La clave elegida no existe.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.22: CU–22 — Añadir Firma Digital

ID y Nombre del caso de uso:	**CU–23 — Verificar Firma Digital**
Dependencias:	Depende del requisito RF–27: Verificar Firma Digital.
Disparadores:	Activar “Descifrar Archivo” o “Descifrar Texto”.
Descripción:	El usuario determinará si pretende comprobar la existencia y veracidad de la firma digital, y por ende el mensaje completo, en un archivo o texto.
Precondiciones:	PRE–1 — La clave elegida como firma debe existir
Postcondiciones:	POST–1 — La creación del archivo o texto descifrado dependerá de la veracidad de la firma digital.
Flujo normal:	**23.0 Se muestra una ventana de confirmación** 1. El usuario elige si desea verificar la firma digital. 2. En caso afirmativo el usuario vuelve elige una clave adicional en una nueva ventana, que será la usada para comprobar la firma digital. 3. El proceso de desencriptado comienza aislando los datos que corresponden a la firma digital y comprobando si esta es correcta. 4. El proceso de desencriptado continúa normalmente.
Flujos alternativos:	**23.1 Se cancela la operación de firma digital** 1. El proceso de desencriptado continúa normalmente sin la firma digital.
Excepciones:	**23.0.E1 La clave elegida no existe.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.23: CU–23 — Verificar Firma Digital

4. Diseño del Software

4.1. Introducción

En esta sección del documento se explicarán en detalle los componentes diferenciados de la interfaz de usuario, así como la estructura de su arquitectura lógica y física.

4.2. Interfaz – <Contenedor>

A partir de este momento entenderemos como <Contenedor> al cuadro que delimita los límites de la ventana de la aplicación, que cuenta con las barras de menú y estado, y un espacio central reservado a los elementos visual de los diferentes métodos de cifrado, de modo que únicamente se cargue en pantalla y en la memoria del dispositivo los elementos del método de cifrado que se está empleando actualmente. El <Contenedor> puede ser redimensionado, en cuyo caso también ajusta el tamaño de su contenido.

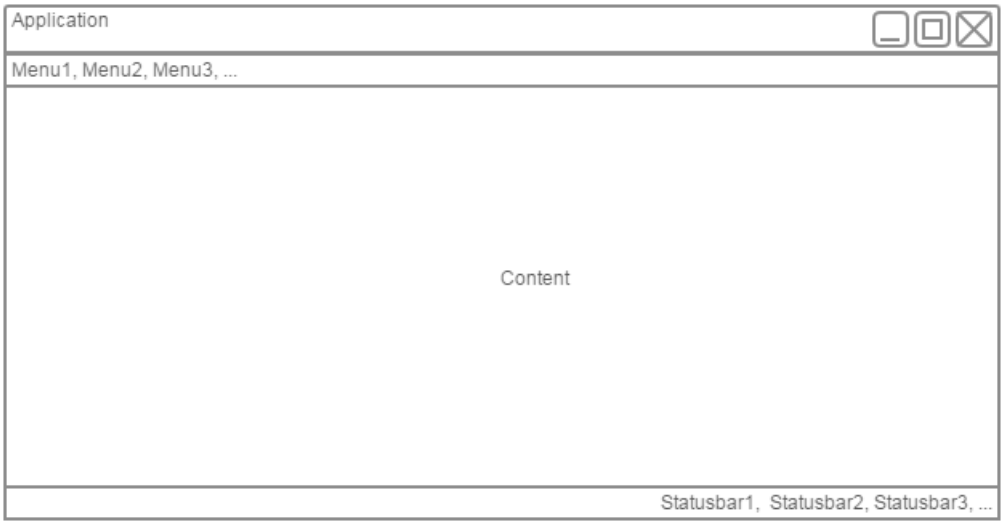


Figura 4.2: Interfaz – <Contenedor>

4.3. Interfaz – <Contenido>

A partir de este momento entendemos como <Contenido> al conjunto de elementos que se mostrarán en el interior del cuadro vacío del <Contenedor>, que cuenta con 2 cuadros de texto para mostrar las operaciones sobre texto plano, y los botones necesarios para llevar a cabo el resto de operaciones.

Los métodos de cifrado creados durante este desarrollo del proyecto están basados en una plantilla base, por lo que casi todos cuentan con la misma interfaz, o una muy similar. De este modo, la ilustración presentada a continuación representa con una gran fiabilidad los elementos que podemos encontrar en el <Contenido>.

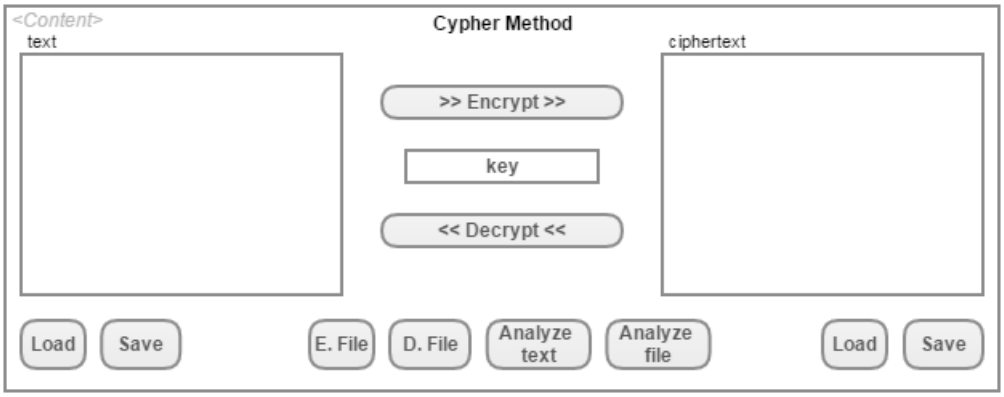


Figura 4.3: Interfaz – <Contenido>

4.4. Arquitectura lógica

La arquitectura lógica representa los componentes lógicos de los que se compone la aplicación y la relación entre ellos.

Como se puede apreciar en el diagrama siguiente, la aplicación está construida bajo el patrón modelo-vista-presentador (MVP), o dicho de otro modo, una versión específica del patrón modelo-vista-controlador.

Concretamente, la construcción de la aplicación está basada en un MVP multicapa, donde cada método o *módulo* de cifrado representa una capa diferenciada, de modo que además de mantener la independencia de cada uno de los 3 grupos de clases principales, dote a la aplicación de una gran escalabilidad, dando la posibilidad de ampliar sus funcionalidades a medio/largo plazo mediante la adición de nuevas capas o *módulos*.

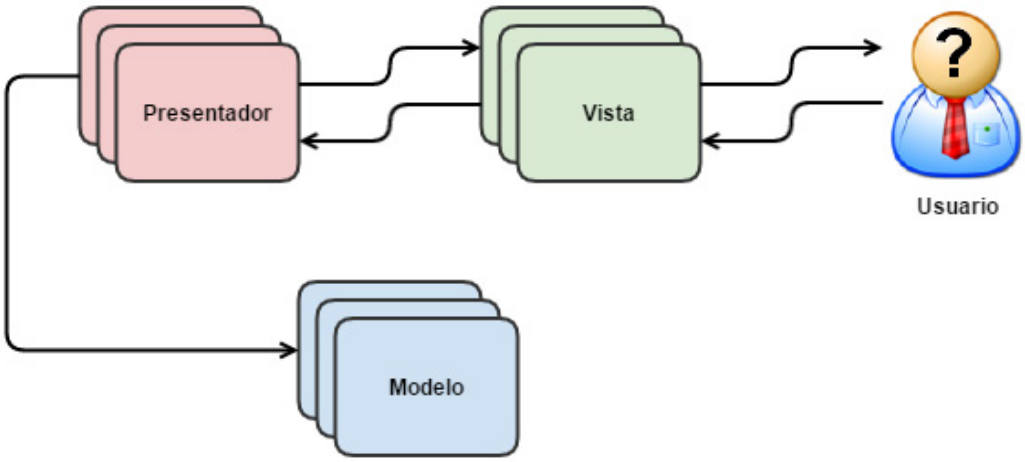


Figura 4.4: Arquitectura lógica

4.5. Arquitectura física

La arquitectura física representa los componentes físicos que intervienen en el funcionamiento de la aplicación. Sin embargo, al tratarse esta de una aplicación de escritorio sin comunicación con servidor o base de datos alguna, nuestra arquitectura física queda limitada al propio dispositivo que ejecuta la aplicación.

Figura 4.5: Arquitectura física

4.6. Diagramas de clases de análisis

Por la división intrínseca del modelo MVP y con objetivo de lograr mayor claridad y entendimiento en su visualización, se mostrará el diagrama UML dividido por cada diferente capa MVP. Para simplificar los diagramas, se omitirán los métodos cuya única función sea la creación de elementos de la interfaz, salvo en el caso de elementos dinámicos o con posibilidades de verse modificados a lo largo del uso de la aplicación; también se omitirán métodos privados que se refieran únicamente a operaciones internas de una clase y que no intervengan en la relación entre clases.

