



Universidad de Valladolid

Escuela de Ingeniería Informática

TRABAJO FIN DE GRADO

Grado en Ingeniería Informática, mención en Ingeniería de Software

**ASISTEV: Asistente Android para traducción
de textos escritos en viajes**

Autor:

David González Verdugo

Tutor:

Miguel Ángel Laguna Serrano

Resumen

Actualmente, el auge de los dispositivos móviles inteligentes es innegable y su uso nos ha facilitado la vida en muchos aspectos. Antes del nacimiento de este tipo de dispositivos, invertíamos una gran parte de nuestro tiempo en hacer todo tipo de cosas que actualmente podemos hacer en segundos, haciendo uso de las aplicaciones de nuestros *smartphones* o *tablet*.

Estos dispositivos móviles nos han facilitado, entre otras cosas, los transcurso de nuestros viajes, ya que nos hemos olvidado de los mapas que tanto nos costaba doblar, de los diccionarios o traductores que no encontraban espacio en nuestras maletas o bolsas de mano, de las guías del lugar que visitábamos e incluso de las brújulas, para los más aventureros. Hoy en día podemos acceder a todo esto y más con un único dispositivo que prácticamente nos cabe en la palma de la mano, en el caso de los *smartphones*.

Siguiendo esta línea, este proyecto pretende facilitar la tarea de traducción de todo tipo de textos en el transcurso de un viaje, haciendo uso de la cámara de un *smartphone* o *tablet* Android y de técnicas de reconocimiento óptico de caracteres u OCR. Los textos a traducir pueden ser desde señales de tráfico y carteles hasta menús de restaurantes, periódicos, revistas u otros.

La aplicación desarrollada, *Asistev*, no es un traductor convencional. Además de reconocer caracteres en más de 30 idiomas y poder traducirlos al mismo número de idiomas, permite guardar las traducciones realizadas para poder consultarlas posteriormente. También ofrece la posibilidad de filtrar traducciones por fecha y/o localización o buscarlas por texto traducido, fecha o localización. Esto es muy útil, por ejemplo, si un viajero hace una ruta por ciudades asiáticas durante un mes y desea consultar una traducción de un cartel que realizó en una fecha y localización concreta.

En cuanto a la arquitectura, se ha utilizado el patrón MVP o Modelo-Vista-Presentador, que mejora la legibilidad y modularidad del código, evitando utilizar una única clase o Activity que contenga todo el código de cada vista.

En cuanto al diseño de la interfaz, se han seguido las directivas *Material Design* de Google, lo que convierte a la aplicación en muy intuitiva y sencilla de utilizar, a la par que agradable a la vista.

En cuanto a modelos de proceso y metodologías, el proyecto se ha llevado a cabo basándose en el modelo de desarrollo ágil *Scrum*.

En esta memoria se encuentra todo la documentación necesaria para comprender el proyecto desarrollado, desde los requisitos, análisis, diseño, implementación y pruebas, hasta el manual de usuario y otros aspectos relevantes.

Abstract

Nowadays, the rise of smart mobile devices is undeniable and its use has made our life easier in many ways. Before the birth of these devices, we have invested a lot of our time doing all kind of things that we can do currently in seconds using *smartphones* or *tablets* apps.

These mobile devices have made us easier, among other things, the course of our journeys, since we have forgotten the maps that we hardly bent, dictionaries or translators which we could not find space in suitcases and handbags to store them, guides and even compasses, for the more adventurous. Today we can access all this and more with a single device that we practically fits in the palm of the hand, in the case of *smartphones*.

Following this line, this project aims to facilitate the task of translating all kinds of texts in the course of a trip, using an Android *smartphone* or *tablet* camera and optical character recognition techniques or OCR. The texts to be translated may be from traffic signs and signboards to restaurant menus, newspapers, magazines or other.

The developed app, *Asistev*, is not a conventional translator. In addition to recognizing characters in more than 30 languages and translate them to the same number of languages, translations can be saved for future queries. It also provides filtering translations by date and/or location or searching by translated text, date or location. This is useful, for example, if a traveler makes an Asian tour for a month and want to see a signboard translation which took place on a specific date and location.

In terms of architecture, it has been used MVP or Model-View-Presenter pattern, which improves readability and code modularity, avoiding the use of a single class or activity which contains all the code.

As for the design of the interface, have been followed the *Google Material Design* directives, which makes app very intuitive, easy to use and sightly.

As process models and methodology, an agile development model was used, in this case *Scrum*.

In this work can be found all necessary documentation to understand the developed project, from requirements, analysis, design, implementation and testing to user manual and other relevant aspects.

Tabla de contenidos

1. Introducción	1
1.1. Contexto y motivación	1
1.2. Objetivos	1
1.3. Metodología utilizada	2
1.4. Resumen del contenido de la memoria	2
2. Entorno tecnológico	5
2.1. Herramientas utilizadas	5
2.2. Entorno de desarrollo	5
3. Plan de Desarrollo de Software	7
3.1. Introducción	7
3.1.1. Propósito	7
3.1.2. Alcance	7
3.1.3. Resumen del plan	7
3.2. Visión general del proyecto	8
3.2.1. Objetivos	8
3.2.2. Suposiciones y restricciones	8
3.2.3. Características del proyecto	8
3.2.4. Elección del modelo de proceso y metodología para desarrollo de software	9
3.2.5. Elección del tipo de desarrollo y de bibliotecas OCR y traducción	11
3.2.6. Entregables del proyecto	14
3.3. Organización del proyecto	14
3.3.1. Interfaces externas	14
3.3.2. Estructura interna	15
3.3.3. Roles y responsabilidades	15
3.3.4. Eventos o métodos operativos del equipo	16
3.4. Planificación del proyecto	17
3.4.1. Estimaciones del proyecto	17
3.4.2. Desviaciones del proyecto	18
3.4.3. Entregas e hitos	19
3.4.4. Datos interesantes	19
3.5. Seguimiento del proyecto	20
4. Plan de Gestión de Riesgos	21
4.1. Introducción	21
4.1.1. Propósito	21
4.1.2. Alcance	21
4.2. Gestión del riesgo	21
4.2.1. Identificación, análisis, planes de acción y seguimiento	22
4.2.2. Control de riesgos	34
5. Requisitos	35
5.1. Los requisitos en desarrollo ágil	35
5.2. Requisitos funcionales	36
5.3. Requisitos no funcionales	37
5.3.1. Requisitos de información	37

6. Análisis	39
6.1. Modelo de Casos de Uso.	39
6.1.1. Actores	39
6.1.2. Diagrama de casos de uso.	39
6.1.3. Especificación de Casos de Uso.	40
6.2. Realización de los Casos de Uso en Análisis	43
6.2.1. Modelo de Dominio	43
6.2.2. Descripción de las clases del modelo de dominio	44
6.2.3. Diagramas de actividad	45
7. Diseño	51
7.1. Diseño de la arquitectura	51
7.1.1. Patrón arquitectónico MVP	51
7.1.2. Patrones de diseño utilizados	52
7.1.3. Arquitectura general	53
7.1.4. Descomposición modular	54
7.1.5. Diseño de los módulos o subsistemas	55
7.1.6. Relación entre módulos	59
7.2. Modelo de Datos	63
7.2.1. Esquema relacional de la base de datos	63
7.2.2. Consideraciones de las tablas del esquema relacional	63
7.3. Realización de los Casos de Uso en Diseño	64
7.3.1. CU01: Añadir traducción. Parte 1	64
7.3.2. CU01: Añadir traducción. Parte 2	65
7.3.3. Detalle de <i>Obtener todos los idiomas</i>	66
7.3.4. Detalle de <i>Guardar traducción</i>	67
7.3.5. CU02: Visualizar traducciones	68
7.3.6. Detalle de <i>Obtener todas las traducciones</i>	69
7.3.7. Detalle de <i>Obtener los filtros seleccionados</i>	70
7.3.8. Detalle de <i>Actualizar y animar la lista de traducciones</i>	71
7.3.9. CU03: Consultar detalle de traducción	72
7.3.10. Detalle de <i>Obtener nombre del idioma a partir de su código</i>	73
7.3.11. CU04: Borrar traducción	74
7.3.12. CU05: Buscar traducción	75
7.3.13. CU06: Filtrar traducciones	76
7.4. Diseño de la interfaz de usuario	77
7.4.1. CU01. Añadir traducción	77
7.4.2. CU02. Visualizar traducciones	79
7.4.3. CU03. Consultar detalle de traducción	80
7.4.4. CU04. Borrar traducción	81
7.4.5. CU05. Buscar traducción	82
7.4.6. CU06. Filtrar traducciones	83
8. Implementación y pruebas	85
8.1. Implementación	85
8.2. Plan de pruebas	85
8.2.1. Introducción	85
8.2.2. Tipos de pruebas realizadas	86
8.2.3. Casos de Pruebas	86
9. Conclusiones	95
9.1. Objetivos alcanzados	95

9.2. Valoración personal	96
9.3. Líneas futuras	96
Bibliografía	97
Anexos	99
A. Glosario de términos	99
B. Manual de usuario	101
B.1. Introducción	101
B.2. Funcionalidades	101
B.2.1. Añadir traducción	101
B.2.2. Visualizar traducciones	106
B.2.3. Consultar detalle de traducción	107
B.2.4. Borrar traducción	108
B.2.5. Buscar traducción	109
B.2.6. Filtrar traducciones	110
C. Plan de seguimiento	111
C.1. Introducción	111
C.1.1. Propósito	112
C.1.2. Alcance	113
C.1.3. Códigos de colores	113
C.2. Sprint 0: 1 de febrero – 21 de febrero 2016	114
C.2.1. Pila del producto	114
C.2.2. US01 - Como product owner quiero tener disponible la documentación inicial del proyecto para poder tener una visión global del proyecto	114
C.2.3. Revisión de Sprint	115
C.3. Sprint 1: 22 de febrero – 10 de marzo 2016	116
C.3.1. Pila del producto	116
C.3.2. US14 - Como Product Owner quiero tener un modelo de dominio de la aplicación y un plan de gestión de riesgos completo, que garantice un control de los riesgos y las medidas de corrección oportunas	116
C.3.3. US06 - Como Usuario quiero que se reconozcan los caracteres contenidos en una imagen	117
C.3.4. Revisión de Sprint	118
C.4. Sprint 2: 11 de marzo – 31 de marzo 2016	119
C.4.1. Pila del producto	119
C.4.2. US07 - Como Usuario quiero realizar una foto con la cámara del dispositivo, que posteriormente será utilizada en el reconocimiento de caracteres	119
C.4.3. US08 - Como usuario quiero seleccionar una imagen de la galería, que posteriormente será utilizada en el reconocimiento de caracteres	120
C.4.4. US09 - Como Usuario quiero seleccionar en qué parte de la imagen se encuentra el texto a reconocer	121
C.4.5. US10 - Como usuario quiero que se traduzcan los caracteres reconocidos de una imagen para poder entender el contenido del texto	122
C.4.6. Revisión de Sprint	122
C.5. Sprint 3: 1 de abril – 24 de abril 2016	124
C.5.1. Pila del producto	124

- C.5.2. US36 - Como Product Owner quiero tener un modelo de dominio del sistema y una descripción de las clases que lo forman 124
- C.5.3. US39 - Como Product Owner quiero tener un diseño de la arquitectura del sistema, así como una descripción de los patrones de diseño utilizados 125
- C.5.4. Revisión de Sprint 126
- C.6. Sprint 4: 25 de abril – 15 de mayo 2016 127
 - C.6.1. Pila del producto 127
 - C.6.2. US46 - Como Product Owner quiero que el código de la aplicación siga el patrón arquitectónico seleccionado, en este caso MVP (Modelo-Vista-Presentador) 127
 - C.6.3. US11 - Como Usuario quiero seleccionar un idioma de reconocimiento 128
 - C.6.4. US12 - Como Usuario quiero seleccionar un idioma al que traducir el texto reconocido . . 129
 - C.6.5. Revisión de Sprint 130
- C.7. Sprint 5: 16 de mayo – 5 de junio 2016 131
 - C.7.1. Pila del producto 131
 - C.7.2. US63 - Como Product Owner quiero tener un modelo de datos del sistema con su correspondiente esquema relacional 131
 - C.7.3. US66 - Como Usuario quiero guardar una traducción realizada 132
 - C.7.4. US70 - Como Usuario quiero visualizar las traducciones guardadas 132
 - C.7.5. Revisión de Sprint 133
- C.8. Sprint 6: 7 de junio – 30 de junio 2016 135
 - C.8.1. Pila del producto 135
 - C.8.2. US74 - Como Product Owner quiero que los idiomas aparezcan como una tabla más en el modelo de datos y que se actualice la base de datos SQLite con estos cambios 135
 - C.8.3. US78 - Como Product Owner quiero tener la documentación actualizada en caso de añadir nueva funcionalidad a la aplicación 136
 - C.8.4. US82 - Como Usuario quiero consultar el detalle de una traducción 136
 - C.8.5. US85 - Como Usuario quiero borrar una traducción 137
 - C.8.6. US89 - Como Usuario quiero buscar una traducción 138
 - C.8.7. US93 - Como Usuario quiero filtrar las traducciones 138
 - C.8.8. US13 - Como Usuario quiero disponer del manual de usuario 139
 - C.8.9. Revisión de Sprint 139

D. Contenido del CD 141

Lista de figuras

3.1. Roles durante el proyecto	15
5.1. Requisitos completos / evolutivos	35
6.1. Diagrama de casos de uso	39
6.2. Modelo de dominio	43
6.3. Diagrama de actividad del CU01. Parte 1	45
6.4. Diagrama de actividad del CU01. Parte 2	46
6.5. Diagrama de actividad del CU02	47
6.6. Diagrama de actividad del CU03	47
6.7. Diagrama de actividad del CU04	48
6.8. Diagrama de actividad del CU05	49
6.9. Diagrama de actividad del CU06	50
7.1. Diferencias entre MVC y MVP	51
7.2. Patrón MVP	52
7.3. Ejemplo de patrón Agente o Broker	52
7.4. Ejemplo de patrón singleton	53
7.5. Modelo general de la arquitectura	53
7.6. Descomposición modular	54
7.7. Diseño detallado del módulo Vista	55
7.8. Diseño detallado del módulo Presentador	56
7.9. Diseño detallado del módulo Modelo	57
7.10. Diseño detallado del módulo Datos	58
7.11. Diseño detallado del módulo Helpers	58
7.12. Relación entre módulos CU01	59
7.13. Relación entre módulos CU02	60
7.14. Relación entre módulos CU03	60
7.15. Relación entre módulos CU04	61
7.16. Relación entre módulos CU05	61
7.17. Relación entre módulos CU06	62
7.18. Esquema relacional de la base de datos	63
7.19. Diagrama de secuencia del CU01. Parte 1	64
7.20. Diagrama de secuencia del CU01. Parte 2	65
7.21. Diagrama de secuencia de <i>Obtener todos los idiomas</i>	66
7.22. Diagrama de secuencia de <i>Guardar traducción</i>	67
7.23. Diagrama de secuencia del CU02	68
7.24. Diagrama de secuencia de <i>Obtener todas las traducciones</i>	69
7.25. Diagrama de secuencia de <i>Obtener los filtros seleccionados</i>	70
7.26. Diagrama de secuencia de <i>Actualizar y animar la lista de traducciones</i>	71
7.27. Diagrama de secuencia del CU03	72
7.28. Diagrama de secuencia de <i>Obtener nombre del idioma a partir de su código</i>	73
7.29. Diagrama de secuencia del CU04	74
7.30. Diagrama de secuencia del CU05	75
7.31. Diagrama de secuencia del CU06	76
7.32. Boceto de la vista principal sin traducciones	77
7.33. Boceto de la vista principal: seleccionar imagen	77

7.34. Boceto de la vista de reconocimiento: seleccionar/descargar idioma de reconocimiento	77
7.35. Boceto de la vista de reconocimiento: seleccionar parte de la imagen y aceptar	78
7.36. Boceto de la vista de traducción: seleccionar idioma de llegada en la traducción	78
7.37. Boceto de la vista de traducción: realizar traducción y guardarla	78
7.38. Boceto de la vista principal con traducciones	79
7.39. Boceto de la vista de detalle de traducción	80
7.40. Boceto de la vista de borrar traducción: deslizar tarjeta para borrar la traducción	81
7.41. Boceto de la vista de borrar traducción: opción para deshacer el borrado	81
7.42. Boceto de la vista de borrar traducción: traducción borrada	81
7.43. Boceto de la vista principal realizando una búsqueda	82
7.44. Boceto de la vista de filtrar traducciones: seleccionar fecha	83
7.45. Boceto de la vista de filtrar traduccione: seleccionar localización	83
7.46. Boceto de la vista de filtrar traducciones: aceptar filtros seleccionados	83
7.47. Boceto de la vista de filtrar traducciones: traducciones filtradas	84
B.1. Instrucciones para obtener imagen de la galería	101
B.2. Instrucciones para obtener imagen desde la cámara	102
B.3. Instrucciones para descargar un idioma de reconocimiento	102
B.4. Instrucciones para borrar un idioma de reconocimiento	103
B.5. Instrucciones para realizar el reconocimiento óptico de caracteres	103
B.6. Instrucciones para realizar la traducción	104
B.7. Instrucciones para guardar la traducción	105
B.8. Visualizar traducciones	106
B.9. Instrucciones para consultar el detalle de una traducción	107
B.10. Instrucciones para borrar una traducción y deshacer su borrado	108
B.11. Instrucciones para borrar una traducción definitivamente	108
B.12. Instrucciones para buscar traducciones	109
B.13. Instrucciones para filtrar traducciones	110
C.1. Estructura de Scrum	111

Lista de tablas

2.1. Descripción del ordenador portátil utilizado	5
2.2. Descripción del dispositivo móvil Android utilizado	6
3.1. Definición de roles	15
3.2. Definición de responsabilidades	15
3.3. Definición de métodos operativos del equipo	16
3.4. Iteraciones del proyecto	17
3.5. Puntos estimados por sprint	17
3.6. Iteraciones del proyecto	18
3.7. Desviación de puntos	18
3.8. Entregas e hitos	19
4.1. Riesgo 1	23
4.2. Riesgo 2	24
4.3. Riesgo 3	25
4.4. Riesgo 4	26
4.5. Riesgo 5	27
4.6. Riesgo 6	28
4.7. Riesgo 7	29
4.8. Riesgo 8	30
4.9. Riesgo 9	31
4.10. Riesgo 10	32
4.11. Riesgo 11	33
4.12. Riesgos producidos	34
5.1. Comparativa de especificación de requisitos entre un proyecto tradicional y otro ágil	35
5.2. Requisitos funcionales del sistema	36
5.3. Requisitos no funcionales del sistema	37
5.4. Requisitos no funcionales de información del sistema	37
6.1. Descripción del CU01	40
6.2. Descripción del CU02	41
6.3. Descripción del CU03	41
6.4. Descripción del CU04	42
6.5. Descripción del CU05	42
6.6. Descripción del CU06	43
8.1. Descripción del CP01	87
8.2. Descripción del CP02	87
8.3. Descripción del CP03	87
8.4. Descripción del CP04	87
8.5. Descripción del CP05	88
8.6. Descripción del CP06	88
8.7. Descripción del CP07	88
8.8. Descripción del CP08	88
8.9. Descripción del CP09	89
8.10. Descripción del CP10	89
8.11. Descripción del CP11	89

8.12. Descripción del CP12	90
8.13. Descripción del CP13	90
8.14. Descripción del CP14	90
8.15. Descripción del CP15	91
8.16. Descripción del CP16	91
8.17. Descripción del CP17	91
8.18. Descripción del CP18	92
8.19. Descripción del CP19	92
8.20. Descripción del CP20	92
8.21. Descripción del CP21	93
8.22. Descripción del CP22	93
8.23. Descripción del CP23	94
C.1. Pila del producto inicial del Sprint 0	114
C.2. Pila del producto inicial del Sprint 1	116
C.3. Pila del producto inicial del Sprint 2	119
C.4. Pila del producto inicial del Sprint 3	124
C.5. Pila del producto inicial del Sprint 4	127
C.6. Pila del producto inicial del Sprint 5	131
C.7. Pila del producto inicial del Sprint 6	135

Capítulo 1

Introducción

1.1. Contexto y motivación

El reconocimiento óptico de caracteres u OCR, del inglés *Optical Character Recognition*, es un proceso de digitalización de textos en el que se identifican símbolos o caracteres de un alfabeto dado a partir de una imagen [1]. Una vez reconocidos los símbolos o caracteres, el texto resultante puede utilizarse para diversos fines.

La popularidad de los teléfonos inteligentes o *smartphones* combinada con la mejora de sus cámaras, ha conllevado a un aumento en el uso de este tipo de técnicas de reconocimiento y a una nueva categoría de aplicaciones móviles que hacen uso de ellas.

Uno de los contextos en los que más utilizamos nuestros *smartphones* es cuando salimos de nuestro lugar de residencia y viajamos a otra ciudad, ya sea de nuestro país o de otro diferente.

Cuando realizamos un viaje al extranjero solemos usar traductores, ya sea en el móvil o en formato papel, pero esto no nos presenta ninguna dificultad, siempre y cuando el alfabeto del idioma de destino sea el mismo que el nuestro y podamos escribir los caracteres y/o buscarlos. La dificultad aparece cuando queremos traducir palabras en otros idiomas que no usan el mismo alfabeto que el nuestro. En este marco nace *Asistev*, que procede de “ASISStente Android para Traducción de textos Escritos en Viajes”.

La solución propuesta en este TFG es el desarrollo de una aplicación móvil para Android capaz de reconocer los caracteres contenidos en una imagen y traducirlos a otro idioma. La imagen puede ser obtenida tanto con la cámara del dispositivo como desde la galería.

Para el reconocimiento, se hace uso de las técnicas de reconocimiento óptico de caracteres u OCR descritas anteriormente.

El presente documento es la propuesta elaborada para la asignatura TFG del Grado en Ingeniería Informática de la Universidad de Valladolid.

1.2. Objetivos

El objetivo fundamental de este trabajo de fin de grado consiste en el desarrollo de una aplicación móvil para Android que, haciendo uso de técnicas de reconocimiento óptico de caracteres, reconozca el texto contenido en una imagen y lo traduzca al idioma seleccionado.

A partir de este objetivo principal, surgen otros objetivos secundarios:

- Estudiar diferentes bibliotecas de reconocimiento óptico de caracteres y traducción.
- Aprender a utilizar una biblioteca OCR.
- Aprender a utilizar Android NDK.
- Aprender a utilizar un API de traducción.
- Diseñar una interfaz sencilla, intuitiva y atractiva.

- Familiarizarse con la metodología Scrum, muy utilizada en empresas del sector.
- Planificar un proyecto software y aprender a gestionar los imprevistos que se puedan producir.
- Afianzar los conocimientos de LaTeX para la creación de documentos profesionales.
- Elaborar la memoria de un proyecto software.

1.3. Metodología utilizada

Para el desarrollo de este proyecto se ha decidido utilizar el modelo de desarrollo ágil Scrum.

Este modelo se caracteriza por [2]:

- Adoptar una estrategia de desarrollo incremental en varias iteraciones o *sprints*.
- Permitir el solapamiento de diferentes fases de desarrollo.
- Flexibilidad ante los cambios.
- Entrega temprana y continuada de software funcional.
- Mitigación anticipada y sistemática de los riesgos.

En el apartado 3.2.3 del Plan de Desarrollo de Software se definen las características de este proyecto y en base a estas características, en el apartado 3.2.4, se explican en detalle las razones de utilizar este tipo de modelo frente a otros modelos de proceso.

1.4. Resumen del contenido de la memoria

En este apartado se describen brevemente los diferentes capítulos y secciones que componen el resto de la memoria de este proyecto:

- **Capítulo 1. Introducción:** se contextualiza y motiva el tema elegido para el proyecto, se exponen los objetivos, la metodología utilizada y el presente resumen.
- **Capítulo 2. Entorno tecnológico:** se explican las diferentes herramientas utilizadas durante el desarrollo del proyecto, así como el entorno de desarrollo.
- **Capítulo 3. Plan de Desarrollo de Software:** se expone una visión global del proyecto que incluye, entre otros, las suposiciones y restricciones, las características del proyecto y los entregables, los participantes del proyecto con los roles asignados y la planificación.
- **Capítulo 4. Plan de Gestión de Riesgos:** se identifican y analizan los riesgos del proyecto, se establecen planes de acción y se detalla el control de los riesgos producidos.
- **Capítulo 5: Requisitos:** se exponen los requisitos funcionales, no funcionales y de información.
- **Capítulo 6: Análisis:** se enumeran los actores del sistema, los casos de uso y su especificación. Se presenta el modelo de dominio y los diagramas de actividad de los casos de uso identificados.
- **Capítulo 7: Diseño:** se presentan los patrones arquitectónicos y de diseño utilizados. Se expone el modelo general de la arquitectura, la descomposición modular del sistema, el modelo de datos y la realización de los casos de uso en diseño mediante diagramas de secuencia. Por último, se muestran los diseños de la interfaz de usuario para cada uno de los casos de uso.

- **Capítulo 8: Implementación y pruebas:** se explican las diferentes versiones de la aplicación, la estructura del código fuente y se presentan las diferentes pruebas realizadas para cada uno de los casos de uso implementados.
- **Capítulo 9: Conclusiones:** se exponen los objetivos alcanzados, la valoración personal del alumno y se presentan las líneas futuras del proyecto.
- **Bibliografía:** se muestra la documentación consultada durante todo el desarrollo del proyecto.
- **Anexos:** se presenta un glosario de términos, el manual de usuario de la aplicación desarrollada y el Plan de Seguimiento con el detalle del trabajo realizado en cada sprint.

Capítulo 2

Entorno tecnológico

2.1. Herramientas utilizadas

Las herramientas utilizadas durante el desarrollo de este proyecto han sido:

- **Android Studio:** herramienta de desarrollo integrado (IDE) utilizada para desarrollar la aplicación para Android.
- **AVD Manager:** herramienta para crear y gestionar emuladores Android.
- **Dropbox** [3]: servicio de alojamiento en la nube usado para almacenar los ficheros necesarios.
- **GitHub** [4]: plataforma de desarrollo colaborativo de software usada para alojar el código de la aplicación y para gestionar el control de versiones con Git.
- **NinjaMock:** herramienta web para realizar el prototipado de la aplicación y establecer el flujo de navegación entre pantallas.
- **Taiga:** plataforma open source de gestión de proyectos ágiles.
- **ShareLaTeX** [5]: editor de LaTeX online utilizado para la redacción de la memoria del proyecto.