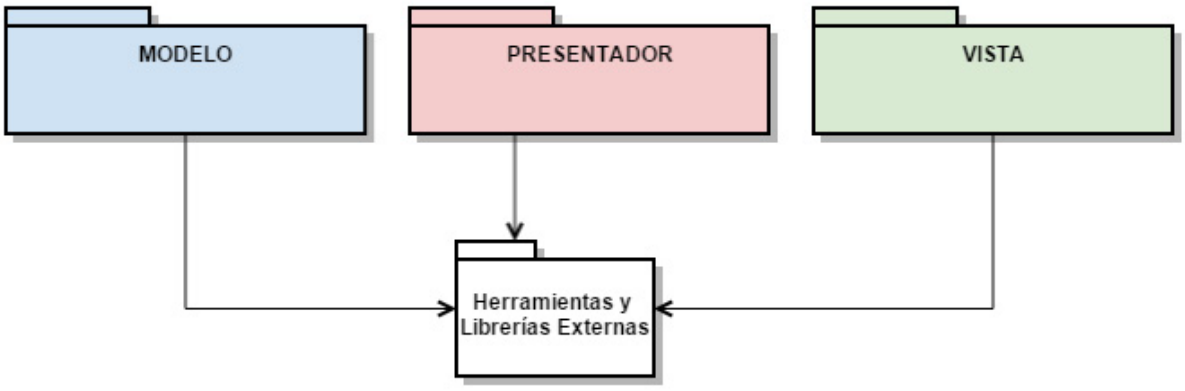


Figura 4.6.1: Diagramas de clases de análisis — Modelo MVP

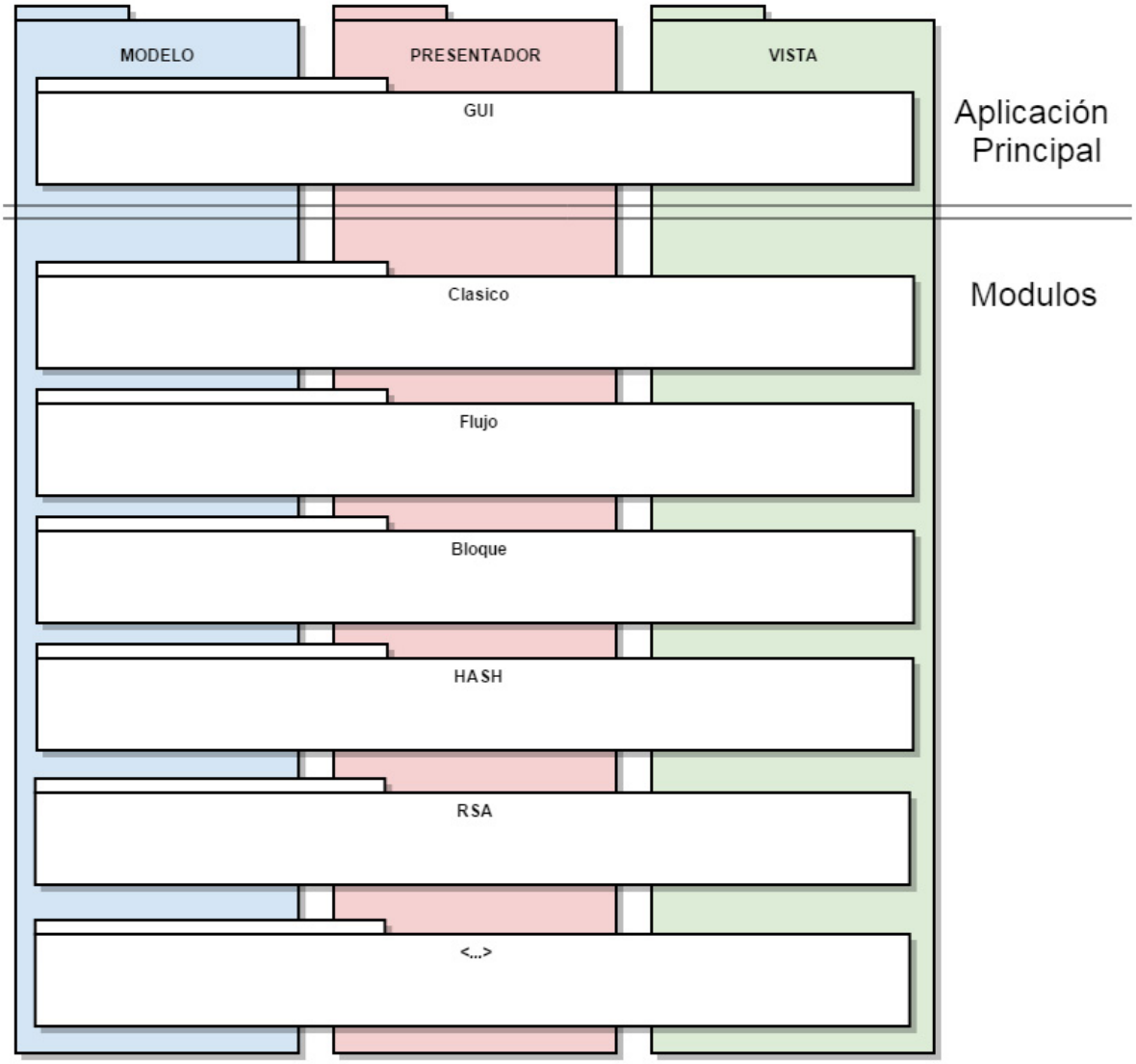


Figura 4.6.2: Diagramas de clases de análisis — División por módulos

Capa de la GUI - Principal

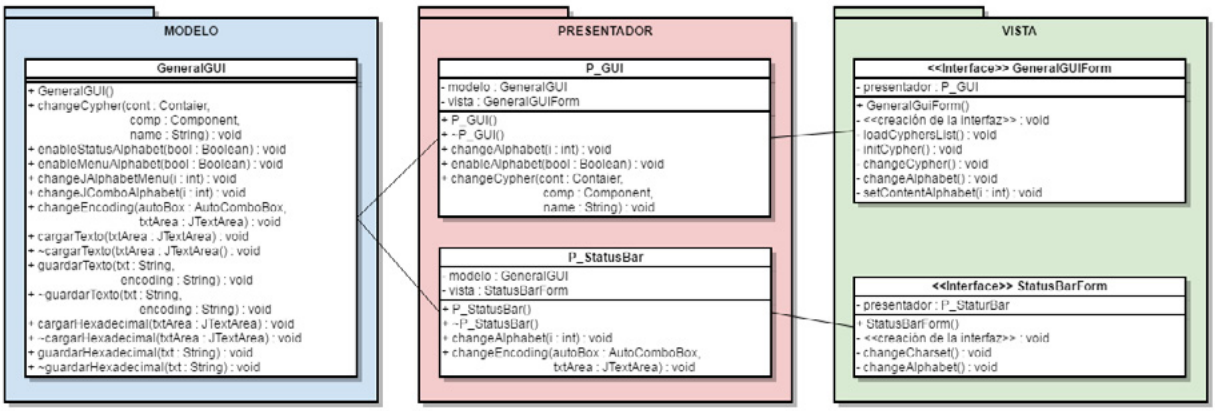


Figura 4.6.3: Diagramas de clases de análisis — Capa GUI

Módulos de Cifrado Clásico

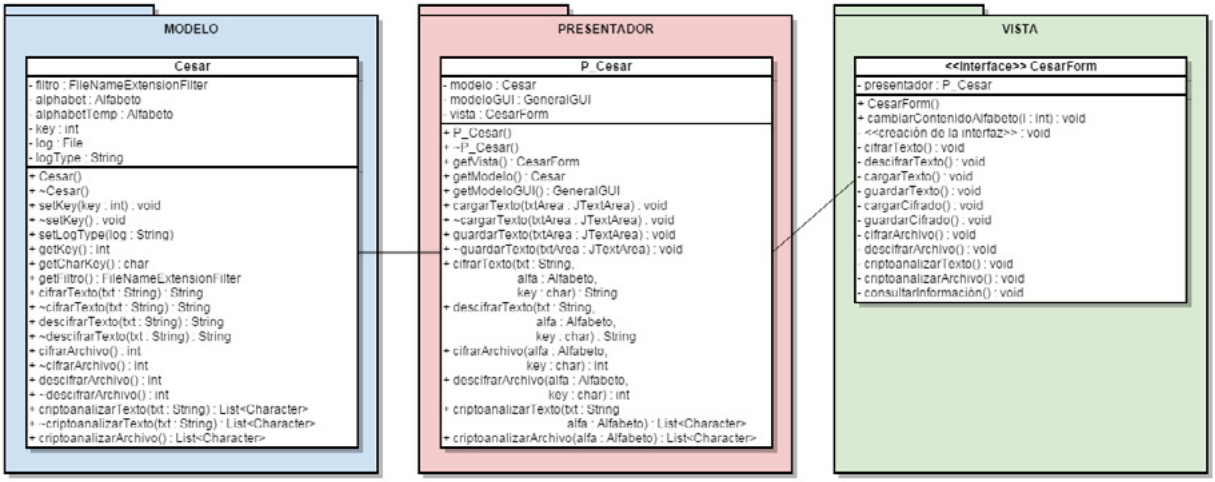


Figura 4.6.4: Diagramas de clases de análisis — Módulo César

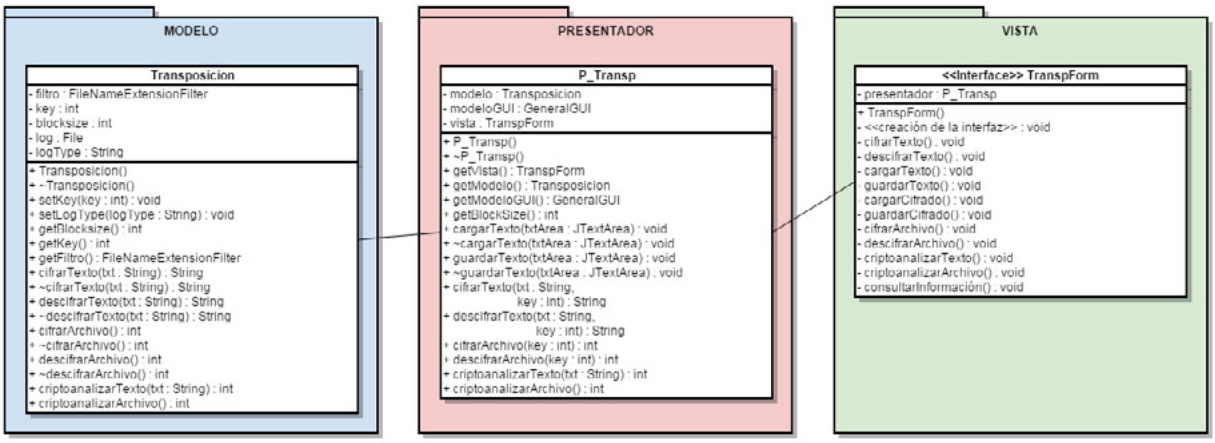


Figura 4.6.5: Diagramas de clases de análisis — Módulo Transposición

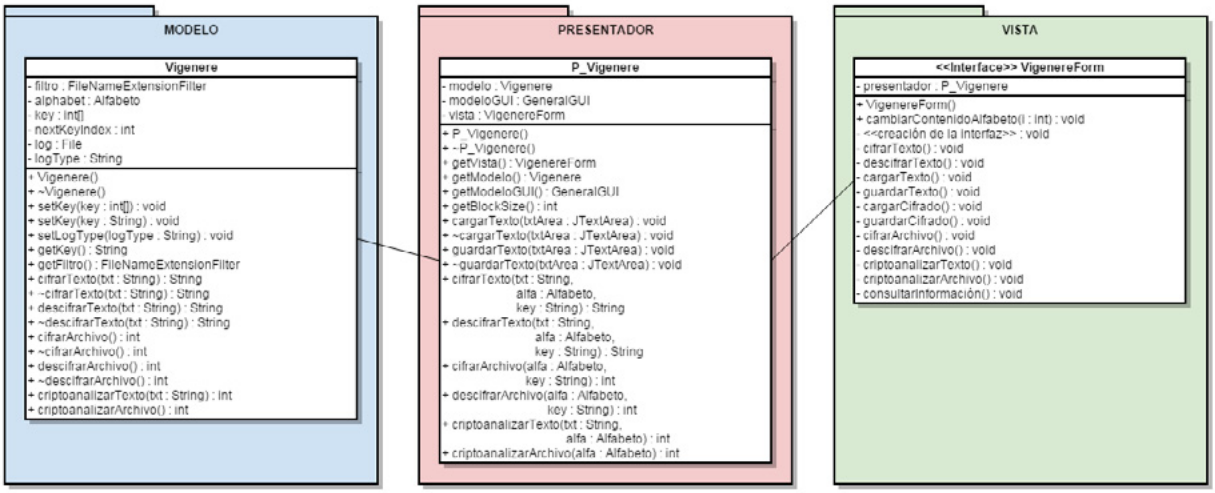


Figura 4.6.6: Diagramas de clases de análisis — Módulo Vigenere

Capa de Cifrado de Flujo

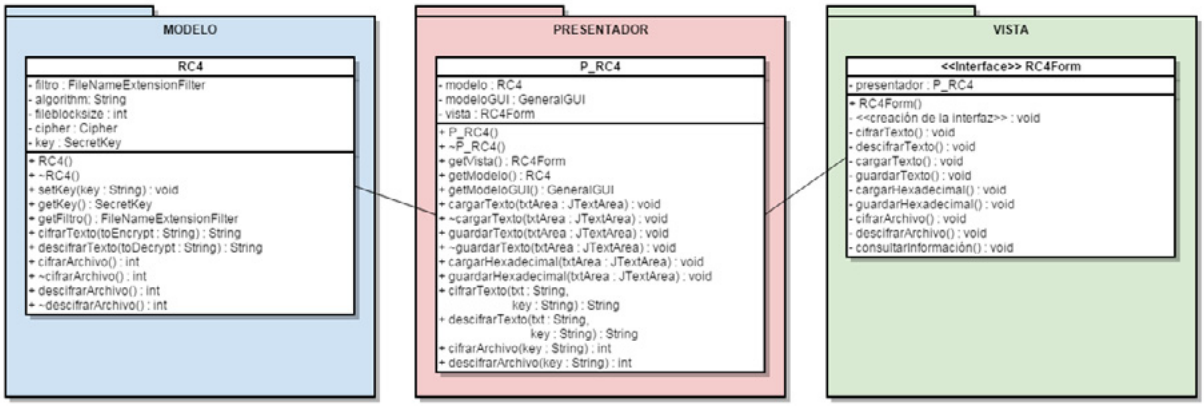


Figura 4.6.7: Diagramas de clases de análisis — Módulo RC4

Capa de Cifrado de Bloque

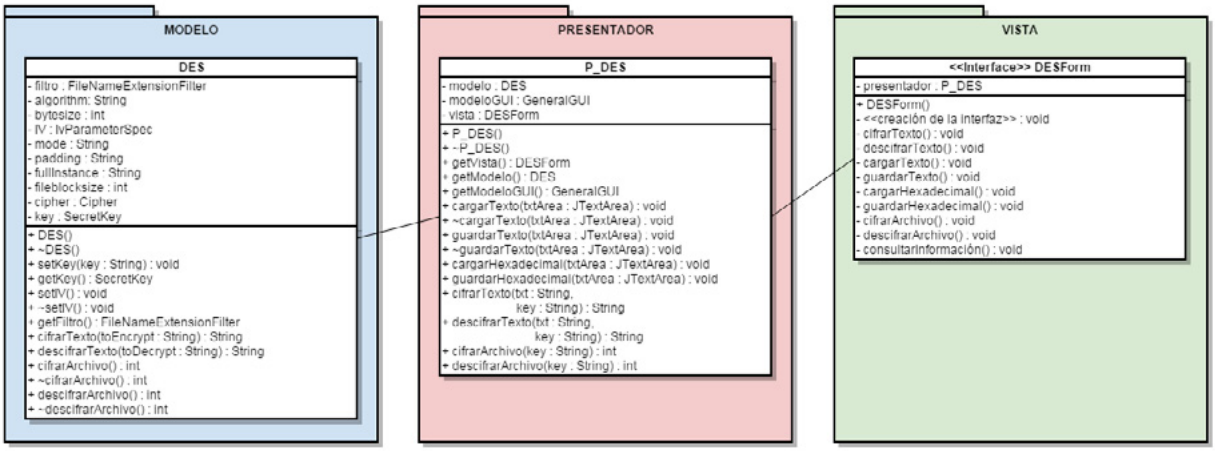


Figura 4.6.8: Diagramas de clases de análisis — Módulo DES

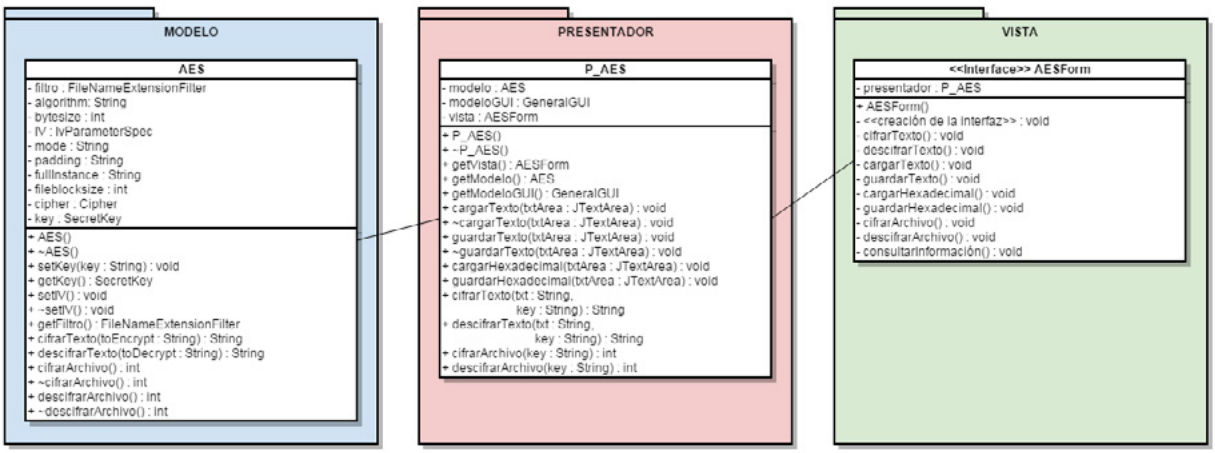


Figura 4.6.9: Diagramas de clases de análisis — Módulo AES

Capa de Cifrado por Hash

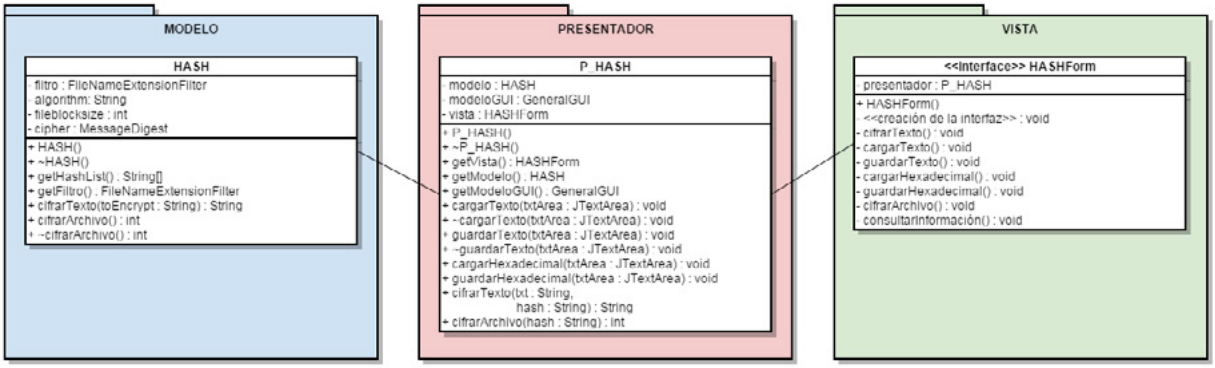


Figura 4.6.10: Diagramas de clases de análisis — Módulo HASH

Capa de Cifrado por Clave Pública

Figura 4.6.11: Diagramas de clases de análisis — Módulo RSA

4.7. Diagramas de secuencia

Los diagramas de secuencia muestran la interacción existente entre los diferentes objetos de la aplicación.

Por la estructura en que está construida la aplicación, los diagramas de secuencia de las diferentes operaciones son extremadamente similares, de modo que únicamente ilustraremos de forma genérica los dos casos más diferenciados y significativos.

Cargar módulo de cifrado

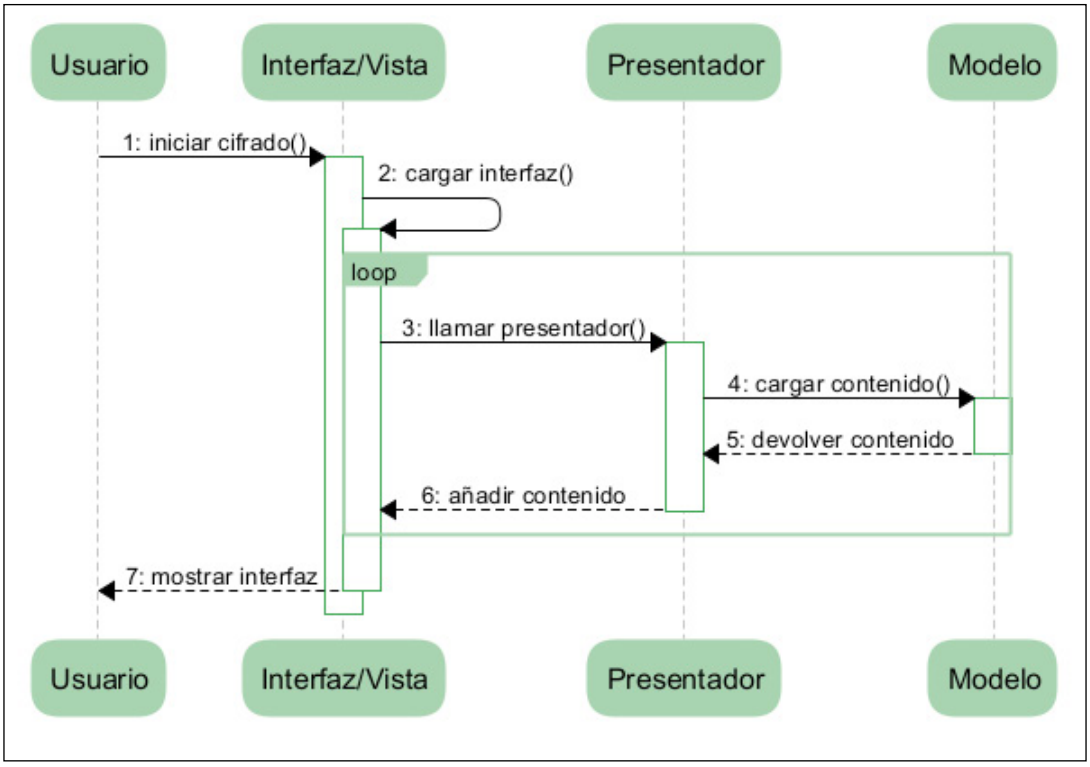


Figura 4.7.1: Diagramas de secuencia — Cargar módulo de cifrado

Realizar operación de cifrado

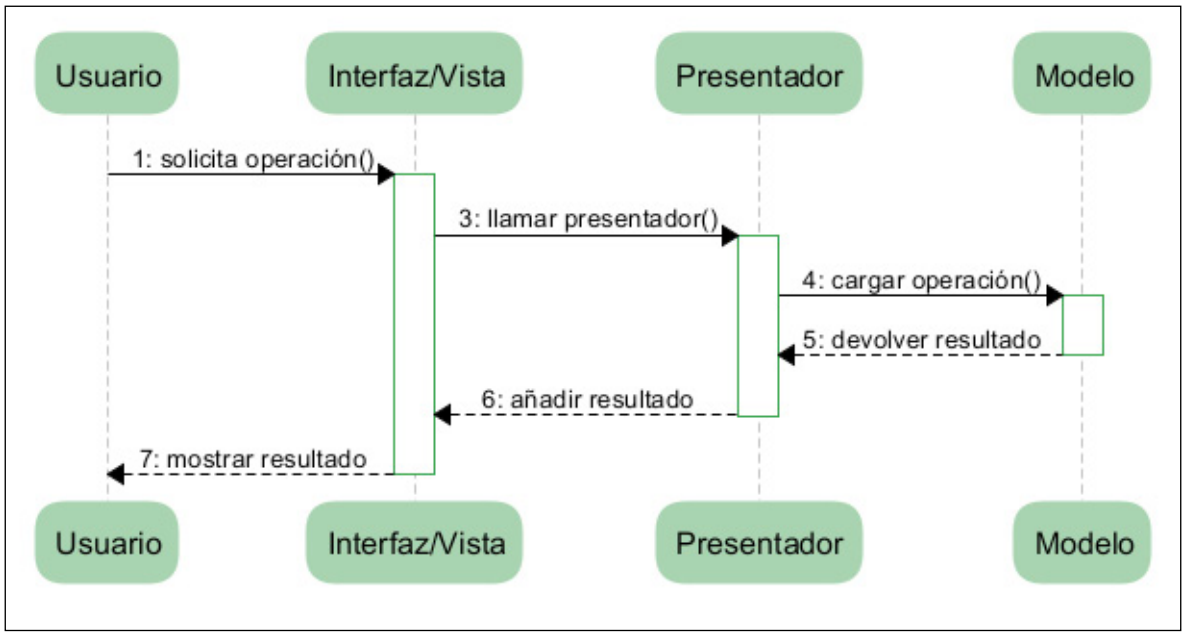


Figura 4.7.2: Diagramas de secuencia — Realizar operación de cifrado

5. Implementación

5.1. Descripción técnica

Todas las funcionalidades de la aplicación se han desarrollado a partir del conjunto de librerías básico de Java, junto a unas librerías de software libre como apoyo, sobre todo, para el manejo de archivos.

Cada operación se realizará sobre el hilo principal de ejecución de la aplicación, bloqueando el programa hasta que terminar la operación antes de permitir iniciar una nueva. Esto ha sido planificado así de modo que la aplicación no consuma demasiada memoria interna en caso de que se manejen archivos de gran tamaño.

5.2. Arquitectura lógica a implementación

La construcción según la arquitectura lógica MVP mostrada anteriormente en el apartado 4.4 permite que cada componente de la aplicación (modelo, vista, y controlador respectivamente) mantenga cierta independencia de los demás. De este modo, junto al hecho de que la arquitectura se ha implementado de acuerdo a una estructura multicapa, permiten tanto actualizar, modificar y remplazar las diferentes capas, como actualizar, modificar y remplazar ciertos elementos de cada capa sin alterar el resto del programa, lo que además ayuda a varios elementos, como la interfaz, sean reutilizables con pocas e incuso ninguna modificación.

La figura a continuación representa la arquitectura lógica del programa

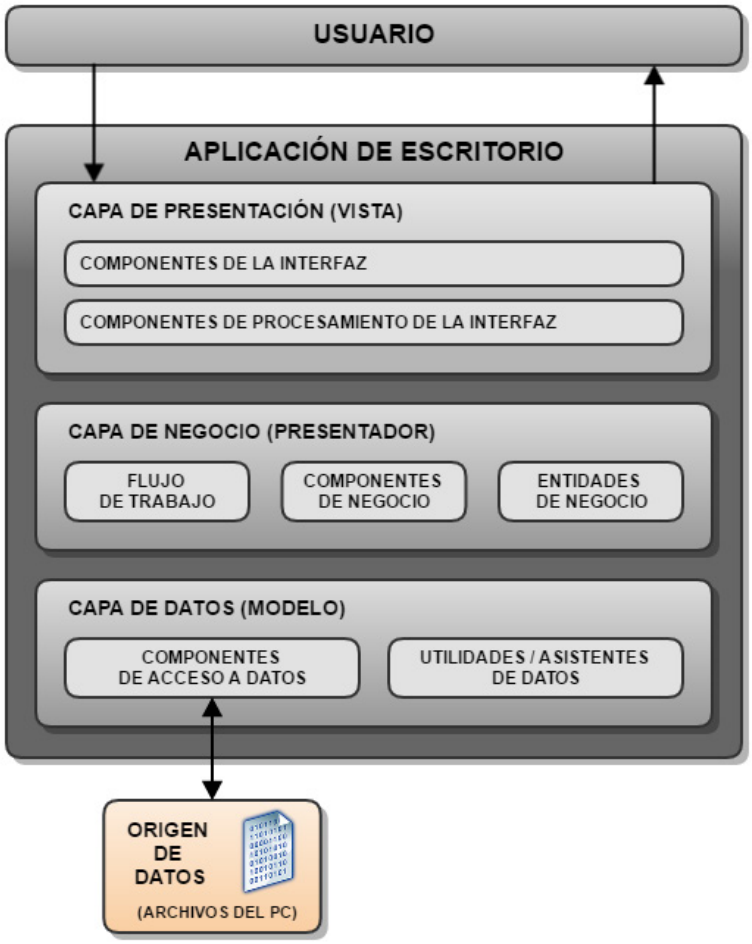


Figura 5.2: Arquitectura lógica

Todas las clases de objetos visibles en los diagramas han sido creadas desde 0 para poder aplicar correctamente nuestra arquitectura MVP y ajustarse a los estándares de seguridad de los diferentes métodos de cifrado; aunque los diferentes métodos de cifrado se derivan de la clase Cipher de java, que es la que proporciona todas las herramientas y especificaciones necesarias para garantizar esta seguridad.