2.2. Entorno de desarrollo

El entorno de trabajo durante todo el desarrollo del proyecto ha sido un ordenador portátil de finales del año 2013, que cuenta con las siguientes características.

Equipo	
Marca	Hewlett Packard
Modelo	Envy 15 j031 ss
Hardware	
Procesador	Intel® Core™ i7-4700MQ @ 2.4GHz (4 cores)
Memoria	8 GB SDRAM DDR3L
Disco duro	750 GB sATA 5400 rpm.
Tarjeta gráfica	NVIDIA GeForce GT 740M
Memoria gráfica	2 GB DDR3 dedicados %
Tamaño pantalla	15,6" (39,62 cm) retroiluminada LED
Resolución	1920 x 1080 Full HD
Software	
OS 1	Windows 10 Pro
OS 2	Ubuntu 14.04

Tabla 2.1: Descripción del ordenador portátil utilizado

Aparte del ordenador portátil, se han utilizado diversos dispositivos móviles Android para testear la aplicación, siendo el más utilizado el descrito en la siguiente tabla.

Dispositivo	
Marca	LG
Modelo	Nexus 5x
Hardware	
Procesador	Qualcomm MSM8992 Snapdragon 808 @ 1.8GHz (6 cores)
Memoria	2 GB LPDDR3
Almacenamiento	32 GB
Procesador gráfico	Adreno 418
Tamaño pantalla	5,2"LCD IPS
Resolución	1920 x 1080 Full HD
Software	
OS 1	Android 6.0.1 Marshmallow

Tabla 2.2: Descripción del dispositivo móvil Android utilizado

Los otros tres dispositivos móviles con los que se han realizado más pruebas son:

- **Samsung Galaxy Grand Neo Plus:** dispositivo de gama media con versión de Android 5.1 Lollipop.
- **Nexus 4:** dispositivo de gama media-alta con versión de Android 5.1.1 Lollipop.
- **Samsung Galaxy S3:** dispositivo de gama alta en el momento de su lanzamiento. Versión de Android 4.3. KitKat.

Capítulo 3

Plan de Desarrollo de Software

3.1. Introducción

Esta parte de la memoria proporciona una visión global del desarrollo del proyecto.

Este proyecto se llevará a cabo basándose en el modelo de desarrollo de software ágil Scrum. La decisión de utilizar este tipo de metodología se detalla en el capítulo 3.2.4.

3.1.1. Propósito

El propósito de este Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el desarrollo del proyecto *Asistev*.

Los usuarios de este documento son:

- El alumno, que se encargará de analizar, diseñar, planificar y desarrollar el proyecto.
- El tutor del TFG, que establecerá los requisitos iniciales, se encargará de revisar la documentación y el desarrollo del proyecto, aportando ideas y posibles soluciones.

3.1.2. Alcance

En este documento se incluye una visión general del proyecto y su organización, se identifican los participantes y sus roles, se desarrolla un plan de trabajo en iteraciones y se muestra otra información relevante para el desarrollo del proyecto.

3.1.3. Resumen del plan

El resto del plan se divide en las siguientes secciones:

- **Visión general del proyecto**, que a su vez contiene:
 - Objetivos del proyecto.
 - Suposiciones y restricciones.
 - Características del proyecto.
 - Elección del modelo de proceso y metodología.
 - Elección del tipo de desarrollo y bibliotecas de reconocimiento y traducción a utilizar.
 - Entregables del proyecto.
- **Organización del proyecto**: incluye la estructura organizativa, así como los roles y responsabilidades del proyecto.
- **Planificación del proyecto** incluye las estimaciones del proyecto, las desviaciones y las entregas e hitos.

- **Seguimiento del proyecto** incluye una referencia al Plan de Seguimiento del proyecto.

3.2. Visión general del proyecto

3.2.1. Objetivos

El objetivo fundamental de este proyecto consiste en el desarrollo de *Asistev*, una aplicación Android que facilitará a los usuarios el transcurso de sus viajes, permitiéndoles reconocer más de 30 idiomas y poder traducir entre ellos. El texto a reconocer y traducir puede ir desde carteles, guías, menús hasta cualquier otro que se vayan encontrando durante su visita.

Además del reconocimiento de caracteres y la traducción, *Asistev* ofrece otra serie de funcionalidades como son:

- **Detalle de traducción:** el usuario puede consultar el detalle de cada traducción, que incluye la imagen de la cual se ha realizado el reconocimiento, la fecha y localización en las que se ha realizado la traducción, el idioma del texto reconocido, el texto reconocido, el idioma al que se ha traducido el texto y el texto traducido.
- **Historial de traducciones:** las traducciones realizadas quedan guardadas con la fecha en las que se han realizado y la localización. Esto es muy útil si se realizan visitas de varios días o rutas por diferentes países, dando al usuario la opción de consultar las traducciones cuando lo desee.
- **Búsqueda y filtrado de traducciones:** el usuario puede buscar entre las diferentes traducciones y filtrarlas por fecha y/o localización.

3.2.2. Suposiciones y restricciones

El proyecto debe cumplir las siguientes restricciones:

1. Restricciones de presupuesto:

- Se deben utilizar servicios gratuitos o bibliotecas de libre distribución para realizar el reconocimiento óptico de caracteres y la traducción.

2. Restricciones de recursos:

- El equipo de trabajo está formado únicamente por dos personas, tutor y alumno.

3. Restricciones de la aplicación:

- La aplicación debe ser desarrollada para el sistema operativo Android.
- La aplicación debe utilizar un sistema de gestión de base de datos relacional de tipo SQLite.

3.2.3. Características del proyecto

Antes de seleccionar el modelo de proceso y la metodología a utilizar, es fundamental definir las características de este proyecto, para tomar una decisión acorde a las mismas.

Las características de este proyecto son:

- Proyecto de corta duración, concretamente 300 horas.
- Realizado por una única persona, con el seguimiento del tutor correspondiente, que guiará y aclarará los requisitos del proyecto.

- Entrega de documentación completa, que incluya todos los modelos, diagramas y diseños correspondientes, así como la planificación, gestión de riesgos y otros.
- Se utilizarán técnicas de reconocimiento óptico de caracteres u OCR, las cuales no son sencillas de implementar y utilizar. Esto nos limita a no saber desde el inicio qué se podrá realizar o no en el tiempo estipulado.
- Se esperan cambios en el proyecto. Se partirá del reconocimiento de caracteres de una serie de idiomas y su traducción correspondiente. En función de cómo vaya avanzando el proyecto, se podrá añadir soporte a nuevos idiomas y otras funcionalidades.
- La arquitectura no es sencilla de definir de forma temprana debido a las peculiaridades de las bibliotecas OCR, por lo que se irá definiendo y mejorando a lo largo del proyecto.
- Según va avanzando el proyecto, los requisitos pueden ser repriorizados, adelantando o atrasando la implementación de cada uno de ellos.
- Es fundamental una buena gestión de riesgos, ya que la biblioteca OCR puede generar fallos y retrasos en el proyecto. En cada iteración del proyecto se irán eligiendo una serie de requisitos a implementar y solo se hará frente a los riesgos de esos requisitos, haciendo posible iniciar una mitigación de manera anticipada y sistemática.

3.2.4. Elección del modelo de proceso y metodología para desarrollo de software

En este apartado se comparan los diferentes modelos de proceso y se justifica por qué se ha escogido un modelo de desarrollo de software ágil.

3.2.4.1. Modelo en cascada

Uno de los modelos más utilizados a la hora de desarrollar software en el ámbito académico y empresarial es el modelo de desarrollo tradicional en cascada [6]. En este caso, no se adapta a este proyecto por los siguientes motivos:

- El modelo en cascada define una secuencia lineal en la que una fase no comienza hasta que finaliza la fase anterior, lo cual es incompatible con este proyecto, en el que se pueden solapar diferentes fases del desarrollo.
- Se tarda mucho tiempo en ver un resultado del software y este proyecto requiere implementar funcionalidades poco a poco e ir viendo los resultados, debido a la dificultad añadida de las bibliotecas OCR.
- Es poco flexible ante cambios en los requisitos. No permite correcciones sobre la marcha.
- Es poco efectivo en gestión de riesgos ya que los problemas se dejan para su posterior resolución, lo que puede provocar que sean ignorados o corregidos sin seguir unas pautas establecidas.
- El tutor del TFG no puede ver el progreso, ya que hasta las etapas finales no se obtiene un producto software funcional.

3.2.4.2. UP

Otro de los modelos más utilizados es UP o Proceso Unificado [7], que se ajusta mejor a las necesidades de este proyecto respecto al modelo en cascada por los siguientes motivos:

- Basado en un desarrollo iterativo e incremental. Cada iteración implementa un subconjunto de la funcionalidad total.

- Flexibilidad, se pueden realizar cambios y mejoras en cada iteración.
- Se elabora una documentación exhaustiva.
- Progreso visible en las etapas tempranas.

No obstante, hay otra serie de características que no se adaptan a este proyecto como son:

- Grado de complejidad elevado, requiere de un gran esfuerzo para aplicarlo en un periodo de tiempo tan limitado.
- Orientado a proyectos empresariales de largo alcance y tiempo.
- El alcance del proyecto se realiza antes del comienzo del proyecto y puede ser revisado, pero existe un proceso estricto que controla los cambios. En este proyecto el alcance puede cambiar a lo largo del desarrollo del mismo y tiene que poder ser modificado de forma sencilla.
- Plan formal de proyecto con varias iteraciones que lo cubren de principio a fin, pero en este proyecto puede no conocerse con seguridad el plan de la siguiente iteración hasta que no se termine la iteración previa.

Por estos motivos, se ha decidido descartar UP como modelo de proceso de software a utilizar en este proyecto.

3.2.4.3. Scrum

Actualmente, las metodologías ágiles están teniendo una buena acogida en nuevos proyectos de desarrollo software, pero hay que tener en cuenta que no todos los proyectos son buenos candidatos para este tipo de metodologías.

Observando las características de este proyecto, lo podríamos resumir en cuatro conceptos:

- **Adaptable:** pueden producirse cambios según va avanzando el proyecto y debe existir respuesta antes esos cambios.
- **Incremental e iterativo:** las funcionalidades se van añadiendo incrementalmente, en varias iteraciones.
- **Cooperativo:** realimentación con el tutor del TFG mediante varias entregas de documentación y software funcional.
- **Sencillo:** se debe seguir un método fácil de aprender y modificar, bien documentado. Se utilizará la simplicidad para maximizar la cantidad de trabajo en 300 horas.

Estas características se corresponden con las definidas en las metodologías ágiles, pero a pesar de todo, hay una característica que nuestro proyecto tiene y no parece estar definida en las metodologías ágiles, y es la documentación completa.

Las metodologías ágiles anteponen el software que funciona sobre la documentación exhaustiva, pero no abogan por evitar, suprimir o eliminar la documentación, es decir, son totalmente compatibles con la documentación, a pesar de lo que comúnmente se enseña sobre estas metodologías.

También es cierto que el manifiesto ágil [8] aconseja, siempre que sea posible, reducir al mínimo indispensable el uso de documentación; no obstante, debido a las condiciones de entrega de un TFG, es necesario la entrega de una documentación completa, que la flexibilidad de Scrum permite.

Por lo tanto, se ha decidido seguir una de las metodologías ágiles más conocidas, como es Scrum, al ser la que mejor se adapta a las necesidades del proyecto.

3.2.5. Elección del tipo de desarrollo y de bibliotecas OCR y traducción

En este apartado se explica el tipo de desarrollo que se llevará a cabo y la justificación de las bibliotecas OCR y de traducción seleccionadas.

3.2.5.1. Elección del tipo de desarrollo

Aunque una de las restricciones del proyecto sea desarrollar la aplicación para Android, es interesante conocer las alternativas que habría a este tipo de desarrollo.

Actualmente existen diversas formas para realizar una aplicación móvil, lo que conlleva a decidir si una aplicación será nativa, es decir, proyectada específicamente a un sistema operativo como Android o iOS o será una aplicación multidispositivo desarrollada en HTML5 y que haga uso de la tecnología web.

En el caso de este proyecto, la aplicación a desarrollar hará uso de una biblioteca OCR importada en la propia aplicación; por consiguiente, la aplicación aprovechará características específicas del hardware para realizar el reconocimiento óptico de caracteres, siendo necesario el uso de las características nativas del dispositivo.

Por lo tanto, no hubiera sido sencillo realizar la aplicación híbrida, dejando la opción de desarrollo nativo como la más adecuada. Para desarrollar para Android de forma nativa, son necesarias las herramientas de desarrollo para Android de Google, en este caso Android Studio.

Se podría utilizar otro IDE como Eclipse pero Android Studio actualmente se encuentra en una versión muy estable y cuenta con más funcionalidades y facilidades para el desarrollo en Android.

3.2.5.2. Elección de biblioteca OCR

Para el reconocimiento óptico de caracteres existen varias opciones, que podemos clasificar en función del lugar donde se lleve a cabo el proceso OCR:

1. En el propio dispositivo donde se ejecute la aplicación:

- Ventajas:
 - a) No es necesaria una conexión a internet.
 - b) Evita un consumo excesivo de datos móviles, ya que las fotografías obtenidas en los móviles actuales pueden ser potencialmente grandes y al ser enviadas a un servidor puede agotar nuestra tarifa de datos de internet.
- Inconvenientes:
 - a) El procesado de las imágenes puede ser más lento que en un servidor con hardware más potente.
 - b) Es necesario descargarse un fichero para cada idioma a reconocer, lo cual puede mermar la memoria del dispositivo.
 - c) Posible aumento de consumo de batería.

2. En un servidor propio o alquilado que implemente un servicio OCR gestionado por el alumno:

- Ventajas:
 - a) Mayor capacidad de procesamiento de imágenes y reconocimiento de caracteres, por lo que devuelve el resultado más rápido.
- Inconvenientes:

- a) Servidor puede sufrir sobrecargas y no devolver el resultado tan rápido como se esperaba.
- b) Es necesario en todo momento una conexión a internet, la cual no siempre es fácil de obtener en los países a visitar.
- c) Necesidad de adquirir un servidor donde implementar el servicio OCR, el cual será llamado desde la aplicación. Esto implicará unos costes derivados.

3. Mediante la adquisición de una aplicación comercial o COTS, que permita la utilización de un servidor con OCR:

- Ventajas:

- a) El mantenimiento del servidor es realizado por la empresa propietaria.
- b) La actualización y mejora del sistema de reconocimiento de caracteres es llevado a cabo por la empresa propietaria.

- Inconvenientes:

- a) Poca flexibilidad. No nos permite mejorar o adaptar el reconocimiento de caracteres a la aplicación a desarrollar.
- b) Coste elevado, distribuido en cuotas mensuales o anuales.

Teniendo en cuenta estas consideraciones [9] y la restricción de utilizar servicios o bibliotecas de libre distribución, se ha seleccionado un motor de OCR denominado *Tesseract*, que actualmente es desarrollado por Google, distribuido bajo la licencia *Apache* y considerado como uno de los motores OCR libres con mayor precisión disponibles actualmente [10].

Existen varias posibilidades para utilizar *Tesseract*:

- Ejecutándolo como un servicio en la nube almacenado en un servidor Linux y accediendo desde la aplicación cuando sea necesario.
- Construyendo una biblioteca nativa que será enlazada con la aplicación Android.

De estas dos posibilidades se ha seleccionado la segunda ya que puede que el usuario no tenga conexión a internet en un momento determinado o haya consumido todos sus datos y se encuentre en situaciones en las que necesita traducir algún texto.

Se ha decidido utilizar un fork de *Tesseract Android Tools* denominado *tess-two* [11] y desarrollado por Robert Theis.

También existen otro software OCR de libre distribución como *FreeOCR* [12] o *OCRopus* pero no son tan precisos, soportan menos idiomas (34 de *Tesseract* frente a 25 de *FreeOCR*) y no tienen bibliotecas que se puedan usar para desarrollar con Android.

El resto de software OCR a destacar tienen licencia propietaria, por lo que no pueden ser utilizados para el desarrollo del proyecto.

3.2.5.3. Selección de biblioteca de traducción

Una vez reconocidos los caracteres, el texto será traducido a un idioma dado.

La traducción del texto reconocido se puede realizar de cuatro formas diferentes:

1. Con un diccionario de traducción incluido en el propio dispositivo:

- Ventajas:

a) No es necesaria una conexión a internet.

■ Inconvenientes:

a) Los paquetes de idiomas ocupan mucho espacio, en torno a 100 MB por cada idioma.

b) La traducción puede ralentizarse si se trata de textos muy largos o complejos.

2. Con un servicio de traducción implementado en un servidor gestionado por el alumno:

■ Ventajas:

a) Se pueden almacenar todos los paquetes de idiomas que sean necesarios, sin preocuparse de la capacidad del dispositivo.

b) Traducción más rápida y precisa al tener mayor capacidad de procesado.

■ Inconvenientes:

a) Servidor puede sufrir caídas o sobrecargas y no realizar la traducción o tardar más de lo esperado en devolver el texto traducido.

b) Es necesaria una conexión a internet, la cual no siempre es fácil de conseguir en algunos países como hemos dicho anteriormente.

c) Costes de alquiler y mantenimiento del servidor.

3. Con un servicio de traducción completo o parcialmente gratuito que incluya un API al que invocar desde la aplicación:

■ Ventajas:

a) Mantenimiento del servidor y del sistema de traducción realizado por la comunidad si es de código abierto o por la empresa propietaria del servicio, si es parcialmente gratuito.

b) Traducciones de calidad.

c) Traducciones a una gran variedad de idiomas.

■ Inconvenientes:

a) Límite de traducciones gratuitas en caso de servicios parcialmente gratuitos. Al superar este límite es necesario pagar una cuota.

4. Con un servicio de traducción de pago:

■ Ventajas:

a) Mantenimiento del servidor y del sistema de traducción realizado por la empresa propietaria del servicio.

b) La mejora y actualización de nuevas palabras e idiomas es realizado por la empresa propietaria.

c) Traducción de calidad, con muchos idiomas disponibles.

■ Inconvenientes:

a) Poca flexibilidad.

b) Coste elevado.

Siguiendo con la idea de permitir el uso de la aplicación en modo offline, lo lógico sería utilizar un servicio de traducción que permitiera descargar paquetes de idiomas para usarlos sin conexión.

No obstante, después de una búsqueda exhaustiva, no ha sido posible encontrar servicios de traducción gratuitos, de calidad y con esta funcionalidad. La única opción de este tipo es *Apertium* [13], una plataforma libre de traducción automática que, después de haber realizado varias pruebas con ella, se ha considerado que realiza una traducción demasiado literal y aún le faltan muchos idiomas por implementar.

Por lo tanto, teniendo en cuenta estas consideraciones y la restricción de utilizar servicios gratuitos o bibliotecas de libre distribución, se ha decidido utilizar el API de traducción de *Microsoft*. Este servicio es gratuito hasta la traducción de 2 millones de caracteres y ofrece multitud de idiomas y una traducción de calidad [14].

La parte negativa de esto es que sería necesario una conexión a internet para realizar la traducción, pero Microsoft ya está trabajando en traducciones offline que pueden ser añadidas a la aplicación en un futuro.

Por último, cabe destacar que el API de traducción de Google es de pago y tampoco permite solicitar traducciones a su aplicación Android desde otras aplicaciones. Esto nos imposibilita su uso en el desarrollo de *Asistev*.

3.2.6. Entregables del proyecto

En este apartado se describen cada uno de los documentos que serán utilizados y entregados en el proyecto.

Es preciso destacar que de acuerdo a la filosofía ágil, todos los entregables o artefactos están sujetos a cambios y la versión definitiva se presentará al término del proceso. Por ello, el resultado de cada iteración permite alcanzar un cierto grado de completitud y estabilidad de los artefactos.

A continuación, se listan los correspondientes entregables:

- Plan de Desarrollo de Software.
- Plan de Gestión de Riesgos.
- Especificación de requisitos.
- Modelo de Análisis.
- Modelo de Diseño y arquitectura.
- Modelo de Datos.
- Prototipos de la interfaz de usuario.
- Implementación y Plan de Pruebas.
- Manual de usuario.
- Versiones alfa, beta y final del producto.
- Plan de Seguimiento.

3.3. Organización del proyecto

3.3.1. Interfaces externas

La interfaz externa de este proyecto es el Product Owner o cliente.

3.3.2. Estructura interna

Como podemos ver en la figura 3.1, la estructura interna del proyecto está compuesta por dos únicos roles.



Figura 3.1: Roles durante el proyecto

3.3.3. Roles y responsabilidades

Las funciones de cada uno de los roles del proyecto se muestra en la tabla 3.1 y la asignación de dichos roles a los participantes del TFG en la tabla 3.2.

Roles	Competencias
Product Owner (cliente)	Encargado de definir los objetivos del producto, de dirigir los resultados del proyecto y colaborar con el equipo para planificar, revisar y detallar los objetivos a realizar en cada iteración. [15]
Scrum Master (facilitador)	Encargado de apoyar al equipo de desarrollo, eliminar las barreras organizativas y mantener la consistencia del proyecto ágil.
Team (equipo)	Lo forman el grupo de profesionales que realizan el incremento de cada iteración del proyecto. Es un equipo multifuncional y autoorganizado, los miembros pueden pasar por diferentes roles a lo largo del proyecto: arquitecto, analista, diseñador, desarrollador y tester. [8]

Tabla 3.1: Definición de roles

Nombre	Rol
Miguel Ángel Laguna	Product Owner (cliente)
David González Verdugo	Scrum Master (facilitador) y Team (miembro del equipo)

Tabla 3.2: Definición de responsabilidades

Nota importante: aunque Miguel A. L. asume el rol de Product Owner, en algunas fases David G.V. ha tomado el rol de Product Owner, proponiendo al tutor nuevas funcionalidades y requisitos que añadir al proyecto y recibiendo apoyo y soporte del mismo.

3.3.4. Eventos o métodos operativos del equipo

Al seguir Scrum como modelo de desarrollo, se realizarán los eventos correspondientes a la tabla 3.3 [8].

Método operativo	Descripción
Planificación del proyecto	Es la planificación inicial. Incluye todo lo presente en este documento.
Reunión de Planificación del Sprint	Se realiza al comienzo de cada sprint y es donde se fijan los objetivos del mismo. También se identifican las tareas a realizar para completar cada requisito y se estima el tiempo de cada una de ellas.
Scrum diario	Al comienzo de cada jornada de trabajo el alumno identifica el trabajo realizado el día anterior y el trabajo a realizar durante ese día. También se estudian los obstáculos de días anteriores y si se pueden solucionar para el día actual.
Revisión de Sprint	Reunión al final de cada sprint, donde se mostrará al Product Owner la documentación realizada o funcionalidad añadida al producto durante ese sprint.
Retrospectiva del Sprint	Se realiza una vez finalizado un sprint. Se analiza qué ha ido bien en ese sprint, qué cosas se podrían cambiar y como realizar los cambios.

Tabla 3.3: Definición de métodos operativos del equipo

3.4. Planificación del proyecto

3.4.1. Estimaciones del proyecto

3.4.1.1. Estimación temporal

En la tabla 3.4 se representan los sprints planificados para el desarrollo del proyecto, con sus fechas de inicio y fin aproximadas. Se ha decidido realizar 6 sprints con una duración de 3 semanas cada uno y un último sprint final de 3 semanas y 4 días.

Sprint	Fecha de inicio	Fecha de fin
0	1 de febrero de 2016	21 de febrero de 2016
1	22 de febrero de 2016	13 de marzo de 2016
2	14 de marzo de 2016	3 de abril de 2016
3	4 de abril de 2016	24 de abril de 2016
4	25 de abril de 2016	15 de mayo de 2016
5	16 de mayo de 2016	5 de junio de 2016
6	6 de junio de 2016	30 de junio de 2016

Tabla 3.4: Iteraciones del proyecto

Considerando como días de trabajo de lunes a viernes y jornadas de 4 horas hombre/día, realizamos el siguiente cálculo para estimar las horas totales a realizar:

- $4 \text{ horas hombre/día} \times 5 \text{ días/semana} \times 3 \text{ semanas/sprint} \times 7 \text{ sprints} + 16 \text{ horas de los 4 días extra del último sprint} = \mathbf{436 \text{ horas hombre}}$

3.4.1.2. Estimación de puntos

Para cada uno de los sprints se han estimado una serie de puntos. Este tipo de medida es muy utilizada en metodologías ágiles y representan la complejidad de cada historia de usuario a implementar en cada sprint.

Sprint	Puntos estimados
0	14 puntos
1	16 puntos
2	16 puntos
3	16 puntos
4	15 puntos
5	14 puntos
6	24.5 puntos
Total	115.5 puntos

Tabla 3.5: Puntos estimados por sprint

3.4.1.3. Estimación de costes

No puede existir ningún tipo de coste económico ya que el alumno es un estudiante universitario y no se va a recibir ninguna retribución por desarrollar este proyecto.

No obstante, se podría poner un ejemplo de estimación basándonos en las horas hombre estimadas y un coste medio de la hora de un Ingeniero Informático:

- 436 horas hombre * 15 €/hora = **6540 €**

3.4.2. Desviaciones del proyecto

Como se puede ver en la tabla 3.6, se han producido desviaciones en las fechas de los siguientes sprints:

- **Sprint 1:** desviación debida al aplazamiento de la realización del modelo de dominio, lo que ha acortado el sprint un día, acabando el jueves 10 de marzo en vez del viernes 11 de marzo.
- **Sprint 2:** afectado por el acortamiento del sprint 1, se adelanta un día y comienza el viernes 11 de marzo.
- **Sprint 3:** su comienzo se adelanta al viernes 1 de abril pero su fecha de finalización se retrasa un día debido a complicaciones en los diagramas de la arquitectura MVP. Este retraso permite que el sprint termine en la fecha planificada y que no afecte a la planificación del resto de sprints.