6. Pruebas

6.1. Introducción

Esta sección tratará sobre las diferentes pruebas a las que ha sido sometido el proyecto durante y al finalizar su desarrollo. Existen dos categorías generales para diferencias las pruebas:

- Pruebas de caja blanca: se realizan sobre las distintas funcionalidades de cada módulo/capa de la aplicación a medida que se desarrolla.
- Pruebas de caja negra: se centran en las respuestas del programa tras una simulación de su uso normal.

Estas pruebas no son excluyentes, sino que se realizan ambas por separado de forma complementaria para asegurar el correcto funcionamiento de la aplicación en su conjunto.

6.2. Pruebas de caja blanca

Dado que el número de pruebas de caja blanca está a la par con el número de funciones que contiene la aplicación, son demasiadas para mostrarlas en esta sección, incluso de forma resumida.

Ya que las pruebas de caja blanca se han realizado durante el desarrollo de la aplicación, se ha asegurado que todas ellas resultaban positivas antes de continuar con la siguiente. En el caso de las más complejas se ha intentado poner a prueba el máximo número de variaciones posibles para asegurar que ninguna combinación u orden concreto de órdenes a la aplicación tenga la posibilidad de provocar un error.

Como nota adicional, la inmensa mayoría de las pruebas se han realizado en un sistema operativo Windows 7 con distintas versiones de java (7.0 y 8.0)

6.3. Pruebas de caja negra

Las pruebas de caja negra se centran en verificar que la lectura y escritura de los distintos archivos involucrados (incluyendo las librerías internas y los módulos de cifrado por defecto) se realicen correctamente, así como que el resultado de las operaciones no provoque ningún error en la aplicación o resulte en información equivocada o engañosa.

PCN–1	**Crear librerías y módulos por defecto**
Objetivo:	Comprobar que el programa descomprime y crea su propio directorio una serie de carpetas de archivo con ciertos módulos de cifrado por defecto y sus librerías java, así como otros elementos necesarios para el buen funcionamiento de la aplicación.
Precondición:	El archivo a crear no debe existir.
Resultado esperado:	El programa crea una serie de carpetas localizadas en su directorio y comienza su ejecución normalmente.
Resultado obtenido:	Correcto.

Tabla 6.3.1: PCN–01 — Crear librerías y módulos por defecto

PCN–2	Cargar módulos como clase dinámica
Objetivo:	Comprobar que el programa es capaz de cargar cada módulo individual
Precondición:	Iniciar el programa, o elegir un módulo de cifrado diferente desde el menú.
Resultado esperado:	El programa crea una serie de carpetas localizadas en su directorio y se ejecuta normalmente.
Resultado obtenido:	Correcto.

Tabla 6.3.2: PCN–02 — Cargar módulos como clase dinámica

PCN–3	Leer alfabetos para tipos de cifrado clásico
Objetivo:	Comprobar que el programa es capaz de leer correctamente los alfabetos añadidos en una carpeta con la función específica de almacenarlos
Precondición:	Iniciar el programa.
Resultado esperado:	El programa carga los alfabetos en como un objeto interno.
Resultado obtenido:	Correcto.

Tabla 6.3.3: PCN–03 — Leer alfabetos para tipos de cifrado clásico

PCN–4	Cambiar el idioma de la aplicación
Objetivo:	Comprobar que el programa es capaz de cambiar el idioma de la interfaz de forma dinámica.
Precondición:	Elegir el nuevo idioma en el botón de opción desplegable.
Resultado esperado:	Los diferentes textos de la aplicación cambian inmediatamente al idioma elegido.
Resultado obtenido:	Correcto.

Tabla 6.3.4: PCN–04 — Cambiar el idioma de la aplicación

PCN–5	Mostrar información del cifrado
Objetivo:	Comprobar que el programa es capaz de mostrar un archivo con una explicación del funcionamiento del tipo de cifrado en uso, en el idioma actual del programa
Precondición:	Pulsar el botón “Información”
Resultado esperado:	El programa crea un archivo temporal e intenta ejecutarlo a su vez con el lector pdf predeterminado del sistema operativo.
Resultado obtenido:	Correcto.

Tabla 6.3.5: PCN–05 — Mostrar información del cifrado

PCN–6	Cargar archivo de texto
Objetivo:	Comprobar que el programa es capaz de cargar un archivo de texto plano en la aplicación.
Precondición:	Pulsar el botón “Cargar Archivo de Texto”
Resultado esperado:	El programa muestra una ventana de navegación de directorios, y permite seleccionar archivos, de extensión “.txt” por defecto, de no más de 1 MB para cargar su contenido en el cuadro de texto principal.
Resultado obtenido:	Correcto.

Tabla 6.3.6: PCN–06 — Cargar archivo de texto

PCN–7	Guardar archivo de texto
Objetivo:	Comprobar que el programa es capaz de guardar en un archivo el texto plano contenido en la aplicación.
Precondición:	Pulsar el botón “Guardar Archivo de Texto”
Resultado esperado:	El programa muestra una ventana de navegación de directorios, y permite crear un archivo, de extensión “.txt” por defecto, con el contenido del cuadro de texto principal.
Resultado obtenido:	Correcto.

Tabla 6.3.7: PCN–07 — Guardar archivo de texto

PCN–8	Cargar codificación de un archivo de texto
Objetivo:	Comprobar que el programa es capaz de detectar automáticamente la codificación de texto de un archivo que esta siendo cargado en la aplicación.
Precondición:	Cargar correctamente un archivo de texto (consultar PCN–6)
Resultado esperado:	El programa detecta la codificación de texto del archivo y cambia automáticamente la que se encontrase en uso.
Resultado obtenido:	Semicorrecto. En ciertas ocasiones detecta una codificación equivocada, pero que comparte la gran mayoría de caracteres con la original, por lo que el texto mostrado es correcto en la mayor parte de los casos.

Tabla 6.3.8: PCN–08 — Cargar codificación de un archivo de texto

PCN–9	Cambiar la codificación del texto principal
Objetivo:	Comprobar que el programa es capaz de modificar la codificación del texto “no cifrado” contenido en el programa.
Precondición:	Elegir la nueva codificación de texto en el botón de opción desplegable.
Resultado esperado:	El programa muestra el texto introducido en la aplicación de acuerdo a la codificación de texto elegida.
Resultado obtenido:	Correcto.

Tabla 6.3.9: PCN–09 — Cambiar la codificación del texto principal

PCN–10	Cargar archivo cifrado
Objetivo:	Comprobar que el programa es capaz de cargar un archivo cifrado previamente mediante el módulo de cifrado que se encuentra actualmente en uso.
Precondición:	Pulsar el botón “Cargar Archivo Cifrado”
Resultado esperado:	El programa muestra una ventana de navegación de directorios, y permite seleccionar archivos, por defecto con una extensión única según su tipo de cifrado, de no más de 1 MB para cargar su contenido en el cuadro correspondiente como texto (plano o hexadecimal).
Resultado obtenido:	Correcto.

Tabla 6.3.10: PCN–10 — Cargar archivo cifrado

PCN–11	Guardar archivo cifrado
Objetivo:	Comprobar que el programa es capaz de guardar en un archivo el resultado de una operación de cifrado
Precondición:	Pulsar el botón “Guardar Archivo de Texto”
Resultado esperado:	El programa muestra una ventana de navegación de directorios, y permite crear un archivo, con una extensión única por defecto según el método de cifrado de cifrado, que contenga el contenido del cuadro de texto encriptado.
Resultado obtenido:	Correcto.

Tabla 6.3.11: PCN–11 — Guardar archivo cifrado

PCN–12	Cifrar Texto
Objetivo:	Comprobar que el programa es capaz de realizar una operación de encriptado sobre un mensaje de texto plano cargado previamente en el programa.
Precondición:	Pulsar el botón “Cifrar”
Resultado esperado:	El programa ejecuta la operación de encriptado según el método de cifrado en uso, y muestra el resultado en el cuadro de texto correspondiente.
Resultado obtenido:	Correcto.

Tabla 6.3.12: PCN–12 — Cifrar Texto

PCN–13	Descifrar Texto
Objetivo:	Comprobar que el programa es capaz de realizar una operación de desencriptado sobre un texto encriptado cargado previamente en el programa.
Precondición:	Pulsar el botón “Descifrar”
Resultado esperado:	El programa ejecuta la operación de desencriptado según el método de cifrado en uso, y muestra el resultado en el cuadro de texto correspondiente.
Resultado obtenido:	Correcto.

Tabla 6.3.13: PCN–13 — Descifrar Texto

PCN–14	Cifrar Archivo
Objetivo:	Comprobar que el programa es capaz de realizar una operación de encriptado sobre un archivo a elección del usuario.
Precondición:	Pulsar el botón “Cifrar Archivo”
Resultado esperado:	El programa muestra una ventana de navegación de directorios, y permite seleccionar un archivo cualquiera sobre el que realizar una operación de encriptado según el método de cifrado en uso. Al terminar, crea un nuevo archivo con una extensión predeterminada en el mismo directorio donde se encontraba el archivo original.
Resultado obtenido:	Correcto.

Tabla 6.3.14: PCN–14 — Cifrar Archivo

PCN–15	Descifrar Archivo
Objetivo:	Comprobar que el programa es capaz de realizar una operación de desencriptado sobre un archivo previamente cifrado por el mismo método.
Precondición:	Pulsar el botón “Descifrar Archivo”
Resultado esperado:	El programa muestra una ventana de navegación de directorios, y permite seleccionar un archivo sobre el que realizar una operación de encriptado según el método de cifrado en uso. Al terminar, crea un nuevo archivo resultante en el mismo directorio donde se encontraba el archivo original.
Resultado obtenido:	Correcto.

Tabla 6.3.15: PCN–15 — Descifrar Archivo

PCN–16	Criptoanalizar Texto
Objetivo:	Comprobar que el programa es capaz de realizar una operación de criptoanálisis sobre un texto previamente cifrado por el mismo método de cifrado.
Precondición:	Pulsar el botón “Extraer Clave”
Resultado esperado:	El programa realiza una operación de criptoanálisis según el método de cifrado en uso sobre el contenido del cuadro de texto cifrado de la aplicación. Al terminar, crea un nuevo archivo “log” con el resultado de la operación, así como una lista de las claves más probables calculadas.
Resultado obtenido:	Semicorrecto. Aunque los resultados han sido correctos, en determinadas situaciones el tiempo de respuesta es demasiado elevado. Únicamente se ha implementado esta opción en los métodos de cifrado clásico, los más inseguros.

Tabla 6.3.16: PCN–16 — Criptoanalizar Texto

PCN–17	Criptoanalizar Archivo
Objetivo:	Comprobar que el programa es capaz de realizar una operación de criptoanálisis sobre un archivo previamente cifrado por el mismo método de cifrado.
Precondición:	Pulsar el botón “Extraer Clave de Archivo”
Resultado esperado:	El programa realiza una operación de criptoanálisis según el método de cifrado en uso sobre el contenido de un archivo. Al terminar, crea un nuevo archivo “log” con el resultado de la operación, así como una lista de las claves más probables calculadas.
Resultado obtenido:	Semicorrecto. Aunque los resultados han sido correctos, en determinadas situaciones el tiempo de respuesta es demasiado elevado. Únicamente se ha implementado esta opción en los métodos de cifrado clásico, los más inseguros.

Tabla 6.3.17: PCN–17 — Criptoanalizar Archivo

PCN–18	Crear Clave/Par de claves segura(s)
Objetivo:	Comprobar que el programa es capaz de añadir una clave segura al método de cifrado en uso.
Precondición:	Pulsar el botón “Crear clave”
Resultado esperado:	El programa crea un nuevo archivo en el directorio de recursos del programa, y después procede a añadirlo a la lista dinámica de claves.
Resultado obtenido:	Correcto.

Tabla 6.3.18: PCN–18 — Crear Clave/Par de claves segura(s)

PCN–19	Eliminar Clave/Par de claves
Objetivo:	Comprobar que el programa es capaz de eliminar una clave del método de cifrado en uso.
Precondición:	Pulsar el botón “Eliminar clave”
Resultado esperado:	El programa elimina el archivo elegido del directorio de recursos del programa, y después procede a actualizar también la lista de claves.
Resultado obtenido:	Correcto.

Tabla 6.3.19: PCN–19 — Eliminar Clave/Par de claves

PCN–20	Exportar Clave/Par de claves
Objetivo:	Comprobar que el programa es capaz de exportar una clave del método de cifrado en uso.
Precondición:	Pulsar el botón “Exportar clave” del cuadro “Importar/Exportar”
Resultado esperado:	El programa crea una copia de la clave elegida en una carpeta mediante un menú de navegación por directorios.
Resultado obtenido:	Correcto.

Tabla 6.3.20: PCN–20 — Exportar Clave/Par de claves

PCN–21	Importar Clave/Par de claves
Objetivo:	Comprobar que el programa es capaz de importar una clave del método de cifrado en uso.
Precondición:	Pulsar el botón “Importar clave” del cuadro “Importar/Exportar”
Resultado esperado:	El programa crea una copia del archivo elegido en la carpeta de recursos del programa.
Resultado obtenido:	Correcto.

Tabla 6.3.21: PCN–21 — Importar Clave/Par de claves

PCN–22	Añadir firma digital
Objetivo:	Comprobar que el programa es capaz de añadir una firma digital al resultado de una operación de encriptado.
Precondición:	Realizar una operación de “Cifrar Texto” o “Cifrar Archivo”
Resultado esperado:	El programa añade la firma digital al resultado de la operación de encriptado.
Resultado obtenido:	Correcto.

Tabla 6.3.22: PCN–22 — Añadir firma digital

PCN–23	Verificar firma digital
Objetivo:	Comprobar que el programa es capaz de identificar correctamente la firma digital de un texto o archivo previamente cifrado mediante el sistema en uso
Precondición:	Realizar una operación de “Descifrar Texto” o “Descifrar Archivo”
Resultado esperado:	El programa muestra un mensaje informativo con el resultado de la verificación de la firma digital.
Resultado obtenido:	Semicorrecto. Si el usuario no especifica que se quiere verificar la firma digital en un archivo con firma digital, lo detectará como erroneo.

Tabla 6.3.23: PCN–23 — Verificar firma digital

7. Recursos (/res)

7.1. Introducción

Esta sección exponemos la organización de varios archivos necesarios o complementarios para el correcto funcionamiento de la aplicación.

7.2. Directorios

7.2.1. alphabets

Este directorio contiene los archivos alfabeto con datos como la secuencia completa de caracteres y su entropía (frecuencia de uso en el idioma). Estos archivos son necesarios para varios tipos de cifrado clásico, y de forma predeterminada se incluyen los alfabetos de los idiomas “Español” e “Inglés”.

En la imagen siguiente se muestra un ejemplo de los datos contenidos por un archivo alfabeto de forma legible (los archivos leídos por el programa son datos en formato de objetos Java).

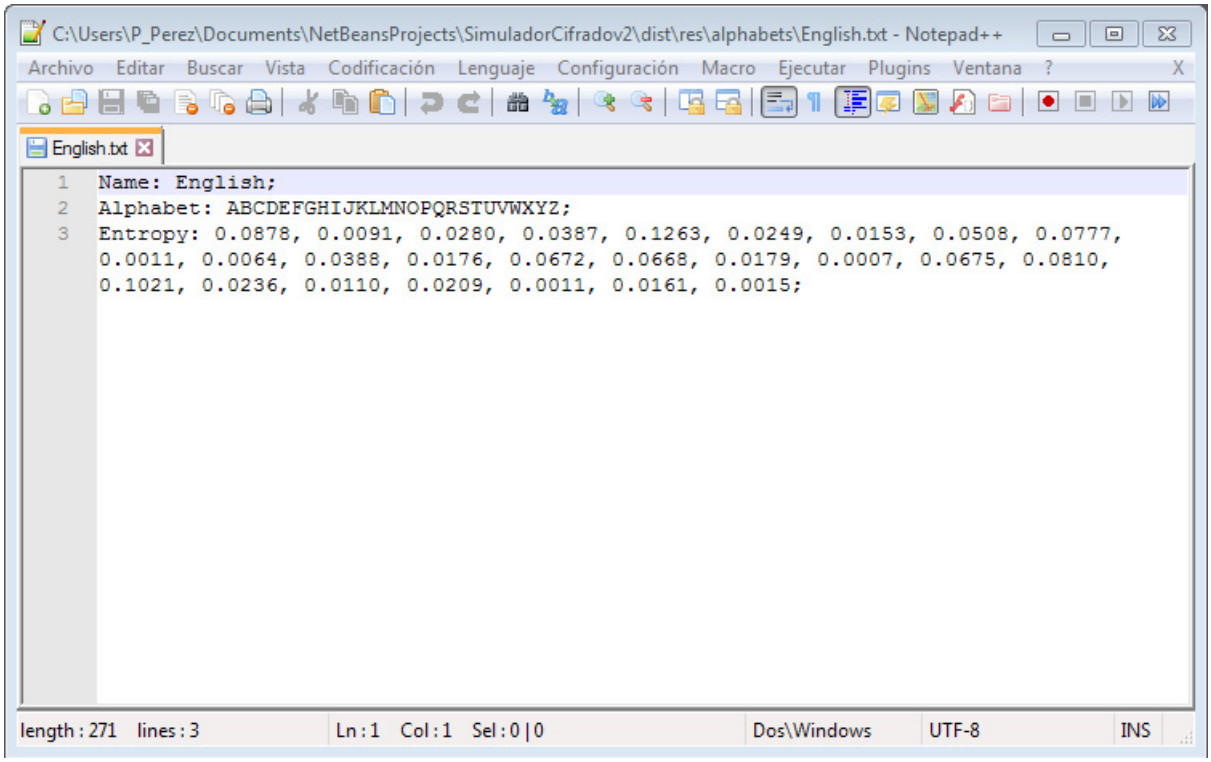


Figura 7.2.1.1: Directorios – alphabets – ejemplo de alfabeto.txt

7.2.2. cyphers

Este directorio contiene un conjunto de subdirectorios que dividen los diferentes módulos de cifrado disponibles. Al iniciarse, el programa lee e identifica los archivos compatibles de estas subdirectorios, proporcionando acceso a los archivos que detecte como módulos de cifrados que pueda cargar en la aplicación principal.

La siguiente imagen muestra un árbol de directorios desde la carpeta principal que contiene el programa hasta un directorio final mostrando su módulo de cifrado contenido como ejemplo:

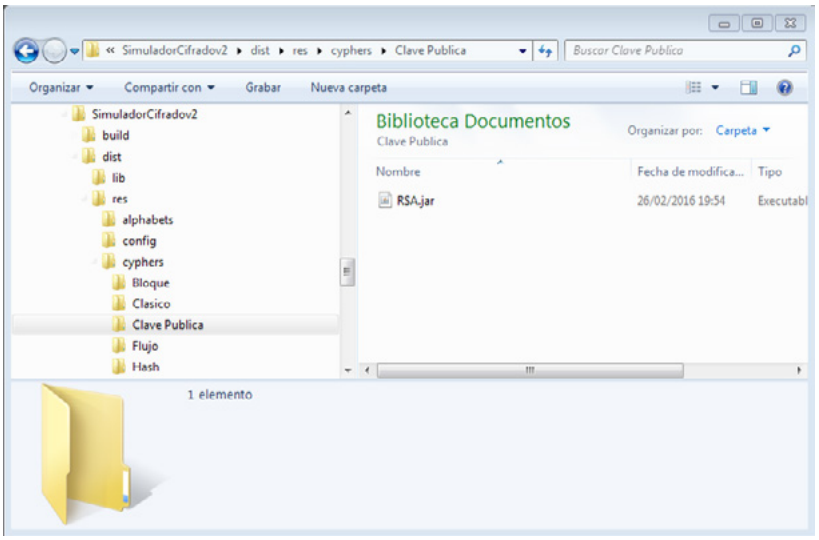


Figura 7.2.2: Directorios – cyphers

7.2.3. keys

Este directorio contiene un conjunto de claves, empleadas por los módulos de cifrado más complejos que no pueden producir un encriptado seguro solo a partir de palabras clave.

La siguiente imagen muestra un árbol de directorios desde la carpeta principal que contiene el programa hasta el directorio final mostrando algunos de estos archivos clave como ejemplo:

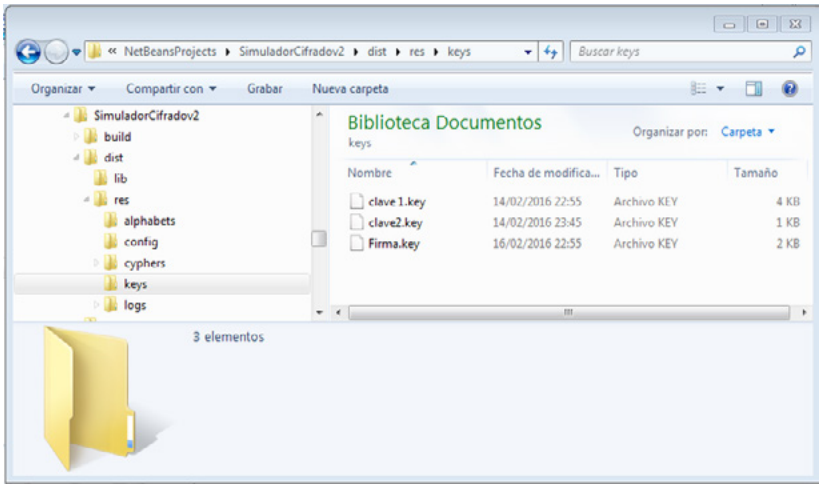


Figura 7.2.3: Directorios – keys

7.2.4. logs

Este directorio contiene los resúmenes en texto de actividades complejas o de larga duración que pueda realizar el programa. En un primer momento se ha establecido para almacenar los registros de tareas de criptoanálisis.

En las siguientes imágenes se puede ver un árbol de directorios desde la carpeta principal que contiene el programa hasta el directorio final mostrando algunos de estos archivos de registro, así como algunos ejemplos de su contenido:

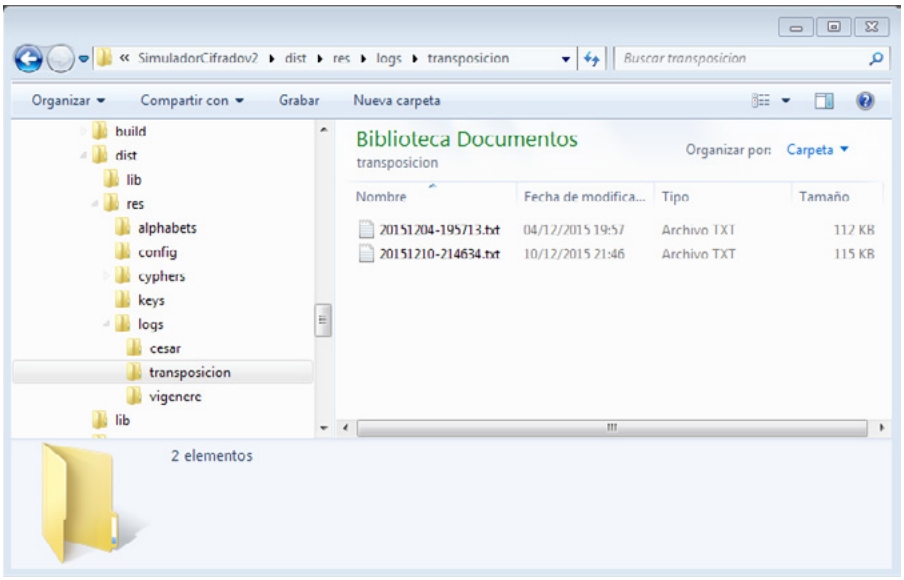


Figura 7.2.4.1: Directorios – logs – ejemplo de log: césar.

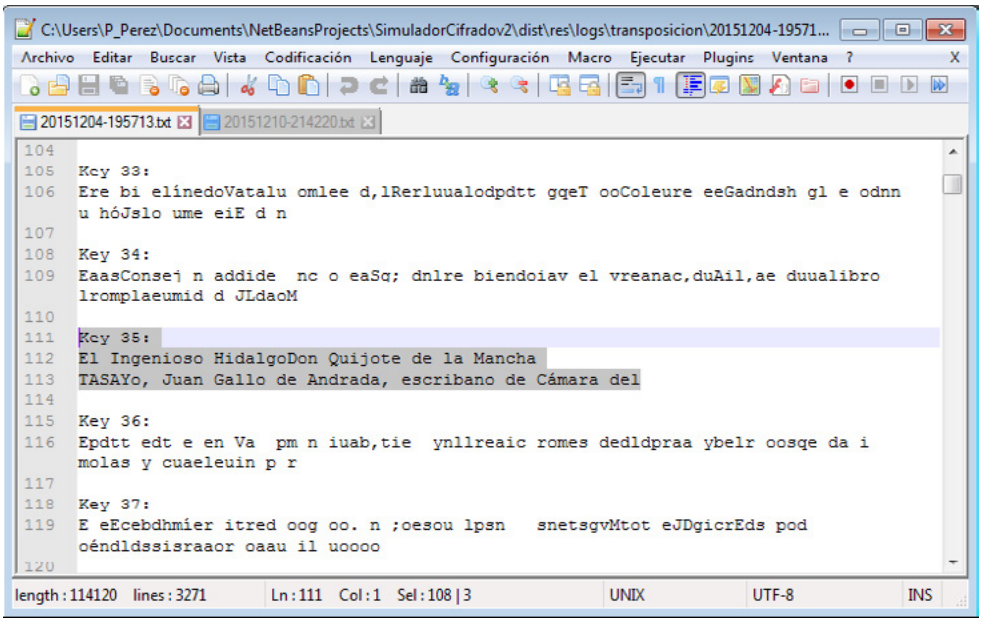


Figura 7.2.4.2: Directorios – logs – ejemplo de log: transposición.

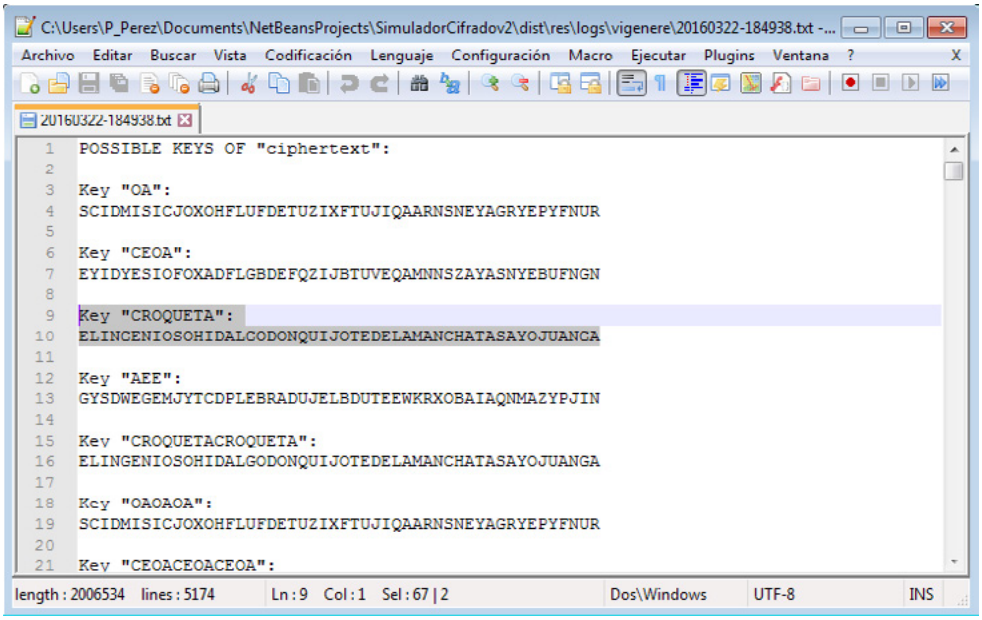


Figura 7.2.4.3: Directorios – logs – ejemplo de log: vigenere.

8. Manuales

8.1. Manual de instalación

Al ser totalmente portable y multiplataforma no es necesario realizar ningún proceso de instalación propiamente dicho. El propio archivo ejecutable proporciona cualquier archivo externo necesario para su ejecución.

A continuación se muestran una lista de requisitos necesarios para el uso del programa:

	Requisitos mínimos	Requisitos recomendados
Procesador	Procesador de 32 bits (x86) o 64 bits (x64) con dos núcleos a 1.8 gigahercios (GHz) o más.	Procesador de 32 bits (x86) o 64 bits (x64) con dos núcleos a 2.6 gigahercios (GHz) o más.
Memoria RAM	1 gigabyte (GB)	4 gigabytes (GB)
Espacio en disco duro	Entre 10-15 MB de espacio disponible.	Entre 20-25 MB de espacio disponible.
Versión de Java	Java 1.7 o superior	Java 1.7 o superior

Tabla 8.1: Manual de Instalación — Requisitos

8.2. Manual de usuario

Introducción

Para comodidad de los usuarios se ha construido la aplicación de modo que su uso sea lo más simple e intuitivo posible, por lo que este compendio de normas de uso pretende ser una guía paso a paso de las distintas posibilidades que ofrece el programa.

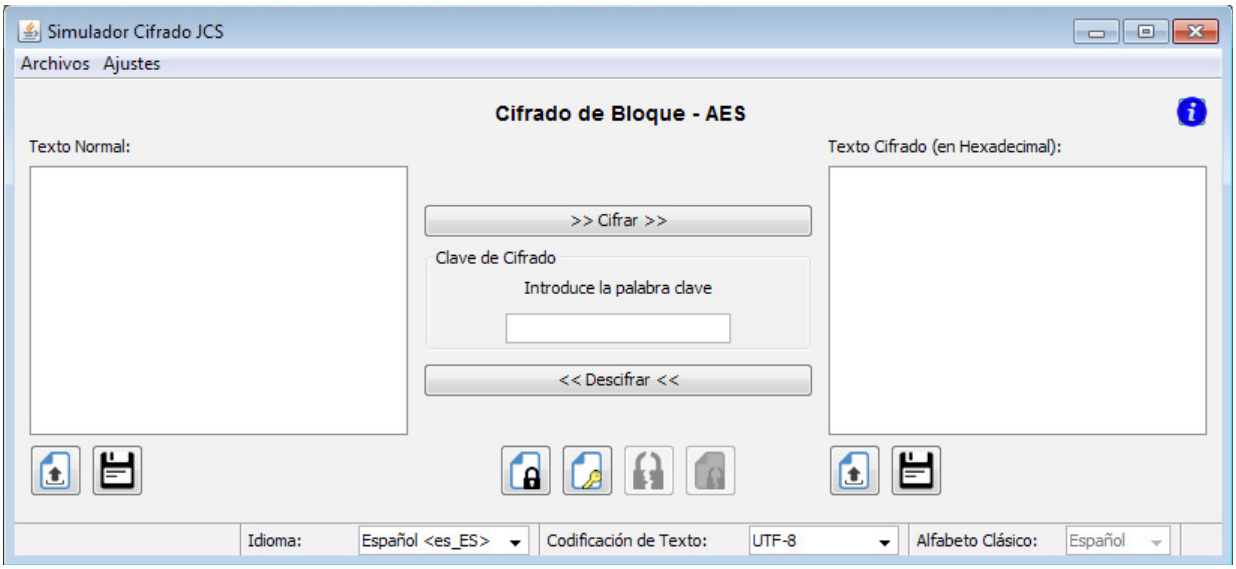


Figura 8.2.1: Manual de Usuario — Interfaz General

Menú Archivo

- **Cifrado:** En este menú se cargan de forma dinámica los accesos a los diferentes módulos de cifrado almacenados en su carpeta raíz, permitiendo al usuario cambiar el método de cifrado que desea utilizar.
- **Salir:** Acceso rápido para cerrar el programa.

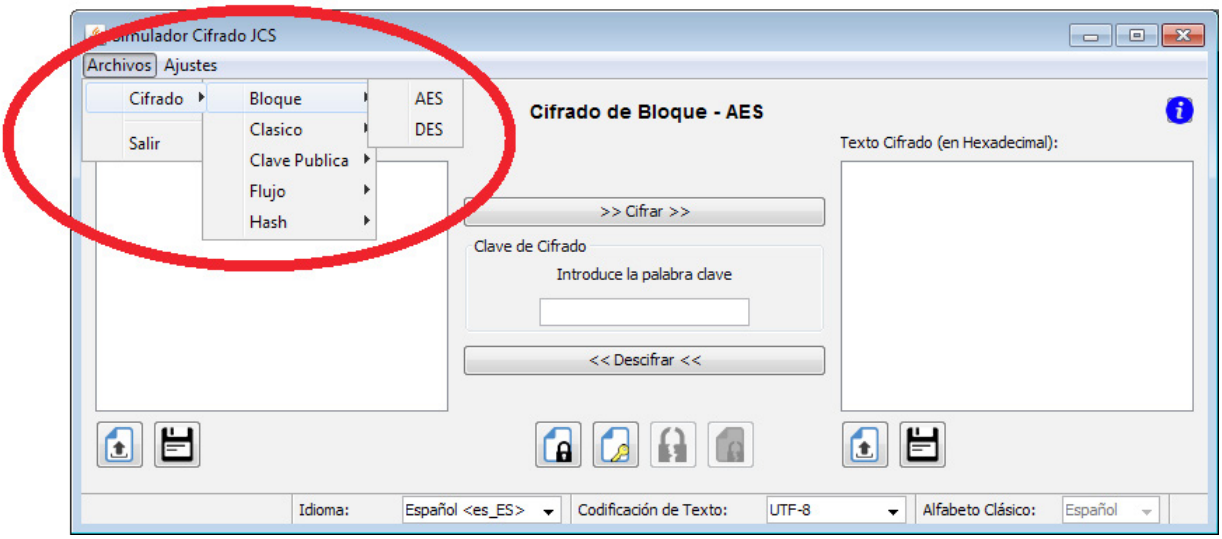


Figura 8.2.2: Manual de Usuario — Menú Archivo

Menú Ajustes

- **Alfabeto:** Permite elegir uno de los alfabetos disponibles, entre los predeterminados del programa y los añadidos en la carpeta correspondiente.

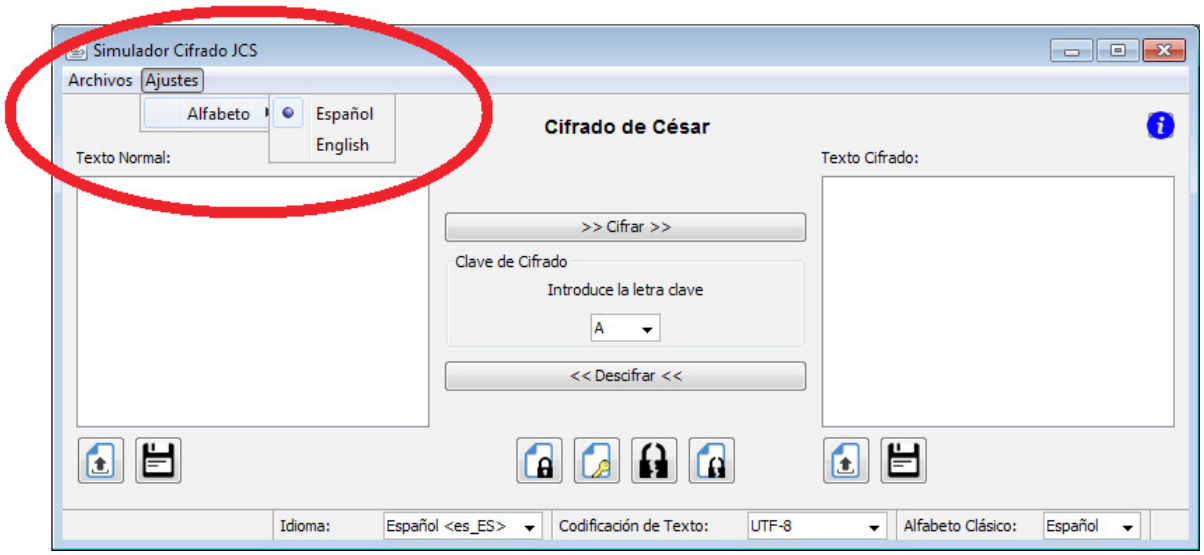


Figura 8.2.3: Manual de Usuario — Menú Ajustes

Barra de Estado

- **1.- Idioma:** Permite elegir el idioma en el que se muestra el programa.
- **2.- Codificación de texto:** Modifica el texto de acuerdo a su codificación por bits/bytes.
- **3.- Alfabeto Clásico:** Permite elegir uno de los alfabetos disponibles, entre los predeterminados del programa y los añadidos en la carpeta correspondiente (ver "Menú Ajustes — Alfabeto" en la página anterior).