Sprint	Fecha de inicio	Fecha de fin
0	1 de febrero de 2016	21 de febrero de 2016
1	22 de febrero de 2016	10 de marzo de 2016
2	11 de marzo de 2016	31 de marzo de 2016
3	1 de abril de 2016	24 de abril de 2016
4	25 de abril de 2016	15 de mayo de 2016
5	16 de mayo de 2016	5 de junio de 2016
6	6 de junio de 2016	30 de junio de 2016

Tabla 3.6: Iteraciones del proyecto

En la tabla 3.7 se puede observar como en el sprint 1 se realizan menos puntos de lo planificado y en el sprint 3 se realizan más puntos de lo planificado.

Sprint	Puntos planificados	Puntos realizados
0	14 puntos	14 puntos
1	16 puntos	14 puntos
2	16 puntos	16 puntos
3	16 puntos	18 puntos
4	15 puntos	15 puntos
5	14 puntos	14 puntos
6	24.5 puntos	24.5 puntos
Total	115.5 puntos	115.5 puntos

Tabla 3.7: Desviación de puntos

3.4.3. Entregas e hitos

En la tabla 3.8 se muestran las entregas de documentación o funcionalidad realizadas al final de cada sprint.

Sprint	Fecha de inicio	Fecha de fin	Puntos	Entregas e hitos
0	1 de febrero de 2016	21 de febrero de 2016	14	<ul style="list-style-type: none"> ■ Plan de Desarrollo de Software. ■ Requisitos iniciales. ■ Modelo de Análisis: actores, casos de uso. ■ Plan de Gestión de Riesgos: introducción e identificación de riesgos.
1	22 de febrero de 2016	10 de marzo de 2016	14	<ul style="list-style-type: none"> ■ Prueba de reconocimiento de caracteres. ■ Plan de Gestión de Riesgos: análisis e impacto, planes de acción.
2	11 de marzo de 2016	31 de marzo de 2016	16	<ul style="list-style-type: none"> ■ Versión 0.1 o Alpha de la aplicación . ■ Casos de prueba para la versión 0.1.
3	1 de abril de 2016	24 de abril de 2016	18	<ul style="list-style-type: none"> ■ Modelo de Análisis: Modelo de dominio. ■ Modelo de Diseño: arquitectura, patrones, realización de casos de uso, prototipos de la interfaz de usuario.
4	25 de abril de 2016	15 de mayo de 2016	15	<ul style="list-style-type: none"> ■ Versión 0.2 o Beta de la aplicación. ■ Casos de prueba para la versión 0.2.
5	16 de mayo de 2016	5 de junio de 2016	14	<ul style="list-style-type: none"> ■ Modelo de Datos. ■ Implementación de CU01 completo y CU02 . ■ Casos de prueba para la funcionalidad correspondiente al CU01 y CU02.
6	6 de junio de 2016	30 de junio de 2016	24.5	<ul style="list-style-type: none"> ■ Versión 1.0 o Release de la aplicación. ■ Casos de prueba para la versión 1.0.

Tabla 3.8: Entregas e hitos

3.4.4. Datos interesantes

- **Velocidad media del equipo:** 16.5 puntos/sprint
- **Estimación de puntos/hora:** 0.26 puntos/hora hombre

3.5. Seguimiento del proyecto

Para ver el seguimiento completo del proyecto, consultar el Plan de Seguimiento en el apéndice C.

Capítulo 4

Plan de Gestión de Riesgos

4.1. Introducción

Para asegurar el éxito de un proyecto medido en cumplimiento de plazos, alcance y calidad, es fundamental realizar una correcta Gestión de Riesgos.

Una buena Gestión de Riesgos empieza por la identificación y análisis por adelantado de los riesgos que puedan afectar al proyecto.

4.1.1. Propósito

Este documento incluye el análisis de los riesgos identificados durante la fase inicial del proyecto *Asistev*.

Para cada riesgo identificado se estudiará su efecto y el contexto de aparición. Además, se definirán estrategias para evitar su aparición o minimizar el impacto en el proyecto en el caso de que se produzcan.

Según va avanzando el proyecto, pueden aparecer nuevos riesgos potenciales, que deberán ser clasificados, analizados e incluidos en planes de contingencia que concreten las acciones a realizar para evitar el riesgo o minimizar su impacto en el caso de que se lleve a cabo.

4.1.2. Alcance

El análisis de riesgos es una tarea a realizar durante todo el desarrollo del proyecto, no solo en su fase inicial.

La lista de riesgos se revisará y actualizará durante todo el desarrollo del proyecto.

4.2. Gestión del riesgo

El proceso llevado a cabo por la gestión de riesgos está formado por las siguientes fases:

- **Identificación:** posibles riesgos que se pueden dar a lo largo del proyecto.
- **Análisis de los riesgos:** incluye la probabilidad de que los riesgos ocurran, el impacto que producirían y su evaluación.
- **Planes de acción:** forma de actuar en caso de que se produzcan los riesgos. El objetivo de estos planes es evitar los riesgos o mitigarlos en el peor caso.
- **Seguimiento y control de riesgos:** se realiza una supervisión de los riesgos producidos y sus consecuencias en el proyecto.

4.2.1. Identificación, análisis, planes de acción y seguimiento

Los riesgos identificados atienden a la siguiente clasificación:

- **Riesgo de proyecto:** amenaza el plan de proyecto. Si uno de estos riesgos se originara, podría producir variaciones en el tiempo o costes estimados inicialmente.
- **Riesgo de proceso:**
 - **De gestión:** relacionados con una planificación ineficaz, una documentación escasa o mal elaborada, falta de revisiones y otros.
 - **Técnico:** relacionados con el diseño e implementación de la aplicación. Este tipo de riesgos pueden derivar en un producto de peor calidad y problemas en la planificación.
- **Riesgo de producto:** relacionados con la falta de experiencia, diseños demasiado complejos, interfaces mal definidas, requisitos vagos o incompletos. Amenazan a la viabilidad del producto a desarrollar.

Para analizar los riesgos se han utilizado los siguientes criterios:

- **Probabilidad de ocurrencia:** grado de probabilidad de que el riesgo finalmente se produzca. Se mide en una escala porcentual de 1 a 100.
- **Nivel de impacto:** grado de impacto en el proyecto en el caso de que el riesgo finalmente se produzca. Se mide en una escala de 1 a 5, siendo 1 un impacto bajo y 5 el máximo impacto.
- **Evaluación del riesgo:** valor numérico resultante del producto de la probabilidad de ocurrencia por el nivel de impacto. Este producto representa la prioridad que tendrá la gestión de este riesgo y la implantación de sus medidas preventivas o correctoras.

En los planes de acción y seguimiento aparecen las siguientes variables [16]:

- **Escenario:** define qué sucedería.
- **Indicador:** métrica a ser seguida.
- **Condiciones de disparo:** valores que indican que el riesgo puede haberse producido.
- **Punto de comprobación:** cuándo/dónde comprobar la métrica.
- **Estrategia de resolución:** cómo se debe manejar el riesgo. Las estrategias utilizadas en este proyecto han sido:
 - **Reducción del riesgo:** reducir la probabilidad y/o consecuencia del riesgo después de que ocurra.
 - **Reserva del riesgo:** utilizar la planificación reservada previamente o la holgura del presupuesto.
 - **Protección del riesgo:** reducir la probabilidad y/o consecuencia del riesgo antes de que ocurra.
- **Plan de acción:** estrategia para evitar el riesgo o minimizar sus consecuencias en el peor caso.

Id	R01
Nombre	Requisitos incompletos o poco claros.
Descripción	<p>Los requisitos representan lo que el cliente espera de la aplicación y guían el desarrollo del proyecto. Una elicitación de requisitos incorrecta o insuficiente puede afectar a la planificación del proyecto y la insatisfacción del cliente al no obtener lo que esperaba.</p> <p>En este caso, al estar trabajando con metodología ágil, existe más flexibilidad y no es tan grave el hecho de que los requisitos no estén del todo fijados al comienzo del proyecto ya que pueden aparecer nuevos en alguna iteración y adaptar el proyecto a ellos.</p> <p>El problema aparece cuando no se llega a recoger algún requisito fundamental o se recoge erróneamente y el proyecto sigue avanzando hasta el final sin que éste sea añadido o corregido.</p>
Categoría	De producto.
Contexto	A lo largo de todo el proyecto.
Probabilidad de ocurrencia	20 %
Nivel de impacto	1 a 5, depende de la fase de aparición, aumentando según avance el proyecto.
Evaluación del riesgo	0.2 a 1
Creador	Alumno
Consecuencias	Ralentización del proyecto, dificultades para cumplir los plazos, insatisfacción del cliente.
Plan de acción y seguimiento	
Escenario	<p>En caso de darse este riesgo en la fase inicial del proyecto, se solicitarían o aclararían los requisitos con el cliente antes de continuar, el cual repriorizaría la pila del producto con los requisitos actualizados. Esto provocaría, con seguridad, un cambio en la planificación.</p> <p>Si este riesgo se produce en el curso de un sprint ya iniciado, se pararía el desarrollo de la historia de usuario correspondiente a ese requisito, se consultaría al cliente para que aclare el requisito y repriorice el product backlog si fuera necesario. Si el cliente no pudiera aclarar los requisitos antes de que finalizara el sprint, se continuaría con los siguientes requisitos de la pila del producto.</p>
Indicador	Seguimiento de los requisitos añadidos a la pila del producto, comprobando su completitud y claridad.
Condiciones de disparo	<ul style="list-style-type: none"> ■ Antes de iniciar el proyecto, el cliente no tiene claro que es lo que espera de la aplicación. ■ Según van avanzando las diferentes iteraciones, el cliente indica nuevos requisitos confusos y poco detallados. ■ Al cliente se le ocurren nuevos requisitos una vez el proyecto ha finalizado.
Punto de comprobación	Al inicio de cada sprint.
Estrategia de resolución	Reducción del riesgo.
Plan de acción	Revisar con el cliente los requisitos para aclarar los que sean confusos y añadir aquello que falte. Repriorizar los requisitos y replanificar las tareas si es necesario.

Tabla 4.1: Riesgo 1

Id	R02
Nombre	No encontrar bibliotecas o servicios OCR y de traducción gratuitos, de calidad y con posibilidad de uso offline.
Descripción	Una de las restricciones de este proyecto es el uso de bibliotecas de libre distribución o servicios gratuitos para implementar OCR y el traductor. También sería interesante que se pudieran usar estas bibliotecas en modo offline.
Categoría	De proceso (técnico).
Contexto	En la fase inicial del proyecto, cuando se seleccionan las bibliotecas y/o servicios a usar.
Probabilidad de ocurrencia	75 %
Nivel de impacto	4
Evaluación del riesgo	3
Creador	Alumno
Consecuencias	Costes indirectos, baja calidad de los resultados del reconocimiento y traducción, mala experiencia final del usuario.
Plan de acción y seguimiento	
Escenario	Para el caso del OCR, en el caso de no encontrarse ninguna biblioteca gratuita, de calidad y con posibilidad de uso offline, se optaría por la opción de mejor calidad que sea online. Para la traducción, se elegiría una opción online semigratuita, que asegure unas traducciones aceptables de calidad.
Indicador	Revisión de las distintas bibliotecas y servicios a utilizar y comprobación de la calidad de sus resultados.
Condiciones de disparo	<ul style="list-style-type: none"> ■ Se realiza una búsqueda exhaustiva y no parece encontrarse ninguna biblioteca o servicio gratuito. ■ Se realizan pruebas con varios servicios o bibliotecas gratuitas y ninguna parece ser de calidad. ■ Ninguna de las opciones gratuitas y de calidad tienen versión offline.
Punto de comprobación	Cuando se realice la búsqueda, testeo y selección de bibliotecas y/o servicios OCR y de traducción.
Estrategia de resolución	Reducción del riesgo.
Plan de acción	Utilizar servicios semigratuitos de traducción online, con un límite de traducciones gratuitas, pero que al menos aseguren unas traducciones de calidad. Utilizar servicios de OCR online y gratuitos.

Tabla 4.2: Riesgo 2

Id	R03
Nombre	Diseño pobre o erróneo.
Descripción	Es posible que el diseño elaborado sea pobre o erróneo, debido a la falta de experiencia del alumno.
Categoría	De proceso (técnico).
Contexto	A lo largo del proceso de diseño.
Probabilidad de ocurrencia	30 %
Nivel de impacto	1 a 5
Evaluación del riesgo	0.3 a 1.5
Creador	Alumno.
Consecuencias	Un diseño pobre o incorrecto puede originar problemas en la implementación e incluso ralentizar el proyecto al ser necesario una actualización del diseño inicial.
Plan de acción y seguimiento	
Escenario	En el caso de producirse un diseño pobre o erróneo será necesario incluir en la pila del producto una nueva historia para solucionarlo, provocando un retraso en el proyecto.
Indicador	Revisión y seguimiento de los entregables relativos a diseño.
Condiciones de disparo	<ul style="list-style-type: none"> ■ La aplicación no cumple con todas las características planificadas o algunas no funcionan correctamente. ■ Se complica en exceso el desarrollo.
Punto de comprobación	Durante el transcurso del sprint en el que se realice el diseño y en la finalización del mismo.
Estrategia de resolución	Reducción del riesgo.
Plan de acción	<p>Añadir la corrección del diseño como un requisito prioritario en la pila del producto, ya que es una parte crítica en el desarrollo de software.</p> <p>Una vez corregido el diseño, revisarlo exhaustivamente para asegurarse de que sea un diseño completo y correcto.</p>

Tabla 4.3: Riesgo 3

Id	R04
Nombre	Falta de experiencia en el uso de bibliotecas OCR.
Descripción	Pueden aparecer dificultades a la hora de implementar una biblioteca de reconocimiento óptico de caracteres en una aplicación Android. Estas dificultades se deben a la falta de experiencia del equipo con el uso de esta tecnología.
Categoría	De producto.
Contexto	Durante el desarrollo de la aplicación.
Probabilidad de ocurrencia	50 %
Nivel de impacto	2
Evaluación del riesgo	1
Creador	Alumno.
Consecuencias	Ralentización del proyecto. Al ralentizarse la parte de desarrollo, puede afectar al resto de partes del proyecto.
Plan de acción y seguimiento	
Escenario	En el caso de no tener conocimientos suficientes en este tipo de bibliotecas, el equipo invertiría tiempo en formarse y en aprender a utilizarlas.
Indicador	Revisión de los conocimientos adquiridos por los miembros del equipo.
Condiciones de disparo	<ul style="list-style-type: none"> ■ El equipo se encuentra perdido al implementar el reconocimiento óptico de caracteres. ■ El desarrollo no avanza como estaba planificado.
Punto de comprobación	Comprobar los conocimientos del equipo antes de iniciar la parte de reconocimiento de caracteres en el desarrollo.
Estrategia de resolución	Reservar el riesgo.
Plan de acción	Consultar documentación, tutoriales, foros de ayuda, ejemplos sobre bibliotecas OCR.