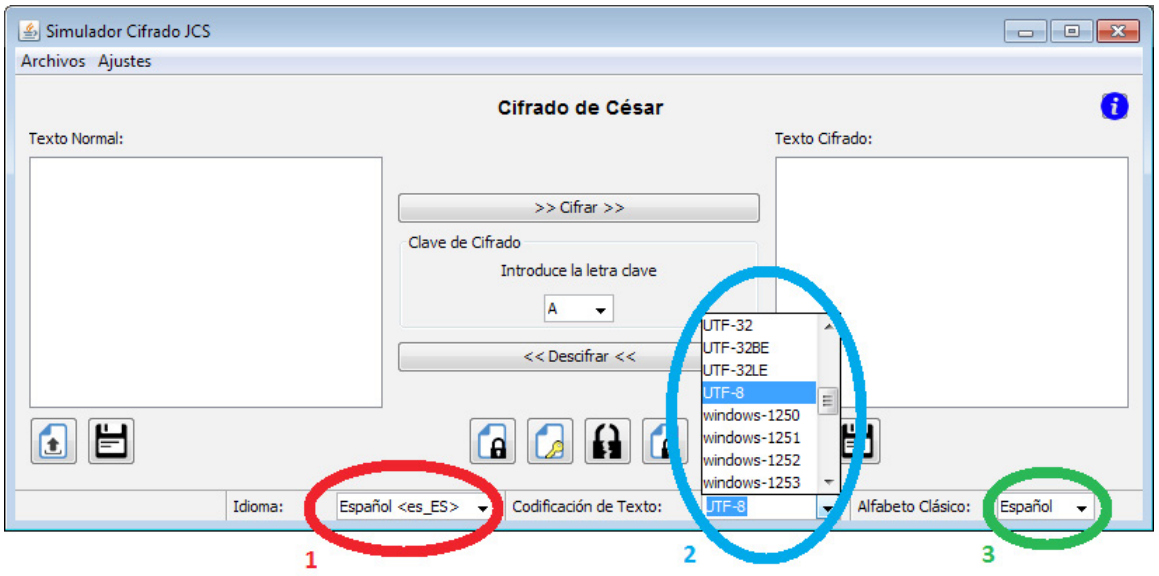
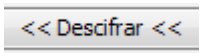


Figura 8.2.4: Manual de Usuario — Barra de Estado

Herramientas de la interfaz interna

- **>> Cifrar >>:** Realiza una operación de encriptado sobre el texto introducido en la aplicación.



- **<< Descifrar <<:** Realiza una operación de desencriptado sobre el texto cifrado cargado en la aplicación previamente cifrado mediante el mismo método de cifrado.

- **Cargar Texto plano/cifrado:** Carga un archivo de texto de no más de 1 MB en la aplicación.

- **Guardar Texto plano/cifrado:** Guarda el texto cargado en la aplicación como un archivo de texto o una cadena de bytes.

- **Cifrar Archivo:** Realiza una operación de encriptado sobre un archivo contenido en el PC.

- **Extraer clave:** Analiza el texto cifrado cargado en la aplicación para extraer la clave más probablemente usada para cifrarlo en un archivo de registro.

- **Crear Clave:** Crea un nuevo archivo clave para utilizar al cifrar/descifrar archivos con la aplicación.

- **Importar/Exportar Clave:** Copia un archivo clave al interior o exterior de los directorios raíz de la aplicación.

- **Descifrar Archivo:** Realiza una operación de desencriptado sobre un archivo contenido en el PC previamente cifrado mediante el mismo método de cifrado.

- **Extraer clave de archivo:** Analiza un archivo cifrado contenido en el PC para extraer la clave más probablemente usada para cifrarlo en un archivo de registro.

- **Borrar Clave:** Borra uno de los archivos clave previamente creados mediante la aplicación.

- **Información:** Carga un archivo .pdf con una explicación sencilla del funcionamiento del tipo de cifrado correspondiente.

Ejemplo de uso típico

Por defecto el programa cargará el primer método de cifrado compatible que detecte, por lo que podemos asumir que siempre habrá un método de cifrado en uso. En este caso, *nuestro método cargado por defecto es el método de cifrado de Bloque – AES*.

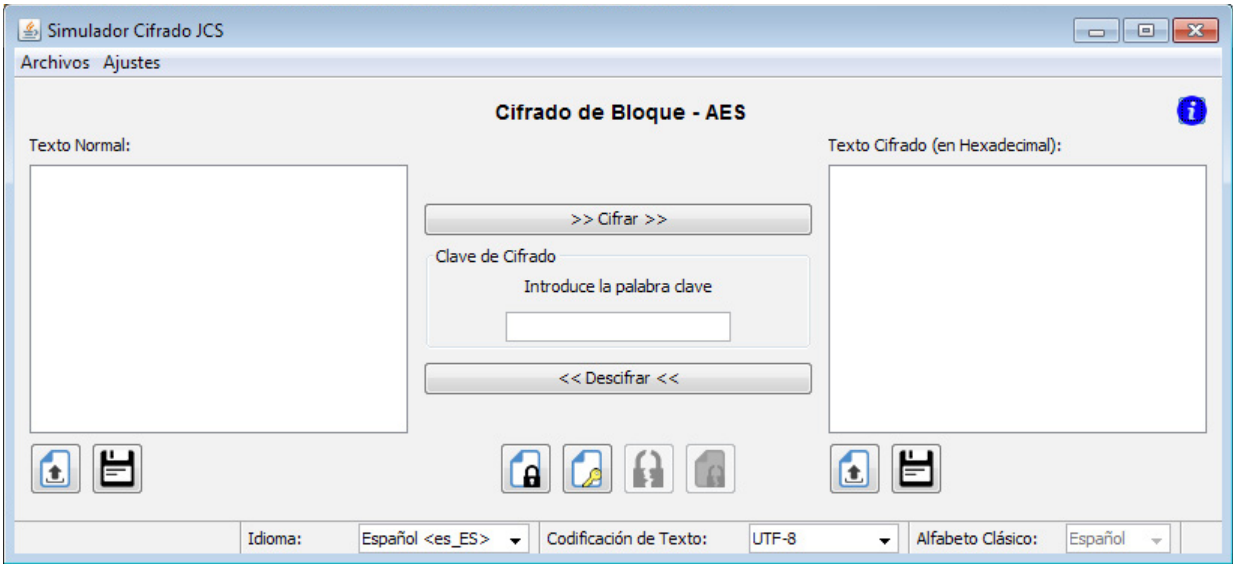


Figura 8.2.5: Manual de Usuario — Interfaz AES

Asumimos que el usuario no conoce este método de cifrado, por lo que antes de hacer nada *pulsará el botón de información* para aclarar sus dudas; en pocos segundos se muestra en pantalla un documento pdf (usando el lector asociado a archivos .pdf, o solicitando un programa para abrir el documento).

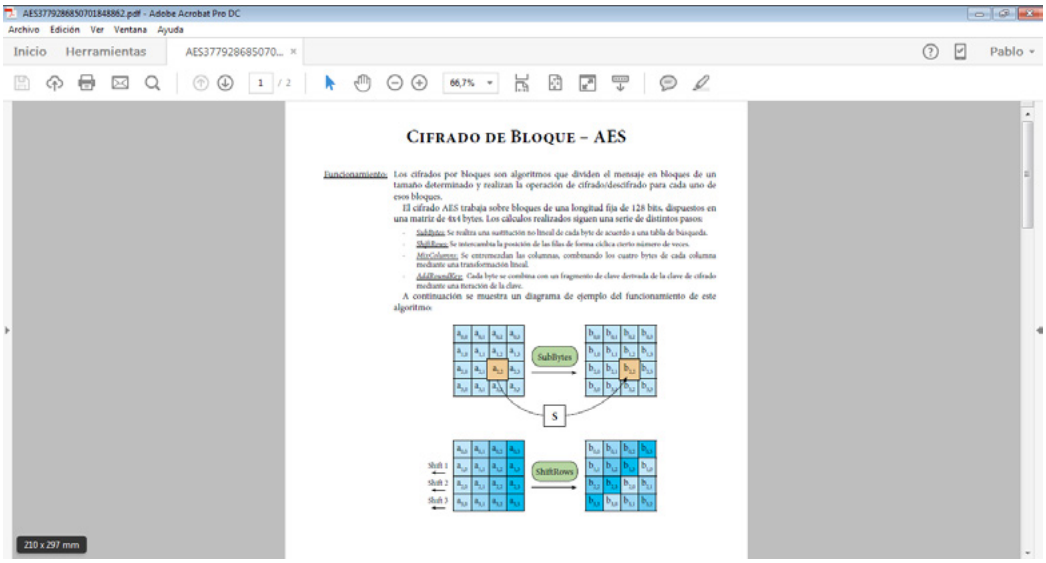


Figura 8.2.6: Manual de Usuario — Documento Información

Ahora que el usuario ha resuelto sus dudas está preparado para realizar algunas pruebas sencillas. Los elementos mejor dispuestos para ello son los cuadros de texto de la aplicación, por lo que es de esperar que intente *cifrar mensajes de texto* para ver por si mismo los resultados.

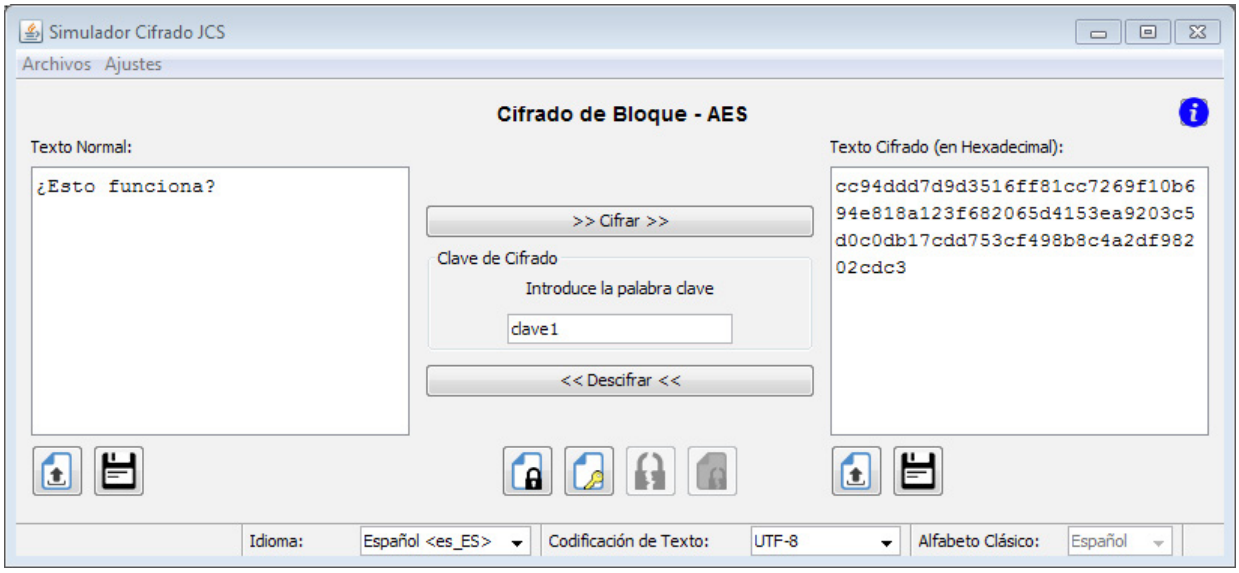


Figura 8.2.6: Manual de Usuario — Cifrar texto

Aunque el programa nos muestra un resultado, a primera vista es difícil decir si en verdad el texto ha sido cifrado correctamente. Como prueba, el usuario decide cambiar la clave antes de *pulsar el botón "descifrar"*, lo que provoca el siguiente resultado.

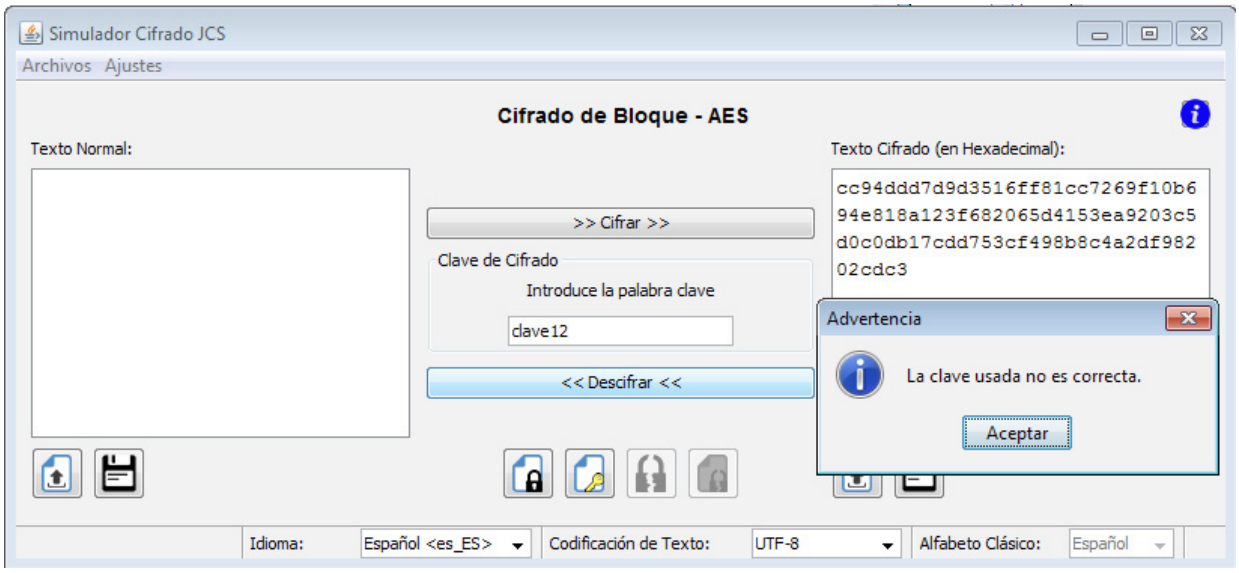


Figura 8.2.7: Manual de Usuario — Descifrar texto incorrecto

Aparentemente el programa reconoce correctamente la clave usada, lo que es suficiente para calmar las dudas del usuario sobre el buen funcionamiento del programa.

Ahora, es probable que el usuario pretenda *encriptar archivos* de su ordenador para mantenerlos a salvo, ya sea porque contienen información personal, privada, o para presumir de seguridad.

El resultado es un nuevo archivo en el directorio en el que se encontraba el original. Del mismo modo, al intentar *desencriptar el archivo* este se creará de nuevo en el mismo directorio en el que se hallaba el archivo cifrado. Por seguridad, si el programa detecta que va a borrar algún archivo en el proceso, solicita la confirmación del usuario antes de continuar.

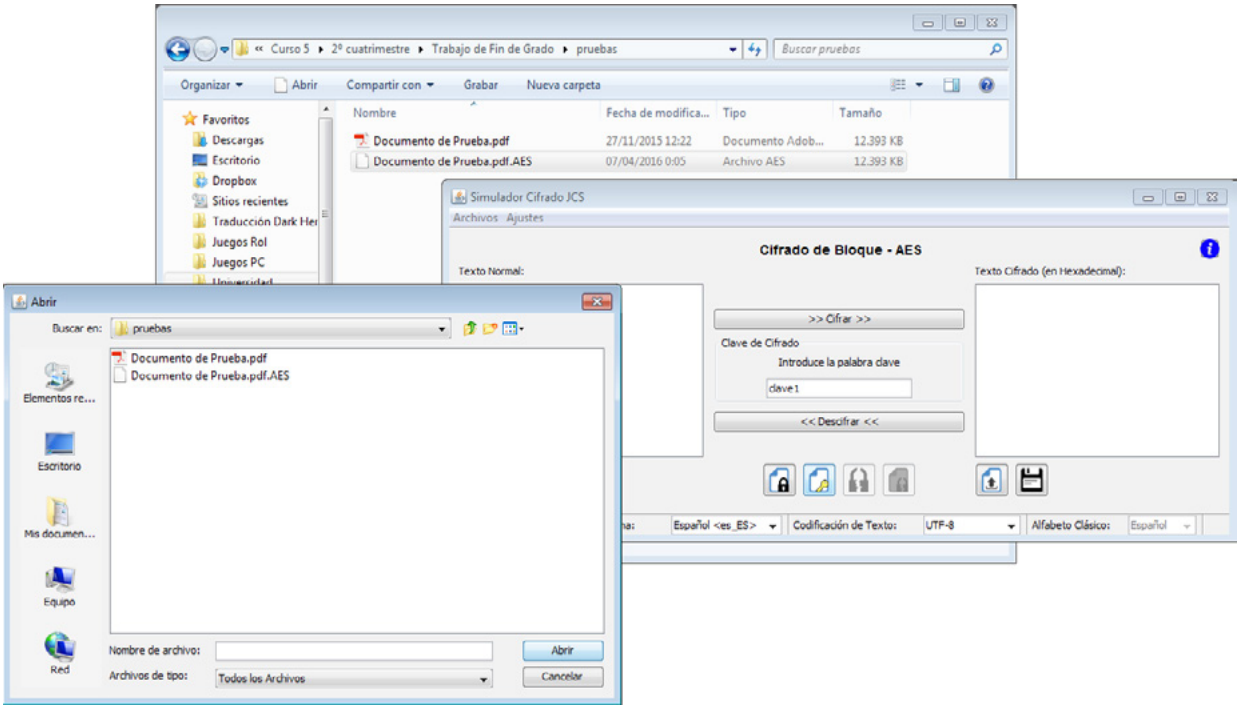


Figura 8.2.8: Manual de Usuario — Cifrar/Descifrar Archivo

Otros métodos de cifrado ofrecen otras opciones, como es el caso de la firma digital en el cifrado por clave pública. En este caso la aplicación preguntará al usuario si desea añadir una firma al texto/archivo que va a encriptar, o si desea verificar que un texto/archivo que va a desencriptar tiene una firma digital concreta, cada vez que realice una de esas operaciones (encriptar o desencriptar).

Las siguientes imagen muestra la elección de una firma digital para el cifrado por clave pública, y el resultado de una verificación correcta de una firma digital.

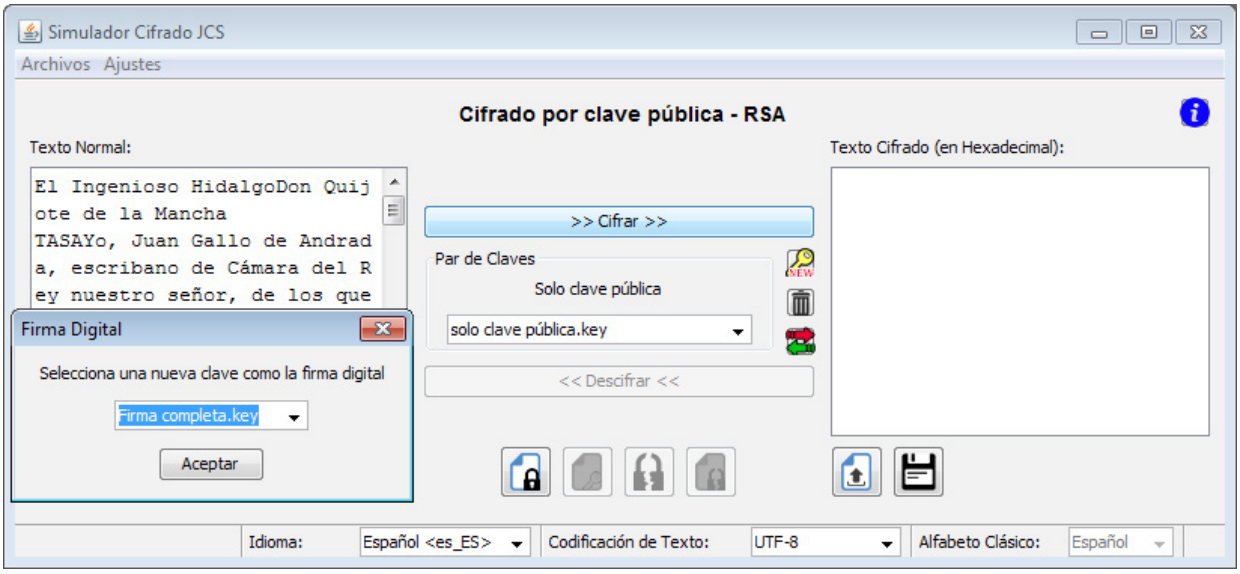


Figura 8.2.9: Manual de Usuario — Elegir Firma Digital

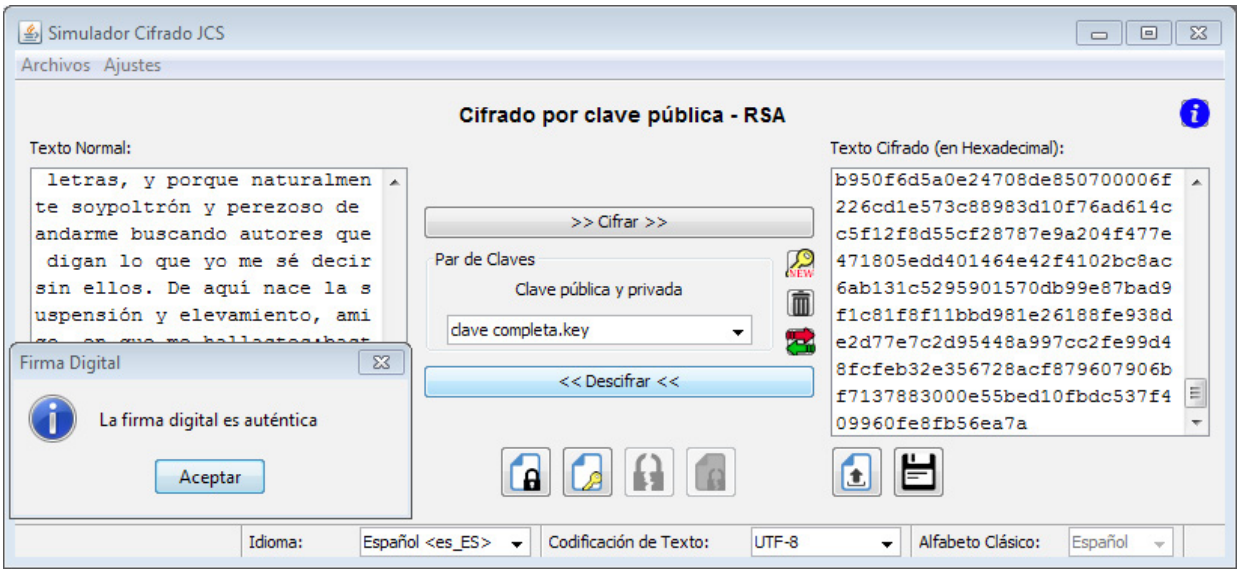


Figura 8.2.10: Manual de Usuario — Verificar Firma Digital

Tras probar todos estos comandos del programa, es posible que el usuario explore las posibilidades que ofrecen otros métodos de cifrado, como la posibilidad de extraer una clave por la fuerza que concede para cifrados clásicos.

En la siguiente imagen tenemos, a la izquierda, el texto sin cifrar (un pequeño fragmento del quijote), a la derecha, el texto cifrado mediante el cifrado de César con la letra clave P (empleando el alfabeto español), y abajo, un archivo de texto como resultado de la operación (se muestra en pantalla automáticamente), el contenido de este archivo indica el porcentaje (valor entre 0–1) de posibilidades que tiene una letra de ser la clave usada, mostradas en orden descendente.

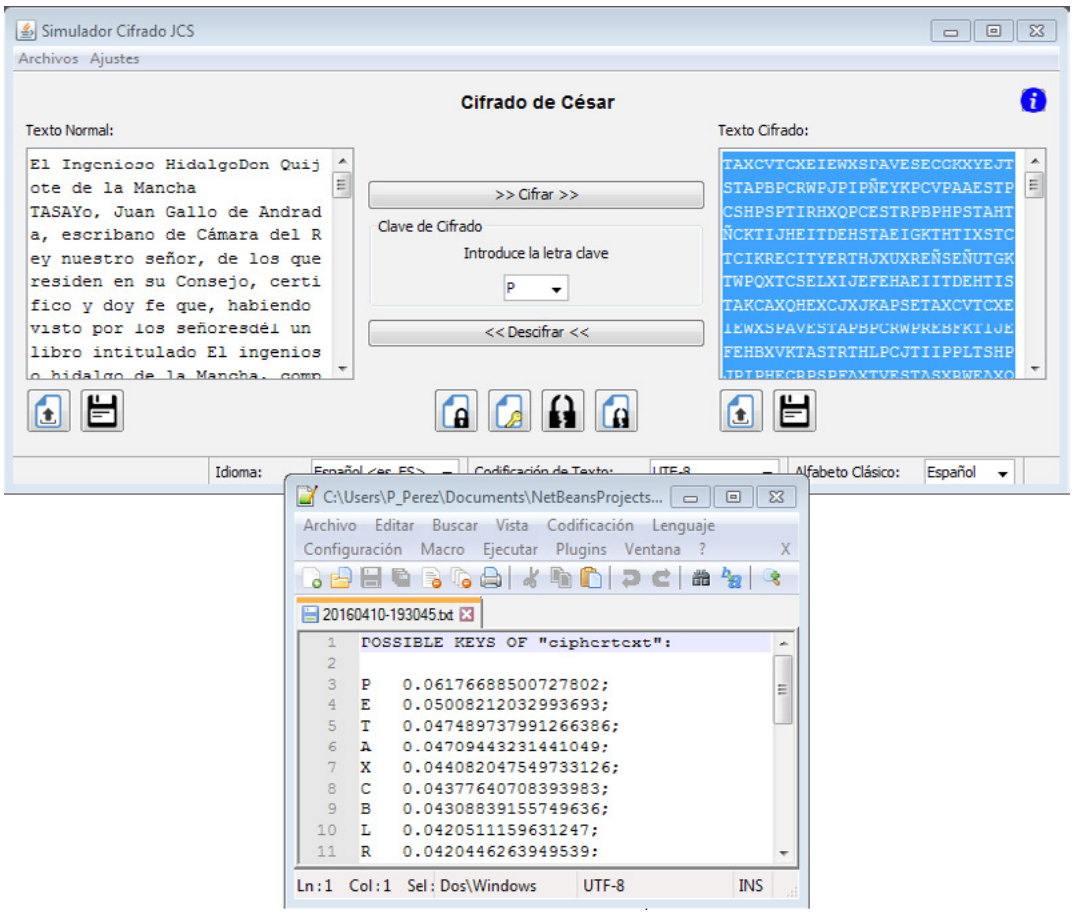


Figura 8.2.11: Manual de Usuario — Extraer Clave de un texto cifrado

Al realizar la misma prueba con texto completo, pero encriptado como archivo, el usuario puede comprobar que el resultado es muy similar al original.

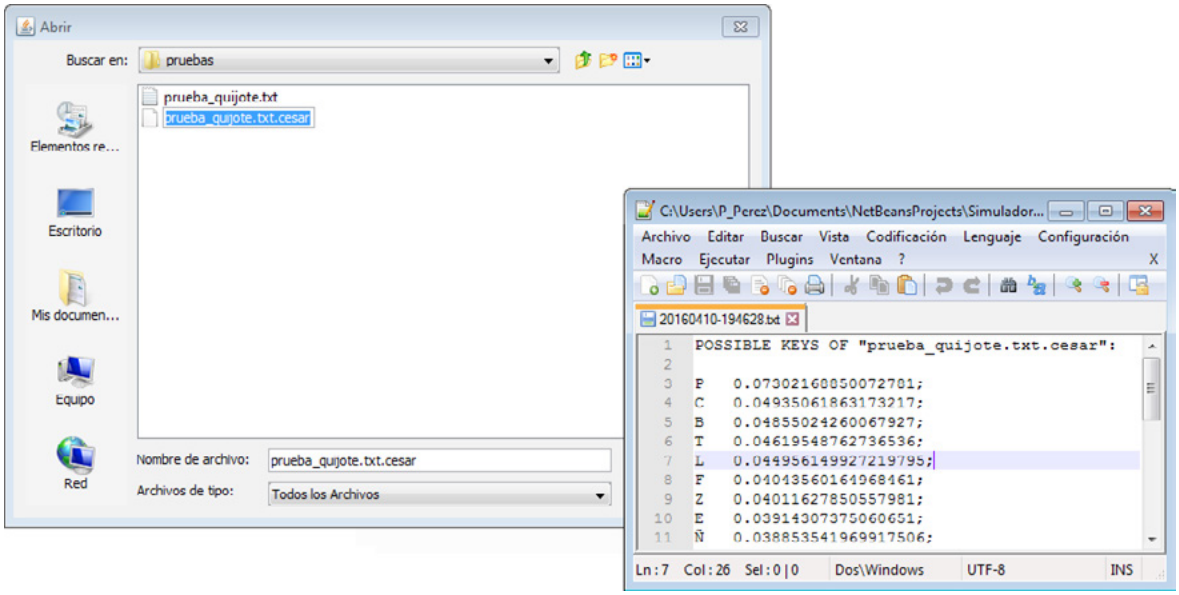


Figura 8.2.12: Manual de Usuario — Extraer Clave de un archivo cifrado

Por último, es posible que el usuario no esté conforme con los métodos de cifrado que proporciona el programa de forma predeterminada, y quiera descargar algún método nuevo que los desarrolladores pongan a su disposición.

En la siguiente imagen podemos ver la lista de métodos de cifrado que proporciona el programa de forma predeterminada, y el directorio reservado en el que los guarda.

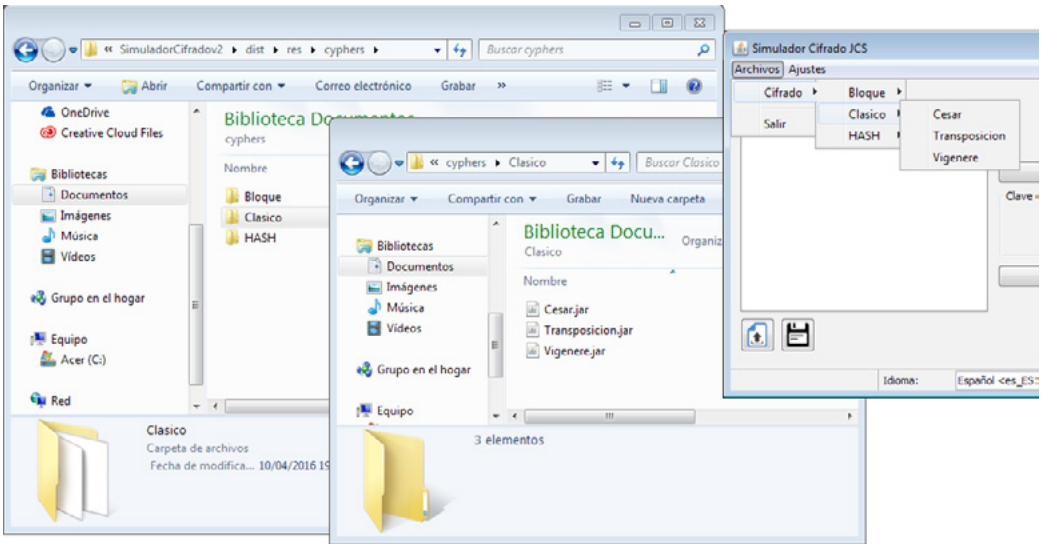


Figura 8.2.13.1: Manual de Usuario — Añadir método de cifrado (1)

Para añadir un método de cifrado nuevo, el usuario solo debe seguir las instrucciones; en este caso, añadir el archivo en el directorio reservado, dentro de una organización de subdirectorios que serán añadidos de manera automática al menú del programa.

En la siguiente imagen se muestra cómo hemos añadido el nuevo método de cifrado al directorio reservado, y cómo ahora aparece en el menú del programa.

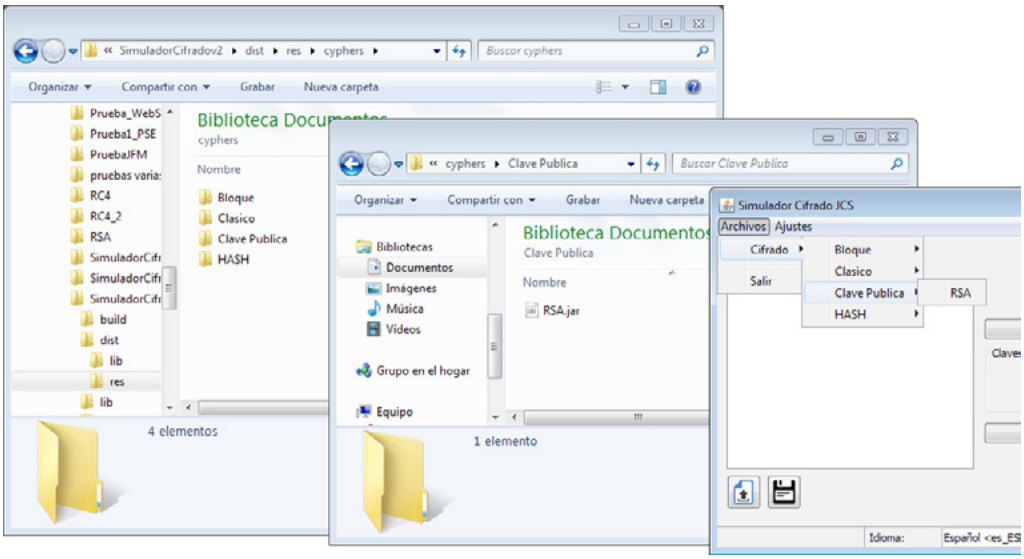


Figura 8.2.13.2: Manual de Usuario — Añadir método de cifrado (2)

9. Conclusiones y trabajos futuros

En este apartado expresaremos las conclusiones a las que se ha llegado a partir de la realización del proyecto y los logros conseguidos, además de posibles líneas de acción para complementar, modificar, actualizar o expandir las funcionalidades de este trabajo.

9.1. Dificultades

La principal dificultad planteada por la aplicación desde el primer momento ha sido la adecuación a los estándares de cifrado actuales para garantizar la máxima seguridad posible de sus métodos de cifrado disponibles. Las librerías "security" y "cipher" de Java proporcionan un gran conjunto de herramientas para este propósito que agilizaron la mayor parte de este proceso, y el hecho de tratarse de una aplicación de escritorio sin conexiones web ni con otros dispositivos nos asegura que el mensaje no sea interceptado por atacantes maliciosos en mitad del proceso de cifrado/descifrado; sin embargo, al tratarse de herramientas generalistas para multitud de métodos de cifrado la seguridad de los mismos nunca está garantizada al 100% si su implementación se realiza de forma incorrecta, lo que se traduce en un gran trabajo de documentación teórica para cada método diferente implementado.

Otra de las mayores dificultades fue simplificar el proceso de cifrado/descifrado de cara al usuario, ya que muchos de los diferentes métodos implican tratar con claves complejas, cadenas pseudoaleatorias de bytes necesarias en ciertos procesos concretos, y cálculos donde a priori sería necesaria una especificación variable en función de ciertos pasos previos. Algunos de estos se solucionaron estableciendo parámetros fijos suficientemente seguros en las clases de cifrado, mientras que en otros se automatizaron o se añadió información estandarizada (lo más transparente posible ante análisis) a los textos/archivos cifrados de forma que se asegurase su recuperación.

Por último, dentro de las dificultades de consideración, se encuentra la construcción de arquitectura modular MVP. Al no haber trabajado nunca con arquitecturas complejas su proceso de construcción requirió de una inversión de tiempo inicial bastante alta en investigación teórica previa así como gran cantidad de pruebas independientes al proyecto para entender el funcionamiento y posibilidades que ofrece esa metodología. Mantener la independencia entre las diferentes capas MVP de un único módulo requirió menos tiempo y se realizó de forma más simple; sin embargo, aislar y separar los diferentes módulos de cifrado de la aplicación principal requirió entre otras cosas de un largo proceso de prueba y error, así como establecer ciertas clases "puente" comunes a los módulos y al programa principal que permitieran cargar los módulos a petición del programa.

9.2. Conclusiones

Como resultado de todo lo expuesto hasta ahora en el documento el resultado es una aplicación poco potente, pero de uso sencillo y con un gran potencial de escalabilidad y que sienta las bases para añadir fácilmente nuevas funcionalidades desarrollando módulos a partir de una plantilla base, ahorrando en tiempo y esfuerzo para su posterior desarrollo.

9.3. Conocimientos adquiridos

En primer lugar, y con carácter general, me ha permitido indagar en el intrincado mundo de la criptografía y el criptoanálisis, obteniendo experiencia sobre los errores más comunes a la hora de construir un sistema de encriptación segura de mensajes y archivos y algunas de sus vulnerabilidades. Del mismo modo, he aprendido sobre distintos tipos de ataque a métodos de cifrado y también a lidiar con ciertos problemas ineludibles derivados de la naturaleza de la encriptación, como el conflicto entre rapidez y seguridad (cuanta más rapidez en el cifrado menor seguridad, y viceversa).

En segundo lugar, he podido profundizar en el funcionamiento de ciertos aspectos de la maquina virtual de java; sus librerías nativas; sus métodos de carga dinámica de librerías, paquetes y clases; y sus efectos específicos sobre procesos ya en ejecución.