Tabla 4.4: Riesgo 4

Id	R05
Nombre	Inestabilidad del IDE Android Studio o fallos producidos por cambios de versiones cuando ya ha comenzado el desarrollo.
Descripción	Para implementar OCR en la propia aplicación, es necesario utilizar Android NDK, lo cual influye notablemente en el tiempo de compilación del proyecto Android. A esto hay que sumarle que el uso de este conjunto de herramientas puede provocar que el IDE llegue a bloquearse en algún momento. Por otra parte, tanto Android como Android Studio se encuentran en continuo desarrollo, lo que lleva a Google a lanzar actualizaciones cada cierto tiempo.
Categoría	De proceso (técnico).
Contexto	Durante el desarrollo de la aplicación.
Probabilidad de ocurrencia	10 %
Nivel de impacto	1
Evaluación del riesgo	0.1
Creador	Alumno.
Consecuencias	El desarrollo se puede ralentizar si Android Studio se vuelve inestable. Por otra parte, si se realiza actualización en el IDE, en la versión de Gradle o en el API de Android, parte del código puede dejar de funcionar, clases y métodos utilizados pueden quedarse obsoletos, teniendo que modificar el código y provocando una ralentización en el proyecto.
Plan de acción y seguimiento	
Escenario	Si Android Studio se vuelve inestable, se esperaría un tiempo prudente a su recuperación y en el caso de que tarde demasiado, se reiniciaría. En el caso de que se haya decidido realizar una actualización y algo no funcione correctamente, se puede valorar volver a la versión previa.
Indicadores	Revisar los periodos de actualizaciones tanto de Android Studio como del API de Android. Revisar páginas oficiales y foros especializados.
Condiciones de disparo	No existen.
Punto de comprobación	Durante el transcurso del sprint/sprints en los que se lleve a cabo el desarrollo.
Estrategia de resolución	Proteger el riesgo.
Plan de acción	Revisar lo que ofrecen las nuevas actualizaciones y en el caso de que aporten valor al proyecto o faciliten su desarrollo, proceder a la actualización.

Tabla 4.5: Riesgo 5

Id	R06
Nombre	Documentación escasa o mal elaborada.
Descripción	El éxito de un proyecto software depende significativamente de la documentación elaborada. Es fundamental que todo trabajo realizado en el proyecto, ya sea de análisis, diseño, código, planificación, riesgos o cualquier otro aspecto, quede documentado detalladamente. Una buena documentación sirve como guía para el equipo de desarrollo y es utilizada para realizar un seguimiento y control de las tareas realizadas y a realizar.
Categoría	De proceso (gestión)
Contexto	A lo largo de todo el proyecto.
Probabilidad de ocurrencia	20 %
Nivel de impacto	4
Evaluación del riesgo	0.8
Creador	Alumno.
Consecuencias	Puede dar lugar a una aplicación que no cumple con todas las funcionalidades esperadas o que directamente funciona mal, porque no se ha documentado bien la parte de análisis y diseño. También puede provocar problemas de planificación, retrasos y una gestión de riesgos insuficiente o errónea.
Plan de acción y seguimiento	
Escenario	En el caso de que se produzca este riesgo, sería necesario añadir las partes faltantes o corregir las partes erróneas de la documentación del proyecto.
Indicadores	Revisión y seguimiento de los entregables al final de cada sprint.
Condiciones de disparo	<ul style="list-style-type: none"> ■ Falta algún requisito del cliente por implementar. ■ En la fase de desarrollo no se sabe con claridad qué implementar. ■ Problemas en la planificación, tareas pendientes que ya se han realizado. ■ No se sabe qué hacer cuando se dispara un riesgo.
Punto de comprobación	A lo largo de todo el proyecto.
Estrategia de resolución	Reducción del riesgo.
Plan de acción	Añadir las partes faltantes o corregir las partes erróneas de la documentación del proyecto.

Tabla 4.6: Riesgo 6

Id	R07
Nombre	Pérdida de documentación y/o otros artefactos.
Descripción	Por alguna causa en concreto se pierde parte o el total de la documentación o código, en algunos casos irre recuperable.
Categoría	De proyecto.
Contexto	A lo largo de todo el proyecto.
Probabilidad de ocurrencia	15 %
Nivel de impacto	4
Evaluación del riesgo	0.6
Creador	Alumno.
Consecuencias	Puede dar lugar a un retraso o a una catástrofe a nivel de proyecto si se pierde la mayoría del mismo.
Plan de acción y seguimiento	
Escenario	De producirse, se intentaría recuperar lo máximo posible de la documentación o código borrado.
Indicadores	Comprobar al final de la jornada la documentación o código realizado durante ese día.
Condiciones de disparo	<ul style="list-style-type: none"> ■ Al abrir el código, no aparece lo último que se haya desarrollado. ■ Al abrir algún documento, falta parte del mismo. ■ Algún directorio del proyecto no tiene ningún fichero.
Punto de comprobación	A lo largo de todo el proyecto.
Estrategia de resolución	Reducción del riesgo.
Plan de acción	Recuperar la última copia disponible. En el caso de que no disponer de ninguna copia accesible, utilizar alguna herramienta de recuperación de archivos.

Tabla 4.7: Riesgo 7

Id	R08
Nombre	Desviación excesiva de los puntos estimados a cada historia de usuario.
Descripción	Los puntos de historia de usuario se utilizan en las metodologías ágiles para estimar el esfuerzo de las diferentes tareas a realizar. Aunque estimar puntos es más sencillo que estimar tiempo, en ocasiones se pueden estimar más o menos puntos de lo necesario.
Categoría	De proceso (gestión).
Contexto	A lo largo de todo el proyecto.
Probabilidad de ocurrencia	50 %
Nivel de impacto	4
Evaluación del riesgo	2
Creador	Alumno.
Consecuencias	Pueden producirse demoras e incumplimiento de las fechas establecidas en la planificación.
Plan de acción y seguimiento	
Escenario	De producirse, habría que reestimar los puntos de las historias de usuario afectadas, repriorizar la pila del producto y replanificar si fuera necesario.
Indicadores	Controlar las tareas que se están realizando y si cumplen con los puntos o tiempo planificado.
Condición de disparo	El equipo tiene que trabajar más o menos horas de las inicialmente planificadas.
Punto de comprobación	A lo largo de todo el proyecto.
Estrategia de resolución	Reducción del riesgo.
Plan de acción	Replanificar.

Tabla 4.8: Riesgo 8

Id	R09
Nombre	Fallos en el servidor donde se aloja el API de traducción.
Descripción	Al utilizar un API de traducción alojado en un servidor externo, es posible que se produzcan caídas que deriven en una pérdida del servicio de traducción que utiliza la aplicación.
Categoría	De proceso (técnico).
Contexto	A lo largo de todo el proyecto.
Probabilidad de ocurrencia	5 %
Nivel de impacto	5
Evaluación del riesgo	0.25
Creador	Alumno.
Consecuencias	La aplicación pierde una de sus funcionalidades fundamentales como es la traducción de un texto, generando insatisfacción en el usuario final.
Plan de acción y seguimiento	
Escenario	De producirse, sería necesario contactar con el proveedor del servicio, para que se solucione el problema lo antes posible.
Indicadores	Comprobar el estado del servicio de traducción.
Condición de disparo	El servicio de traducción no responde y no es un problema de la conexión a internet.
Punto de comprobación	A lo largo de todo el proyecto.
Estrategia de resolución	Reducción del riesgo.
Plan de acción	Notificar el fallo al proveedor del servicio.

Tabla 4.9: Riesgo 9

Id	R10
Nombre	Falta de seguimiento permanente de tareas y actividades.
Descripción	En determinados momentos del proyecto, no se lleva a cabo un seguimiento de las tareas realizadas o por hacer, dando lugar a confusiones sobre las funcionalidades implementadas o por implementar.
Categoría	De proceso (gestión).
Contexto	A lo largo de todo el proyecto.
Probabilidad de ocurrencia	25 %
Nivel de impacto	3
Evaluación del riesgo	0.75
Creador	Alumno.
Consecuencias	Desconocimiento sobre qué requisitos del cliente se han implementado y cuales no. Problemas de planificación, lo que puede derivar en retrasos en el proyecto.
Plan de acción y seguimiento	
Escenario	De producirse, sería necesario revisar la aplicación y comprobar qué tareas se han llevado a cabo y cuales están pendientes de realizar.
Indicadores	Comprobar al final de la jornada que se han documentado todas las tareas realizadas y las tareas que quedan por hacer.
Condición de disparo	Falta documentar tareas ya realizadas, tareas por hacer o aparecen como terminadas tareas aún no realizadas.
Punto de comprobación	A lo largo de todo el proyecto.
Estrategia de resolución	Protección del riesgo.
Plan de acción	Realizar comprobaciones rutinarias de todas las tareas desarrolladas por cada miembro del equipo en una jornada. La reunión diaria o <i>daily</i> puede ser de gran ayuda para revisar qué se ha hecho el día anterior y comprobar si se ha documentado correctamente.

Tabla 4.10: Riesgo 10

Id	R11
Nombre	Equipo de trabajo desmotivado.
Descripción	Es muy frecuente, sobretodo en épocas de sobrecarga del trabajo, que el equipo se encuentre desmotivado. Es importante mantener una motivación de todos los miembros del equipo, lo cual no significa agobiarles con interminables jornadas de trabajo.
Categoría	De proyecto.
Contexto	A lo largo de todo el proyecto.
Probabilidad de ocurrencia	5 %
Nivel de impacto	3
Evaluación del riesgo	0.15
Creador	Alumno.
Consecuencias	La desmotivación de los miembros del equipo puede suponer una bajada en su rendimiento, provocando no solo un aumento en el tiempo de desarrollo, sino un descenso significativo de la calidad del producto resultante.
Plan de acción y seguimiento	
Escenario	De producirse, sería necesario recuperar cuanto antes la motivación del equipo.
Indicadores	Comprobar el rendimiento de los miembros del equipo y la calidad de su trabajo.
Condición de disparo	Los miembros del equipo tardan más tiempo de lo habitual en desarrollar partes de la aplicación o no lo desarrollan con la calidad planificada.
Punto de comprobación	A lo largo de todo el proyecto.
Estrategia de resolución	Protección del riesgo.
Plan de acción	Dividir las tareas complejas en subtareas, establecer periodos de descanso, proponer cambios de actividad cuando exista saturación.

Tabla 4.11: Riesgo 11

4.2.2. Control de riesgos

En la tabla 4.12 se detallan los riesgos que se han producido finalmente.

Riesgos	Sprint	Detalle
R04 - Falta de experiencia en el uso de bibliotecas OCR	Sprint 1	Implementar la biblioteca de reconocimiento Tesseract no ha sido sencillo y ha sido necesario consultar documentación, tutoriales, foros especializados y ejemplos sobre el uso de la biblioteca.
R07 - Pérdida de documentación y/o otros artefactos.	Sprint 2	En este sprint se perdió parte de la memoria debido a un error del editor LaTeX. Fue necesario recuperar la última copia disponible guardada en Dropbox.
R08 - Desviación excesiva de los puntos estimados a cada historia de usuario	Sprint 3	Se produjo una desviación a la hora de estimar los puntos para la realización del diseño de la arquitectura del sistema y la descripción de los patrones de diseño utilizados. Fue necesario aumentar un día la duración de este sprint y replanificar.
R05 - Inestabilidad del IDE Android Studio o fallos producidos por cambios de versiones cuando ya ha comenzado el desarrollo.	Sprint 4	Se produjeron inestabilidades en Android Studio cuando se comenzó a aplicar el patrón MVP en el código. Se descubrió que el problema procedía del comando ndk-build, que tuvo que eliminarse para compilar el proyecto, ya que no era necesario al estar incluidas en el proyecto las bibliotecas ya compiladas. Por otro lado, se actualizó a la versión r11 de Android NDK y se detectaron problemas, por lo que hubo que volver a la versión r10e.