En tercer lugar he aprendido a aprovechar las ventajas del lenguaje de programación Java: mantener su capacidad multiplataforma evitando realizar llamadas al sistema operativo, sacar partido de sus librerías nativas para la carga de objetos externos y lectura de objetos internos compilados, y también explorar las posibilidades de librerías java de licencia gratuita creadas por su inmensa comunidad de desarrolladores.

9.4. Ampliaciones futuras

Añadir nuevos métodos de cifrado

La propia construcción del programa permite añadir un número virtualmente ilimitado de métodos de cifrado, e incluso módulos con otras utilidades diferentes, siempre que se sigan las pautas establecidas en la plantilla compatible que se encuentra disponible junto con este documento.

Automatizar procesos

Casos concretos como, por ejemplo, verificar una firma digital al desencriptar un mensaje pueden ser automatizados de modo que el usuario no tenga que verificar cada operación en cada ocasión. Si se llevara a cabo esta mejora, la aplicación podría simplemente comprobar la operación con las claves disponibles y mostrar al usuario el resultado final (si el mensaje se ha desencriptado correctamente o no, si tenía firma digital, y en este último caso mostrar el usuario/clave asociada guardada en nuestro pc que ha verificado esa firma digital).

Posibles mejoras al programa

En nuestro afán por simplificar el uso del programa se han pasado por alto ciertas funcionalidades que serían de utilidad para usuarios con mayores conocimientos de informática o criptografía. Algunas de estas posibles utilidades, entre otras, serían:

- **Configuración de los métodos de cifrado:** Añadir la personalización de ajustes para los diferentes método de cifrado. Esta mejora corre el riesgo de reducir el nivel de seguridad proporcionado por los métodos de cifrado cuanto más se permita al usuario modificar los ajustes originales por defecto, pero al mismo tiempo abarcaría un mayor número de posibles necesidades de los usuarios.
- **Carga multilenguaje externa:** Aprovechar la estructura de carga de los métodos de cifrado para definir subdirectorios donde almacenar los archivos de traducción de la aplicación. Esto permitiría simplificar también la actualización de nuevos idiomas para la aplicación, así como permitir a los usuarios realizar su propia traducción de los distintos módulos y el programa base.
- **Comunicación por servidor:** Añadir comunicación entre varias máquinas con distintos usuarios conectados en tiempo real, permitiendo el intercambio de archivos o mensajes previo cifrado de los mismos a elección del usuario. Esto podría realizarse tanto en un nuevo módulo independiente como a nivel del programa base.
- **Transformación en app móvil:** La aplicación es lo suficientemente ligera como para que su uso sea fluido en un dispositivo móvil. La división en capas de su arquitectura permitiría eliminar y remplazar de forma sencilla las partes no compatibles como la interfaz gráfica original y aprovechar el resto.

10. Estructura del CD

El CD-ROM que entregado al finalizar la realización de esta práctica se organiza en los siguientes directorios:

10.1. Estructura

- **Aplicación:** Contiene el archivo ejecutable ".jar" del programa compilado, así como el conjunto de directorios que contienen los archivos necesarios para su uso (librerías, alfabetos, módulos de cifrado, etc).
- **Código Fuente:** Contiene el código fuente tanto de la aplicación base como de sus diferentes módulos de cifrado creados, además de la plantilla base usada para crearlos.
- **Diagramas:** Contiene la mayor parte de los diagramas mostrados a lo largo del documento en imágenes en formato ".png" de alta calidad.
  - **Árbol de Características:** Contiene el diagrama de Árbol de características mostrado en el capítulo 1.3 de este documento.
  - **Casos de uso:** Contiene los diferentes casos de uso del programa divididos por los componentes de la interfaz, como aparecen a lo largo del capítulo 3.5 de este documento.
  - **Clases de análisis:** Contiene las diferentes clases de análisis del programa divididas por las capas del patrón arquitectónico utilizado y por los diferentes módulos de cifrado creados, como aparecen en el capítulo 4.6 de este documento.
  - **Interfaz:** Contiene algunos bocetos del aspecto aproximado de la interfaz del programa, como aparece en los capítulos 4.2 y 4.3 de este documento.
  - **Secuencia:** Contiene los diagramas de secuencia más distintivos de las diferentes funcionalidades y aspectos del programa, como aparecen en el capítulo 4.7 de este documento.
  - **Gantt:** Contiene los calendarios y diagramas correspondientes a la planificación de tiempo de este proyecto, como aparecen en los capítulo 2.2 y 2.4 de este documento.
- **Documentación:** Contiene este mismo documento en formato ".pdf".
- **Manuales:** Contiene tanto los manuales de instalación y uso del programa (Capítulo 8 de este documento) como los documentos informativos sobre el funcionamiento de cada diferente módulo de cifrado creados junto a este programa.

Glosario

Término	Definición
Java	Lenguaje de programación de propósito general y orientado a objetos diseñado para tener el menor número posible de dependencias de implementación, permitiendo al mismo programa en condiciones normales ser ejecutado con independencia de la plataforma o sistema operativo.
Encriptado	Traducción literal del término inglés "Encrypt". Su significado literal es realizar una operación de cifrado sobre un mensaje o conjunto de datos para proteger su contenido.
Desencriptado	Traducción literal del término inglés "Decrypt". Su significado literal es realizar una operación de descifrado sobre mensaje o conjunto de datos previamente cifrado haciendo legible su contenido.
Criptoanálisis	Estudio o análisis de un sistema encriptado destinado a encontrar vulnerabilidades en el sistema y romper su seguridad para obtener su información oculta.
Firma digital	Mecanismo criptográfico que permite al receptor de un mensaje o conjunto de datos encriptados verificar la identidad originadora del mensaje.
Codificación de texto	Es el método que identifica una secuencia de bytes determinada como cierto carácter o símbolo concreto de los diferentes alfabetos o conjuntos de símbolos digitales. Algunos ejemplos comunes son: ASCII, UTF–8 o ISO-8859-1.
Apache	(Apache Software Foundation) Organización no lucrativa creada para dar soporte a diferentes proyectos software.
MVP	Modelo–Vista–Presentador. Derivación del patrón arquitectónico "Modelo–Vista–Controlador" donde el peso de la ejecución cae sobre la Interfaz.
UML	Lenguaje Unificado de Modelado. Es el lenguaje de modelado de software más conocido y utilizado. Se emplea para definir un sistema, detallar artefactos del sistema, documentar y construir.

Bibliografía

Apuntes de la asignatura “Protocolos y Comunicaciones Seguras”
— Técnicas criptográficas de protección de datos (3ª Ed.), A. Fúster, Ed. Ra-Ma (2004)
curso 2014–2015, por Jose Ignacio Farrán Martín.

Apuntes de la asignatura “Análisis de Requisitos”
curso 2014–2015 por Miguel Angel Martinez Prieto.

Introduction to Cryptography
— http://crypto.stanford.edu/~dabo/courses/cs255_winter15/syllabus.html
— https://crypto.stanford.edu/~dabo/cs255/syllabus.html
por Dan Boneh (Online, último acceso el 1 de Enero de 2016)

Modelo Constructivo de Costos – COCOMO
— https://es.wikipedia.org/wiki/COCOMO
— https://unpocodejava.wordpress.com/2010/10/14/estimacion-del-esfuerzo-basado-en-puntos-de-funcion-ajustados-3/
(Online, último acceso el 20 de Enero de 2016)

Dynamic Class Loading and Reloading in Java
— http://tutorials.jenkov.com/java-reflection/dynamic-class-loading-reloading.html
(Online, último acceso el 10 de Marzo de 2016)

Anexo A. Casos de uso

ID y Nombre del caso de uso:	**CU–1 — Elegir cifrado**
Dependencias:	Depende del requisito RF–1: Elegir cifrado.
Disparador:	Seleccionar elemento de una lista.
Descripción:	La aplicación muestra una lista con los métodos de cifrado disponibles, organizados por tipo, para que el usuario seleccione el que pretende usar.
Precondiciones:	
Postcondiciones:	POST–1 — La interfaz cambia de acuerdo a las necesidades el método de cifrado elegido.
Flujo normal:	**1.0 Se muestra la lista de métodos de cifrado disponibles** 1. El usuario elige uno de los métodos de cifrado disponibles.
Flujos alternativos:	
Excepciones:	
Prioridad:	Alta
Frecuencia de uso:	1-2 veces por cada uso de la aplicación.

Tabla 3.5.1: CU–01 — Elegir cifrado

ID y Nombre del caso de uso:	**CU–2 — Elegir alfabeto**
Dependencias:	Depende del requisito RF–3: Elegir alfabeto.
Disparador:	Seleccionar elemento de una lista.
Descripción:	La aplicación muestra una lista con los alfabetos previamente cargados en el sistema para que el usuario seleccione el que pretende usar.
Precondiciones:	PRE–1 — Se debe haber seleccionado previamente un tipo de cifrado que utilice alfabetos para poder seleccionar y cambiar el alfabeto.
Postcondiciones:	POST–1 — Se comprueba la clave introducida actualmente, para que se adecue a las necesidades del alfabeto actual.
Flujo normal:	**2.0 Se muestra la lista de alfabetos disponibles** 1. El usuario elige uno de los alfabetos disponibles. 2. La aplicación lee el archivo del sistema que contiene el alfabeto elegido y comprueba que el archivo exista y tenga un formato correcto. 3. La aplicación comprueba que la clave está compuesta solo por caracteres alfanuméricos existentes en el alfabeto elegido, y borra el resto de caracteres de la clave.
Flujos alternativos:	
Excepciones:	**2.0.E1 El alfabeto elegido no existe o tiene un formato incorrecto** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación. 3. La aplicación elimina el archivo (si existía) 4. La aplicación recarga la lista de alfabetos y repite la operación con el primero de la lista.
Prioridad:	Media
Frecuencia de uso:	1-2 veces por cada uso de la aplicación.

Tabla 3.5.2: CU–02 — Elegir alfabeto

ID y Nombre del caso de uso:	**CU–3 — Elegir codificación de texto**
Dependencias:	Depende del requisito RF–6: Elegir codificación de texto.
Disparador:	Seleccionar elemento de una lista.
Descripción:	La aplicación muestra una lista con los tipos de codificación de texto soportados por java en el sistema para que el usuario seleccione el que pretende usar.
Precondiciones:	
Postcondiciones:	POST–1 — Se modifica el texto no cifrado de la aplicación, de modo que la codificación de ese texto pasa a ser la nueva codificación elegida.
Flujo normal:	**3.0 Se muestra la lista de tipos de codificación disponibles** 1. El usuario elige uno de los tipos de codificación disponibles.
Flujos alternativos:	
Excepciones:	**3.0.E1 La codificación de texto no es soportada** 1. La aplicación cancela la operación. 2. La aplicación repite la operación, cambiando la codificación automáticamente por "UTF–8".
Prioridad:	Media
Frecuencia de uso:	1-2 veces por cada uso de la aplicación.

Tabla 3.5.3: CU–03 — Elegir codificación de texto

ID y Nombre del caso de uso:	**CU–4 — Elegir idioma**
Dependencias:	Depende del requisito RF–28: Elegir idioma.
Disparador:	Seleccionar elemento de una lista.
Descripción:	La aplicación muestra una lista con los idiomas existentes en el sistema para que el usuario seleccione el que desea usar.
Precondiciones:	
Postcondiciones:	POST–1 — Se modifican los elementos mostrados en la interfaz para cambiar su texto al idioma seleccionado.
Flujo normal:	**4.0 Se muestra la lista de idiomas disponibles** 1. El usuario elige uno de los idiomas disponibles. 2. La aplicación modifica el texto de cada elemento de la interfaz.
Flujos alternativos:	
Excepciones:	**4.0.E1 No se encuentra el archivo interno con los textos del idioma correspondiente** 1. La aplicación muestra un mensaje interno. 2. La aplicación cancela la operación. 3. La aplicación se cierra.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.4: CU–04 — Elegir idioma

ID y Nombre del caso de uso:	CU–5 — Cargar texto
Dependencias:	Depende del requisito RF–9: Cargar texto.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un archivo de tipo ".txt" de su PC, y el contenido se carga en el cuadro de texto de la aplicación.
Precondiciones:	PRE–1 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se muestra en un cuadro de texto editable el contenido del archivo elegido.
Flujo normal:	**5.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación detecta automáticamente el tipo de codificación del texto que contiene el archivo. 3. Se modifica la codificación de texto actual a la detectada por el archivo. 4. Se carga el texto contenido en el archivo en un cuadro de texto editable.
Flujos alternativos:	**5.1 No se puede detectar la codificación archivo, o no está soportada por la aplicación** 1. La aplicación trata la operación como si la codificación detectada hubiera sido "UTF-8".
Excepciones:	**5.0.E1 El archivo elegido no existe.** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación.
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.5: CU–05 — Cargar texto

ID y Nombre del caso de uso:	CU–6 — Guardar texto
Dependencias:	Depende del requisito RF–9: Guardar texto.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un nuevo archivo de su PC para guardar el texto contenido en el cuadro de texto correspondiente de la aplicación.
Precondiciones:	
Postcondiciones:	POST–1 — Se crea un archivo con el contenido del cuadro de texto correspondiente con codificación elegida.
Flujo normal:	**6.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un nuevo archivo de su PC. 2. Se guarda el contenido del cuadro de texto correspondiente en el archivo elegido.
Flujos alternativos:	**6.1 El archivo elegido ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o elegir uno nuevo.
Excepciones:	**6.0.E1 La codificación elegida no está soportada por la aplicación** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación. 3. La aplicación reestablece la codificación "UTF-8" en el texto introducido en la aplicación.
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.6: CU–06 — Guardar texto

ID y Nombre del caso de uso:	CU–7 — Cargar texto cifrado
Dependencias:	Depende del requisito RF–11: Cargar texto cifrado.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un archivo de su PC con la extensión adecuada en función del método de cifrado empleado, y el contenido se carga en el cuadro de texto cifrado de la aplicación.
Precondiciones:	PRE–1 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se muestra en un cuadro de texto el contenido del archivo elegido.
Flujo normal:	**7.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación lee el contenido del archivo como un texto con codificación “UTF-8”. 3. Se carga el contenido en un cuadro de texto.
Flujos alternativos:	
Excepciones:	**7.0.E1 El archivo elegido no existe.** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación.
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.7: CU–07 — Cargar texto cifrado

ID y Nombre del caso de uso:	CU–8 — Guardar texto cifrado
Dependencias:	Depende del requisito RF–12: Guardar texto cifrado.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un nuevo archivo de su PC para guardar el texto contenido en el cuadro de texto correspondiente de la aplicación.
Precondiciones:	
Postcondiciones:	POST–1 — Se crea un archivo con el contenido del cuadro de texto correspondiente con codificación “UTF-8”.
Flujo normal:	**8.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un nuevo archivo de su PC. 2. Se guarda el contenido del cuadro de texto correspondiente en el archivo elegido.
Flujos alternativos:	**8.1 El archivo elegido ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o elegir uno nuevo.
Excepciones:	
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.8: CU–08 — Guardar texto cifrado

ID y Nombre del caso de uso:	CU–9 — Cifrar texto
Dependencias:	Depende del requisito RF–15: Cifrar texto.
Disparador:	Pulsar un botón.
Descripción:	La aplicación cifra el texto en función del método de cifrado y la clave elegidos y lo muestra en el cuadro de texto cifrado de la aplicación.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe haberse introducido una clave válida. PRE–3 — Debe proporcionarse el texto a cifrar.
Postcondiciones:	POST–1 — Se mostrará en un cuadro de texto el resultado de la operación.
Flujo normal:	**9.0 El usuario pulsa el botón que inicia la operación** 1. La aplicación realiza la operación de cifrado de forma inadvertida al usuario. 2. La aplicación carga en el cuadro de texto cifrado el resultado de la operación.
Flujos alternativos:	
Excepciones:	**9.0.E1 No se ha proporcionado texto o una clave válida** 1. La aplicación cancela la operación.
Prioridad:	Alta
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.9: CU–09 — Cifrar texto

ID y Nombre del caso de uso:	CU–10 — Descifrar texto
Dependencias:	Depende del requisito RF–16: Guardar texto.
Disparador:	Pulsar un botón.
Descripción:	La aplicación descifra el texto en función del método de cifrado y la clave elegidos y lo muestra en el cuadro de texto de la aplicación.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe haberse introducido una clave válida. PRE–3 — Debe proporcionarse el texto a descifrar.
Postcondiciones:	POST–1 — Se mostrará en un cuadro de texto el contenido del archivo elegido.
Flujo normal:	**10.0 El usuario pulsa el botón que inicia la operación** 1. La aplicación realiza la operación de descifrado de forma inadvertida al usuario. 2. La aplicación carga en el cuadro de texto el resultado de la operación.
Flujos alternativos:	
Excepciones:	**10.0.E1 No se ha proporcionado texto o una clave válida** 1. La aplicación cancela la operación.
Prioridad:	Alta
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.10: CU–10 — Descifrar texto

ID y Nombre del caso de uso:	**CU–11 — Cifrar archivo**
Dependencias:	Depende del requisito RF–17: Cifrar archivo.
Disparador:	Pulsar un botón.
Descripción:	El usuario elige un archivo de su PC, y la aplicación generará un nuevo archivo como resultado de cifrar su contenido con el método de cifrado y la clave elegidos.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe haberse introducido una clave válida. PRE–3 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se creará un nuevo archivo con la extensión adecuada en función del método de cifrado elegido
Flujo normal:	**11.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación realiza la operación de cifrado de forma inadvertida al usuario. 3. La aplicación guarda en un nuevo archivo el resultado de la operación.
Flujos alternativos:	**11.1 El archivo generado por la aplicación ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o cancelar la operación.
	11.2 Se utiliza un método de cifrado clásico (solo cifra texto) 1. La aplicación detecta automáticamente la codificación del texto contenido por el archivo. 2. Se lee en contenido del archivo como un texto con la codificación detectada. 3. El texto resultante de la operación de cifrado se guarda con la codificación “UTF-8”.
Excepciones:	**11.0.E1 El archivo elegido no existe, o la clave no es válida** 1. La aplicación cancela la operación.
Prioridad:	Alta
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.11: CU–11 — Cifrar archivo

ID y Nombre del caso de uso:	CU–12 — Descifrar archivo
Dependencias:	Depende del requisito RF–18: Descifrar archivo.
Disparador:	Pulsar un botón.
Descripción:	El usuario elige un archivo de su PC, y la aplicación generará un nuevo archivo como resultado de descifrar su contenido con el método de cifrado y la clave elegidos.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe haberse introducido una clave válida. PRE–3 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se creará un nuevo archivo con el nombre y extensión del archivo original (previo al cifrado).
Flujo normal:	**12.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación realiza la operación de descifrado de forma inadvertida al usuario. 3. La aplicación guarda en un nuevo archivo el resultado de la operación.
Flujos alternativos:	**12.1 El archivo generado por la aplicación ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o cancelar la operación.
	12.2 Se utiliza un método de cifrado clásico (solo cifra texto) 1. El texto resultante de la operación de descifrado se guarda con la codificación "UTF-8".
Excepciones:	**12.0.E1 El archivo elegido no existe, o la clave no es válida** 1. La aplicación cancela la operación.
Prioridad:	Alta
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.12: CU–12 — Descifrar archivo

ID y Nombre del caso de uso:	CU–13 — Criptoanalizar texto
Dependencias:	Depende del requisito RF–19: Criptoanalizar texto.
Disparador:	Pulsar un botón.
Descripción:	La aplicación examina el texto cifrado de la aplicación, genera un archivo resumen con los resultados de las operaciones de análisis y lo muestra en pantalla.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — Debe proporcionarse el texto a analizar.
Postcondiciones:	POST–1 — Se creará un nuevo archivo con información sobre el resultado de las operaciones realizadas. POST–2 — Se mostrará el archivo creado por pantalla.
Flujo normal:	**13.0 El usuario pulsa el botón que inicia la operación** 1. La aplicación realiza la operación de criptoanálisis del texto de forma inadvertida al usuario. 2. La aplicación crea el archivo resultante con la información recopilada. 3. La aplicación muestra el archivo creado por pantalla.
Flujos alternativos:	
Excepciones:	**13.0.E1 No se ha proporcionado texto.** 1. La aplicación cancela la operación.
Prioridad:	Baja
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.13: CU–13 — Criptoanalizar texto

ID y Nombre del caso de uso:	CU–14 — Criptoanalizar archivo
Dependencias:	Depende del requisito RF–20: Criptoanalizar archivo.
Disparador:	Pulsar un botón.
Descripción:	El usuario elige un archivo de su PC, la aplicación generará archivo resumen como resultado de analizar su contenido, y mostrará ese archivo por pantalla.
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se creará un nuevo archivo con información sobre el resultado de las operaciones realizadas. POST–2 — Se mostrará el archivo creado por pantalla.
Flujo normal:	**14.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación realiza la operación de análisis de forma inadvertida al usuario. 3. La aplicación crea el archivo resultante con la información recopilada. 4. La aplicación muestra el archivo creado por pantalla.
Flujos alternativos:	
Excepciones:	**14.0.E1 El archivo elegido no existe** 1. La aplicación cancela la operación.
Prioridad:	Baja
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.14: CU–14 — Criptoanalizar archivo

ID y Nombre del caso de uso:	CU–15 — Consultar información
Dependencias:	Depende del requisito RF–7: Consultar información.
Disparador:	Pulsar un botón.
Descripción:	La aplicación abre un archivo ".pdf" predefinido que explica brevemente el funcionamiento del tipo de cifrado correspondiente
Precondiciones:	PRE–1 — Debe haberse elegido un método de cifrado. PRE–2 — El archivo predefinido debe existir.
Postcondiciones:	POST–1 — Se mostrará por pantalla el archivo predefinido.
Flujo normal:	**15.0 El usuario pulsa el botón que inicia la operación** 1. La aplicación muestra el archivo predefinido por pantalla.
Flujos alternativos:	
Excepciones:	**15.0.E1 El archivo elegido no existe** 1. La aplicación muestra el mensaje de error correspondiente. 1. La aplicación cancela la operación.
Prioridad:	Baja
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.15: CU–15 — Consultar Información

ID y Nombre del caso de uso:	CU–16 — Cargar cifrado hexadecimal
Dependencias:	Depende del requisito RF–13: Cargar cifrado hexadecimal.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un archivo de su PC con la extensión adecuada en función del método de cifrado empleado, y el contenido se carga como texto hexadecimal en el cuadro de texto cifrado de la aplicación.
Precondiciones:	PRE–1 — El archivo elegido debe existir.
Postcondiciones:	POST–1 — Se muestra en un cuadro de texto el contenido del archivo elegido.
Flujo normal:	**16.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un archivo de su PC. 2. La aplicación lee el contenido del archivo como una cadena de bytes. 3. Se carga el contenido como texto hexadecimal en un cuadro de texto.
Flujos alternativos:	
Excepciones:	**16.0.E1 El archivo elegido no existe.** 1. La aplicación muestra un mensaje con el error correspondiente. 2. La aplicación cancela la operación.
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.16: CU–16 — Cargar cifrado hexadecimal

ID y Nombre del caso de uso:	CU–17 — Guardar cifrado hexadecimal
Dependencias:	Depende del requisito RF–14: Guardar cifrado hexadecimal.
Disparador:	Pulsar un botón.
Descripción:	El usuario selecciona un nuevo archivo de su PC para guardar el texto contenido en el cuadro de texto correspondiente de la aplicación.
Precondiciones:	
Postcondiciones:	POST–1 — Se crea un archivo con una cadena de bytes como el contenido del cuadro de texto correspondiente .
Flujo normal:	**17.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige un nuevo archivo de su PC. 2. Se transforma el contenido del cuadro de texto correspondiente en la cadena de bytes que representa la secuencia de texto hexadecimal. 3. Se guarda en el archivo elegido la cadena de bytes resultante.
Flujos alternativos:	**17.1 El archivo elegido ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar el archivo o elegir uno nuevo.
Excepciones:	
Prioridad:	Media
Frecuencia de uso:	1-5 veces por cada uso de la aplicación.

Tabla 3.5.17: CU–17 — Guardar cifrado hexadecimal

ID y Nombre del caso de uso:	CU–18 — Generar clave segura
Dependencias:	Depende del requisito RF–22: Generar clave segura.
Disparador:	Pulsar un botón.
Descripción:	La aplicación genera automáticamente una clave segura aleatoria, válida para el tipo de cifrado elegido, y la guarda en el sistema como un archivo del formato adecuado.
Precondiciones:	
Postcondiciones:	POST–1 — Se creará un nuevo archivo con la nueva clave segura aleatoria. POST–2 — Se recargarán las claves de la lista.
Flujo normal:	**18.0 Se muestra una ventana de tipo formulario** 1. El usuario elige un nombre para su clave. 2. La aplicación genera el nuevo archivo clave. 3. La aplicación recarga la lista de claves.
Flujos alternativos:	**18.1 La clave elegida ya existe** 1. La aplicación pedirá una confirmación al usuario para borrar la clave o cancelar la operación.
Excepciones:	
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.18: CU–18 — Generar clave segura

ID y Nombre del caso de uso:	CU–19 — Eliminar clave
Dependencias:	Depende del requisito RF–23: Eliminar clave.
Disparador:	Pulsar un botón.
Descripción:	La aplicación muestra una ventana de confirmación para que el usuario verifique que realmente quiere eliminar la clave seleccionada.
Precondiciones:	PRE–1 — La clave elegida debe existir
Postcondiciones:	POST–1 — Se eliminará el archivo que guarda la clave elegida. POST–2 — Se recargarán las claves de la lista.
Flujo normal:	**19.0 Se muestra una ventana de confirmación** 1. El usuario elige la opción “Sí” o “No” en la ventana de confirmación
Flujos alternativos:	**19.1 Se confirma la eliminación de la clave** 1. La aplicación elimina el archivo correspondiente a la clave elegida. 2. La aplicación recarga la lista de claves.
	19.2 Se cancela la eliminación de la clave 1. La aplicación anula la operación.
Excepciones:	**19.0.E1 La clave elegida no existe.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.19: CU–19 — Eliminar clave

ID y Nombre del caso de uso:	CU–20 — Exportar clave
Dependencias:	Depende del requisito RF–25: Exportar clave.
Disparador:	Pulsar un botón.
Descripción:	El usuario elegirá la carpeta en la que extraer el la clave que desea exportar, y la aplicación copiará la clave en esa carpeta como un nuevo archivo.
Precondiciones:	PRE–1 — La clave elegida debe existir. PRE–2 — La carpeta elegida debe existir.
Postcondiciones:	POST–1 — Se copiará el archivo que guarda la clave elegida a la carpeta elegida.
Flujo normal:	**20.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige la carpeta en la que exportar la clave. 2. La aplicación realiza una copia del archivo que guarda la clave seleccionada, a la carpeta destino elegida por el usuario.
Flujos alternativos:	
Excepciones:	**20.0.E1 La clave o carpeta elegidas no existen.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.20: CU–20 — Exportar clave

ID y Nombre del caso de uso:	CU–21 — Importar clave
Dependencias:	Depende del requisito RF–24: Importar clave.
Disparador:	Pulsar un botón.
Descripción:	El usuario elegirá la el archivo clave que desea importar, y la aplicación copiará esa clave en el sistema.
Precondiciones:	PRE–1 — El archivo elegido debe existir PRE–2 — El archivo elegido debe tener un formato compatible.
Postcondiciones:	POST–1 — Se copiará el archivo clave elegido al la carpeta de claves predeterminada del sistema. POST–2 — La aplicación recarga la lista de claves
Flujo normal:	**21.0 Se muestra una ventana de navegación por carpetas** 1. El usuario elige el archivo clave que desea importar. 2. La aplicación realiza una copia del archivo que guarda la clave seleccionada, a la carpeta de claves predeterminada del sistema.
Flujos alternativos:	
Excepciones:	**21.0.E1 El archivo no existe, o tiene un formato no compatible.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.21: CU–21 — Importar clave

ID y Nombre del caso de uso:	CU–22 — Añadir Firma Digital
Dependencias:	Depende del requisito RF–26: Añadir Firma Digital.
Disparadores:	Activar “Cifrar Archivo” o “Cifrar Texto”.
Descripción:	El usuario determinará si pretende añadir una firma digital al archivo o mensaje, y en caso afirmativo, elegir la clave que usará como firma.
Precondiciones:	PRE–1 — La clave elegida como firma debe existir
Postcondiciones:	POST–1 — Se añadirá al inicio del archivo o texto resultante un conjunto de datos que se corresponden con la firma digital del archivo.
Flujo normal:	**22.0 Se muestra una ventana de confirmación** 1. El usuario elige si desea añadir la firma digital. 2. En caso afirmativo el usuario vuelve elige una clave adicional en una nueva ventana, que será la usada para realizar la firma digital 3. El proceso de encriptado comienza con la adición de la firma digital al inicio del archivo o texto cifrado. 4. El proceso de encriptado continúa normalmente.
Flujos alternativos:	**22.1 Se cancela la operación de firma digital** 1. El proceso de encriptado continúa normalmente sin la firma digital.
Excepciones:	**22.0.E1 La clave elegida no existe.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.22: CU–22 — Añadir Firma Digital

ID y Nombre del caso de uso:	CU–23 — Verificar Firma Digital
Dependencias:	Depende del requisito RF–27: Verificar Firma Digital.
Disparadores:	Activar “Descifrar Archivo” o “Descifrar Texto”.
Descripción:	El usuario determinará si pretende comprobar la existencia y veracidad de la firma digital, y por ende el mensaje completo, en un archivo o texto.
Precondiciones:	PRE–1 — La clave elegida como firma debe existir
Postcondiciones:	POST–1 — La creación del archivo o texto descifrado dependerá de la veracidad de la firma digital.
Flujo normal:	**23.0 Se muestra una ventana de confirmación** 1. El usuario elige si desea verificar la firma digital. 2. En caso afirmativo el usuario vuelve elige una clave adicional en una nueva ventana, que será la usada para comprobar la firma digital. 3. El proceso de desencriptado comienza aislando los datos que corresponden a la firma digital y comprobando si esta es correcta. 4. El proceso de desencriptado continúa normalmente.
Flujos alternativos:	**23.1 Se cancela la operación de firma digital** 1. El proceso de desencriptado continúa normalmente sin la firma digital.
Excepciones:	**23.0.E1 La clave elegida no existe.** 1. La aplicación cancela la operación. 2. La aplicación recarga la lista de claves.
Prioridad:	Alta
Frecuencia de uso:	1 vez por cada uso de la aplicación.

Tabla 3.5.23: CU–23 — Verificar Firma Digital

Anexo B. Requisitos Funcionales

	RF–1 — Elegir cifrado
Versión	2.0 (20/01/2016)
Dependencias:	RF–29 Cargar métodos de cifrado
Descripción:	El sistema permitirá al usuario elegir uno de los cifrados mostrados en pantalla para realizar las actividades de encriptado/desencriptado.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.1: RF–01 — Elegir cifrado

	RF–2 — Cargar alfabetos
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema cargará automáticamente sus alfabetos predeterminados, además de cualquier otro alfabeto que haya podido añadir el usuario.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.2: RF–02 — Cargar alfabetos

	RF–3 — Elegir alfabeto
Versión	1.0 (21/09/2015)
Dependencias:	RF–2 Cargar Alfabetos
Descripción:	El sistema permitirá al usuario elegir uno de los alfabetos mostrados en pantalla, siempre y cuando esté utilizando un método de cifrado que dependa de un alfabeto.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.3: RF–03 — Elegir alfabeto

	RF–4 — Cargar tipos de codificación de texto
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema cargará automáticamente la lista de diferentes codificaciones de texto soportada por Java.
Prioridad:	Media
Comentarios:	Ninguno

Tabla B.4: RF–04 — Cargar tipos de codificación de texto

	RF–5 — Cargar codificación de un archivo de texto
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema cargará automáticamente la codificación de texto detectada al introducir el contenido de un archivo de texto.
Prioridad:	Baja
Comentarios:	Ninguno

Tabla B.5: RF–05 — Elegir codificación de texto

	RF–6 — Elegir codificación de texto
Versión	1.0 (21/09/2015)
Dependencias:	RF–4 Cargar tipos de codificación de texto
Descripción:	El sistema permitirá al usuario elegir una de las codificaciones de texto mostradas en pantalla, modificando así el texto no cifrado introducido en el programa.
Prioridad:	Media
Comentarios:	Ninguno

Tabla B.6: RF–06 — Elegir codificación de texto

	RF–7 — Consultar información
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema cargará un archivo .pdf temporal con una explicación sencilla del funcionamiento del método de cifrado actual.
Prioridad:	Baja
Comentarios:	Ninguno

Tabla B.7: RF–07 — Consultar información

	RF–8 — Introducir clave
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario introducir una clave alfanumérica a su elección.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.8: RF–08 — Introducir clave

	RF–9 — Cargar texto
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario elegir un archivo .txt para cargar su contenido como texto no cifrado.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.9: RF–09 — Cargar texto

	RF–10 — Guardar texto
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario guardar el texto no cifrado introducido en la aplicación como un archivo .txt codificado de forma predeterminada en formato UTF-8.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.10: RF–10 — Guardar texto

	RF–11 — Cargar texto cifrado
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario elegir un archivo cifrado generado mediante el programa para cargar su contenido como texto cifrado.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.11: RF–11 — Cargar texto cifrado

	RF–12 — Guardar texto cifrado
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario guardar el texto cifrado generado por la aplicación como un archivo cifrado cuyo contenido es un texto codificado de forma predeterminada en formato UTF-8.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.12: RF–12 — Guardar texto cifrado

	RF–13 — Cargar cifrado hexadecimal
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario elegir un archivo cifrado generado mediante el programa para cargar su contenido en bytes como un texto hexadecimal.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.13: RF–13 — Cargar cifrado hexadecimal

	RF–14 — Guardar cifrado hexadecimal
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario guardar el texto cifrado hexadecimal generado por la aplicación como un archivo cifrado cuyo contenido es la cadena de bytes que representa ese texto hexadecimal.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.12: RF–12 — Guardar cifrado hexadecimal

	RF–15 — Cifrar texto
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario cifrar el texto no cifrado introducido en la aplicación según el método de cifrado previamente elegido.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.15: RF–15 — Cifrar texto

	RF–16 — Descifrar texto
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario descifrar el texto cifrado generado por el programa
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.16: RF–16 — Descifrar texto

	RF–17 — Cifrar archivo
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario cifrar un archivo del dispositivo según el método de cifrado previamente elegido.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.17: RF–17 — Cifrar archivo

	RF–18 — Descifrar archivo
Versión	1.0 (21/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario descifrar un archivo del dispositivo previamente cifrado según el método de cifrado previamente elegido.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.16: RF–16 — Descifrar archivo

	RF–19 — Criptoanalizar texto
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario analizar el texto cifrado cargado en el programa, generando un archivo con las claves más probables que pudieran ser usadas para cifrarlo.
Prioridad:	Media
Comentarios:	Ninguno

Tabla B.19: RF–19 — Criptoanalizar exto

	RF–20 — Criptoanalizar archivo
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario analizar un archivo cifrado del dispositivo, generando un archivo con las claves más probables que pudieran ser usadas para cifrarlo.
Prioridad:	Media
Comentarios:	Ninguno

Tabla B.20: RF–20 — Criptoanalizar archivo

	RF–21 — Cargar claves
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema cargará automáticamente la lista de claves seguras generadas o importadas por el programa en los métodos de cifrado correspondientes.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.21: RF–21 — Cargar claves

	RF–22 — Generar clave segura
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario generar una nueva clave segura en los métodos de cifrado que no admitan una clave alfanumérica.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.22: RF–22 — Generar clave segura

	RF–23 — Eliminar clave
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario eliminar una de las claves seguras previamente generadas por el programa.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.23: RF–23 — Eliminar clave

	RF–24 — Importar clave
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario importar una nueva clave segura a partir de un archivo compatible guardado en el dispositivo.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.24: RF–24 — Importar clave

	RF–25 — Exportar clave
Versión	1.0 (22/09/2015)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario exportar a un archivo compatible una clave segura generada previamente por el programa.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.25: RF–25 — Exportar clave

	RF–26 — Añadir firma digital
Versión	1.0 (23/01/2016)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario elegir si desea añadir la firma digital a un archivo o mensaje cifrado.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.26: RF–26 — Añadir firma digital

	RF–27 — Verificar firma digital
Versión	1.0 (23/01/2016)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario elegir si desea verificar la existencia y autenticidad de la firma digital de un archivo o mensaje cifrado.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.27: RF–27 — Verificar firma digital

	RF–28 — Elegir idioma
Versión	1.0 (25/01/2016)
Dependencias:	Ninguna
Descripción:	El sistema permitirá al usuario elegir entre diferentes idiomas para mostrar la interfaz del programa.
Prioridad:	Media
Comentarios:	Ninguno

Tabla B.28: RF–28 — Elegir idioma

	RF–29 — Cargar métodos de cifrado
Versión	1.0 (20/01/2016)
Dependencias:	Ninguna
Descripción:	El sistema cargará automáticamente los métodos de cifrado compatibles almacenados en el directorio reservado.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.29: RF–29 — Cargar métodos de cifrado

	RF–30 — Cargar componentes portables
Versión	1.0 (27/01/2016)
Dependencias:	Ninguna
Descripción:	El sistema cargará automáticamente los métodos de cifrado compatibles almacenados en el directorio reservado.
Prioridad:	Alta
Comentarios:	Ninguno

Tabla B.30: RF–30 — Cargar componentes portables