Tabla 4.12: Riesgos producidos

Para ver información más detallada de los riesgos producidos en cada sprint, consultar el Plan de Seguimiento en el apéndice C.

Capítulo 5

Requisitos

5.1. Los requisitos en desarrollo ágil

La especificación de los requisitos del sistema varía en función del tipo de gestión del proyecto, tradicional o ágil. En ambos casos, es responsabilidad del cliente definir el problema y las funcionalidades que debe aportar la solución.

En la tabla 5.1 se describen las diferencias entre la especificación de los requisitos de un proyecto tradicional y uno ágil [8].

	Proyecto tradicional	Proyecto ágil
¿Dónde se especifican los requisitos?	En documentos formales.	En la pila del producto o lista de historias de usuario.
¿Cómo y cuándo se especifican los requisitos?	De forma completa y cerrada al inicio del proyecto.	Una pila del producto evoluciona durante el desarrollo
¿Quién especifica los requisitos?	Una persona o equipo o especializado en ingeniería de requisitos a través del proceso de elicitación con el cliente.	La visión del cliente la conoce todo el equipo y la pila del producto se realiza de forma continua con los aportes de todos.

Tabla 5.1: Comparativa de especificación de requisitos entre un proyecto tradicional y otro ágil

En la siguiente imagen se representan los requisitos completos de los proyectos tradicionales en comparación con los requisitos evolutivos de los proyectos ágiles [8].

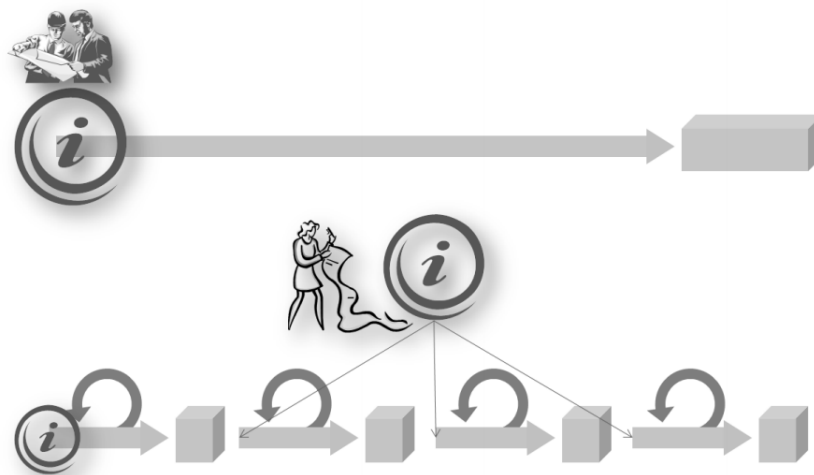


Figura 5.1: Requisitos completos / evolutivos

Teniendo en cuenta esta breve explicación, en los dos siguientes apartados se listan todos los requisitos especificados a lo largo del desarrollo del proyecto. La lista de requisitos NO se ha obtenido de forma completa al iniciar el proyecto, sino que se han ido añadiendo nuevos requisitos durante su desarrollo.

5.2. Requisitos funcionales

ID	Nombre	Descripción
RF01	Reconocimiento de los caracteres contenidos en una imagen.	El sistema deberá permitir al Usuario obtener el texto reconocido a partir de una imagen.
RF02	Selección de la imagen para el reconocimiento.	El sistema deberá permitir al Usuario realizar una foto con la cámara del dispositivo o seleccionar una imagen desde la galería con la que realizar el reconocimiento.
RF03	Selección de la parte de la imagen a reconocer.	El sistema deberá permitir al Usuario seleccionar la parte de la imagen que desee reconocer.
RF04	Traducción de los caracteres reconocidos.	El sistema deberá permitir al Usuario obtener una traducción del texto reconocido al idioma seleccionado.
RF05	Selección del idioma de partida o reconocimiento.	El sistema deberá permitir al Usuario seleccionar el idioma que desee reconocer entre 39 idiomas de reconocimiento diferentes.
RF06	Selección del idioma de llegada, al que traducir el texto reconocido.	El sistema deberá permitir al Usuario seleccionar el idioma al que desee traducir entre 39 idiomas diferentes.
RF07	Guardado de traducciones.	El sistema deberá permitir al Usuario guardar las traducciones que realice.
RF08	Visualización de traducciones.	El sistema deberá permitir al Usuario consultar las traducciones realizadas.
RF09	Detalle de traducciones.	El sistema deberá permitir al Usuario consultar el detalle de las traducciones realizadas, mostrando información sobre el idioma de partida y destino, el texto reconocido y su traducción, así como la fecha y localización.
RF10	Borrado de traducciones.	El sistema deberá permitir al Usuario borrar una traducción.
RF11	Búsqueda de traducciones.	El sistema deberá permitir al Usuario realizar búsquedas de traducciones en función del texto traducido, la fecha o localización.
RF12	Filtro de traducciones por fecha y localización.	El sistema deberá permitir al Usuario filtrar las traducciones realizadas por fecha y localización.
RF13	Localización de traducciones.	El sistema deberá permitir al Usuario activar o desactivar la localización de una traducción antes de guardarla.

Tabla 5.2: Requisitos funcionales del sistema

5.3. Requisitos no funcionales

ID	Nombre	Descripción
RNF01	Facilidad de instalación	La aplicación debe ser fácilmente instalable por cualquier usuario a partir de su apk.
RNF02	Solicitud de permisos	La aplicación debe solicitar, en versiones superiores al API 23 de Android, los permisos necesarios para realizar las tareas correspondientes.
RNF03	Tiempos de respuesta	La aplicación debe asegurar unos tiempos de respuesta no superiores a los 30 segundos en el reconocimiento del texto de las imágenes y 10 segundos en la traducción. No obstante, los tiempos de reconocimiento son muy variables e impredecibles.
RNF04	Sistema de gestión de bases de datos relacional SQLite	La aplicación debe utilizar para la persistencia este tipo de sistema de gestión de bases de datos.
RNF05	Versiones compatibles	La aplicación debe ser compatible con cualquier dispositivo Android con versión comprendida entre 4.1.x y 6.0 (94.7 % de todos los dispositivos Android)
RNF06	Comunicación con Microsoft Translator	La aplicación debe comunicarse con el API de traducción de Microsoft para enviar el texto reconocido y obtener la traducción al idioma correspondiente.
RNF07	Internacionalización	La aplicación debe estar traducida a inglés y castellano como mínimo.
RNF08	Diseño de la interfaz de usuario	La interfaz del sistema seguirá la normativa de diseño Material Design impulsada por Google, con un aspecto claro, intuitivo y sencillo de utilizar

Tabla 5.3: Requisitos no funcionales del sistema

5.3.1. Requisitos de información

ID	Nombre	Descripción
RIN01	Contenido de una traducción	Id, imagen, idioma de partida, texto reconocido, idioma de llegada, texto traducido, fecha de creación y localización.
RIN02	Contenido de un idioma	Código del idioma, nombre y si está descargado o no.

Tabla 5.4: Requisitos no funcionales de información del sistema

Capítulo 6

Análisis

En este apartado se describen los actores del sistema y se modelan y especifican los casos de uso de *Asistev*. También se incluye el modelo de dominio, la descripción de sus clases y los diagramas de actividad.

El objetivo de toda esta información es tener una descripción completa de la funcionalidad requerida por el cliente.

Nota importante: al estar desarrollando el proyecto con una metodología ágil como es Scrum, los requisitos del apartado anterior y los casos de uso se escribirán en forma de historias de usuario de alto nivel. Para más información, consultar el Plan de Seguimiento del apéndice C.

6.1. Modelo de Casos de Uso.

6.1.1. Actores

- Usuario: es el único actor del sistema y representa a la persona que utilizará la aplicación.

6.1.2. Diagrama de casos de uso.

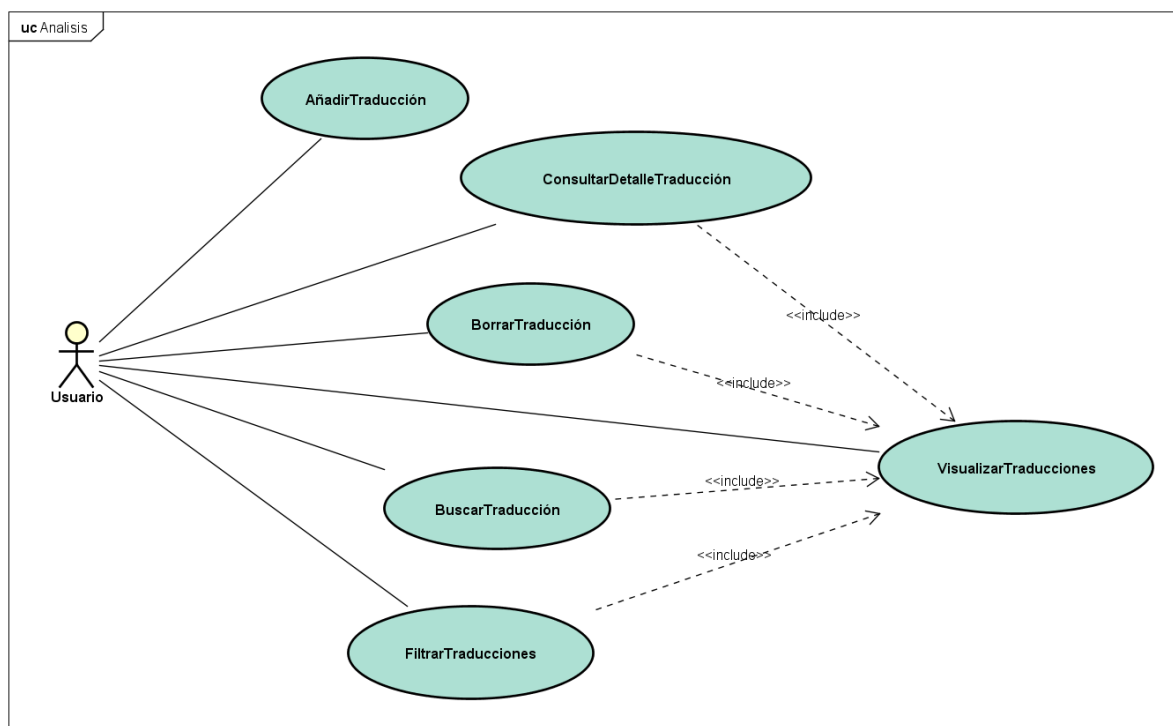


Figura 6.1: Diagrama de casos de uso

6.1.3. Especificación de Casos de Uso.

CU01	AñadirTraducción
Descripción	Un Usuario desea añadir una nueva traducción del texto contenido en una imagen.
Actor	Usuario.
Precondición	El Usuario ha iniciado la aplicación.
Secuencia normal	<ol style="list-style-type: none"> 1. El caso de uso se inicia cuando el Usuario selecciona añadir una nueva traducción. 2. El sistema muestra la opción de obtener la imagen desde la cámara o desde la galería. 3. El Usuario selecciona obtener la imagen desde la cámara. 4. El sistema abre la aplicación de la cámara. 5. El Usuario realiza una foto. 6. El sistema muestra la imagen seleccionada, una lista de idiomas de reconocimiento y un rectángulo ajustable para seleccionar parte de la imagen. 7. El Usuario selecciona el idioma a reconocer, la parte de la imagen que contenga el texto que desea traducir y acepta. 8. El sistema comprueba que el idioma a reconocer está descargado en el dispositivo. El sistema realiza el reconocimiento y muestra el texto reconocido, una lista de idiomas a los que traducir, una opción para realizar la traducción y otra para incluir la localización al guardar la traducción. 9. El Usuario selecciona el idioma de llegada e indica realizar la traducción. 10. El sistema realiza la traducción al idioma seleccionado y la muestra. 11. El Usuario selecciona guardar la traducción. 12. El sistema comprueba que se ha realizado la traducción y la guarda. 13. El caso de uso finaliza.
Postcondición	La traducción ha sido guardada en el sistema.
Flujo alternativo	<p>Si el Usuario cancela en cualquier momento el caso de uso queda sin efecto.</p> <p>3.1. Si el Usuario selecciona obtener imagen de la galería, el sistema abre la galería y cuando el usuario selecciona la imagen deseada, el caso de uso continúa en el paso 6.</p> <p>7.1. Si el Usuario decide descargar un idioma de reconocimiento, el sistema comienza con la descarga y cuando finalice avisa al usuario, continuando el caso de uso en el paso 6.</p> <p>11.1. Si el Usuario activa la localización, el sistema le pide los permisos en el caso de que no lo haya hecho previamente, comprueba los permisos, comprueba si está activada la localización, la inicia y le avisa cuando se haya encontrado. El caso de uso continua en el paso 10.</p>
Excepciones	<p>8.1. Si el sistema detecta que el idioma de reconocimiento seleccionado no está descargado en el dispositivo, se avisa al usuario y el caso de uso continúa en el paso 6.</p> <p>11.1.1. Si el sistema detecta que no está activada la localización, avisa al usuario y el caso de uso continúa en el paso 10.</p>
Frecuencia	Alta.

Tabla 6.1: Descripción del CU01

CU02	VisualizarTraducciones
Descripción	Un Usuario desea ver las traducciones que ha realizado.
Actor	Usuario.
Precondición	<ol style="list-style-type: none"> 1. El Usuario ha iniciado la aplicación. 2. El Usuario ha guardado alguna traducción.
Secuencia normal	<ol style="list-style-type: none"> 1. El caso de uso se inicia cuando el Usuario está en la vista principal de la aplicación. 2. El sistema comprueba si hay algún filtro activado y muestra una lista de traducciones en función de esos filtros, incluyendo la siguiente información de cada una de ellas: <ul style="list-style-type: none"> ■ Miniatura de la imagen que contenía el texto a reconocer. ■ Resumen del texto traducido. ■ Fecha y hora en la que se ha realizado la traducción. ■ Localización (en el caso de que el usuario la haya activado). 3. El caso de uso finaliza.
Postcondición	Ninguna.
Flujo alternativo	2.1. Si el sistema no detecta ningún filtro seleccionado, muestra todas las traducciones y el caso de uso finaliza en el paso 3.
Excepciones	No existen.
Frecuencia	Muy alta.

Tabla 6.2: Descripción del CU02

CU03	ConsultarDetalleTraducción.
Descripción	Un Usuario desea consultar el detalle de una traducción.
Actor	Usuario.
Precondición	<ol style="list-style-type: none"> 1. El Usuario ha iniciado la aplicación. 2. El Usuario ha guardado alguna traducción.
Secuencia normal	<ol style="list-style-type: none"> 1. Se ejecuta el caso de uso VisualizarTraducciones. 2. El Usuario selecciona una de las traducciones mostradas. 3. El sistema muestra la siguiente información de la traducción: <ul style="list-style-type: none"> ■ Miniatura de la imagen que contenía el texto a reconocer. ■ Fecha y localización. ■ Idioma de partida. ■ Texto original completo. ■ Idioma de llegada. ■ Texto traducido completo. 4. El caso de uso finaliza.
Postcondición	Ninguna.
Flujo alternativo	No existe.
Excepciones	No existen.
Frecuencia	Media.

Tabla 6.3: Descripción del CU03

CU04	BorrarTraducción
Descripción	Un Usuario desea borrar una traducción.
Actor	Usuario.
Precondición	1. El Usuario ha iniciado la aplicación. 2. El Usuario ha guardado alguna traducción.
Secuencia normal	1. Se ejecuta el caso de uso VisualizarTraducciones. 2. El Usuario selecciona una de las traducciones para borrarla. 3. El sistema borra la traducción y muestra una opción para deshacer el borrado. 4. El caso de uso finaliza.
Postcondición	La traducción ha sido eliminada del sistema.
Flujo alternativo	3.1. Si el Usuario pulsa la opción de deshacer borrado, se recupera la traducción borrada y el caso de uso finaliza en el paso 4.
Excepciones	No existen.
Frecuencia	Media.

Tabla 6.4: Descripción del CU04

CU05	BuscarTraducción
Descripción	Un Usuario desea buscar una traducción.
Actor	Usuario.
Precondición	1. El Usuario ha iniciado la aplicación. 2. El Usuario ha guardado alguna traducción.
Secuencia normal	1. Se ejecuta el caso de uso VisualizarTraducciones. 2. El Usuario selecciona la opción de buscar traducción. 3. El sistema muestra un espacio donde escribir la consulta y el teclado del dispositivo. 4. El Usuario comienza a escribir la consulta. 5. El sistema comprueba la consulta caracter a caracter y muestra las traducciones que coincidan en tiempo real. 6. El caso de uso finaliza.
Postcondición	Las traducciones han sido filtradas en función de la consulta realizada.
Flujo alternativo	4.1. Si el usuario cancela, el caso de uso finaliza en el paso 6.
Excepciones	No existen.
Frecuencia	Media.

Tabla 6.5: Descripción del CU05

CU06	FiltrarTraducciones
Descripción	Un Usuario desea filtrar las traducciones en función de la fecha y localización.
Actor	Usuario.
Precondición	1. El Usuario ha iniciado la aplicación. 2. El Usuario ha guardado alguna traducción.
Secuencia normal	1. Se ejecuta el caso de uso VisualizarTraducciones. 2. El Usuario selecciona la opción de filtrar traducciones. 3. El sistema muestra dos listas donde seleccionar la fecha y/o localización deseada. 4. El Usuario selecciona una fecha y/o una localización y acepta. 5. El sistema comprueba los filtros seleccionados y muestra las traducciones que coincidan. 6. El caso de uso finaliza.
Postcondición	Las traducciones han sido filtradas en función de la fecha y localización seleccionadas.
Flujo alternativo	4.1. Si el usuario cancela, el sistema no tiene en cuenta la fecha y localización seleccionada y el caso de uso finaliza en el paso 6.
Excepciones	No existen.
Frecuencia	Media.

Tabla 6.6: Descripción del CU06

6.2. Realización de los Casos de Uso en Análisis

6.2.1. Modelo de Dominio

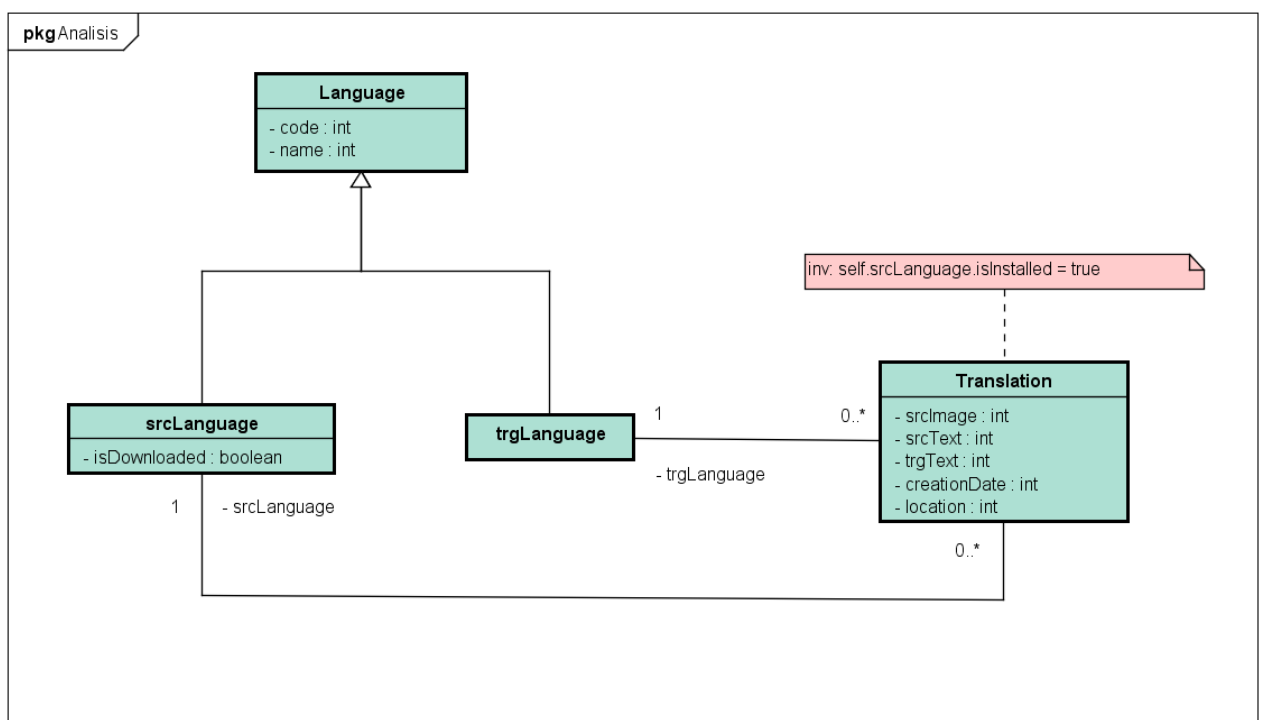


Figura 6.2: Modelo de dominio

6.2.2. Descripción de las clases del modelo de dominio

6.2.2.1. Translation

- **Descripción:** clase que modela una traducción.
- **Responsabilidades:** referenciar una traducción de la aplicación.
- **Atributos:**
 - *srcImage*: imagen que contiene el texto a reconocer.
 - *srcText*: texto reconocido.
 - *trgText*: texto traducido.
 - *creationDate*: fecha en la que se ha realizado la traducción.
 - *location*: localización en la que se ha realizado la traducción.

6.2.2.2. Language

- **Descripción:** clase que modela un idioma.
- **Responsabilidades:** referenciar un idioma en la aplicación.
- **Atributos:**
 - *code*: código del idioma en el estándar ISO 639-3.
 - *name*: nombre del idioma. Este nombre se traduce en función del idioma del dispositivo y se guarda en la base de datos. De esta forma se muestra correctamente al usuario en función del idioma del dispositivo.

6.2.2.3. srcLanguage

- **Descripción:** especialización de la clase Language. Modela el idioma de partida, en el que está el texto a reconocer.
- **Responsabilidades:** referenciar un idioma de partida en la aplicación. Utilizado para iniciar el reconocimiento de caracteres y como idioma de partida en la traducción.
- **Atributos:**
 - *isDownloaded*: indica si el idioma está descargado o no, para poder utilizarse en el reconocimiento de caracteres.

6.2.2.4. trgLanguage

- **Descripción:** especialización de la clase Language. Modela el idioma de llegada, al que será traducido el texto reconocido.
- **Responsabilidades:** referenciar un idioma de llegada en la aplicación. Utilizado para realizar la traducción al idioma seleccionado por el usuario.

6.2.3. Diagramas de actividad

6.2.3.1. CU01: Añadir traducción

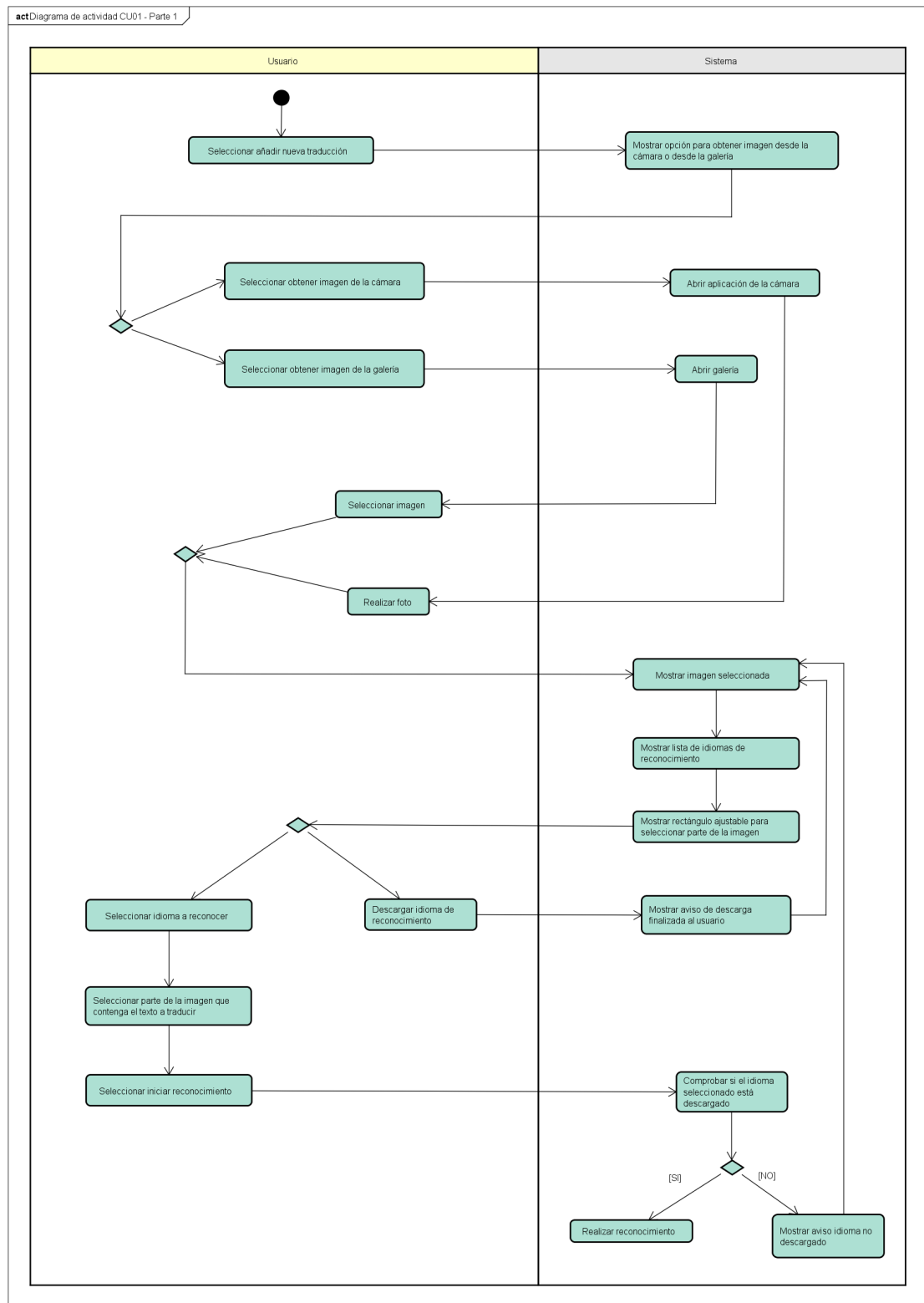


Figura 6.3: Diagrama de actividad del CU01. Parte 1

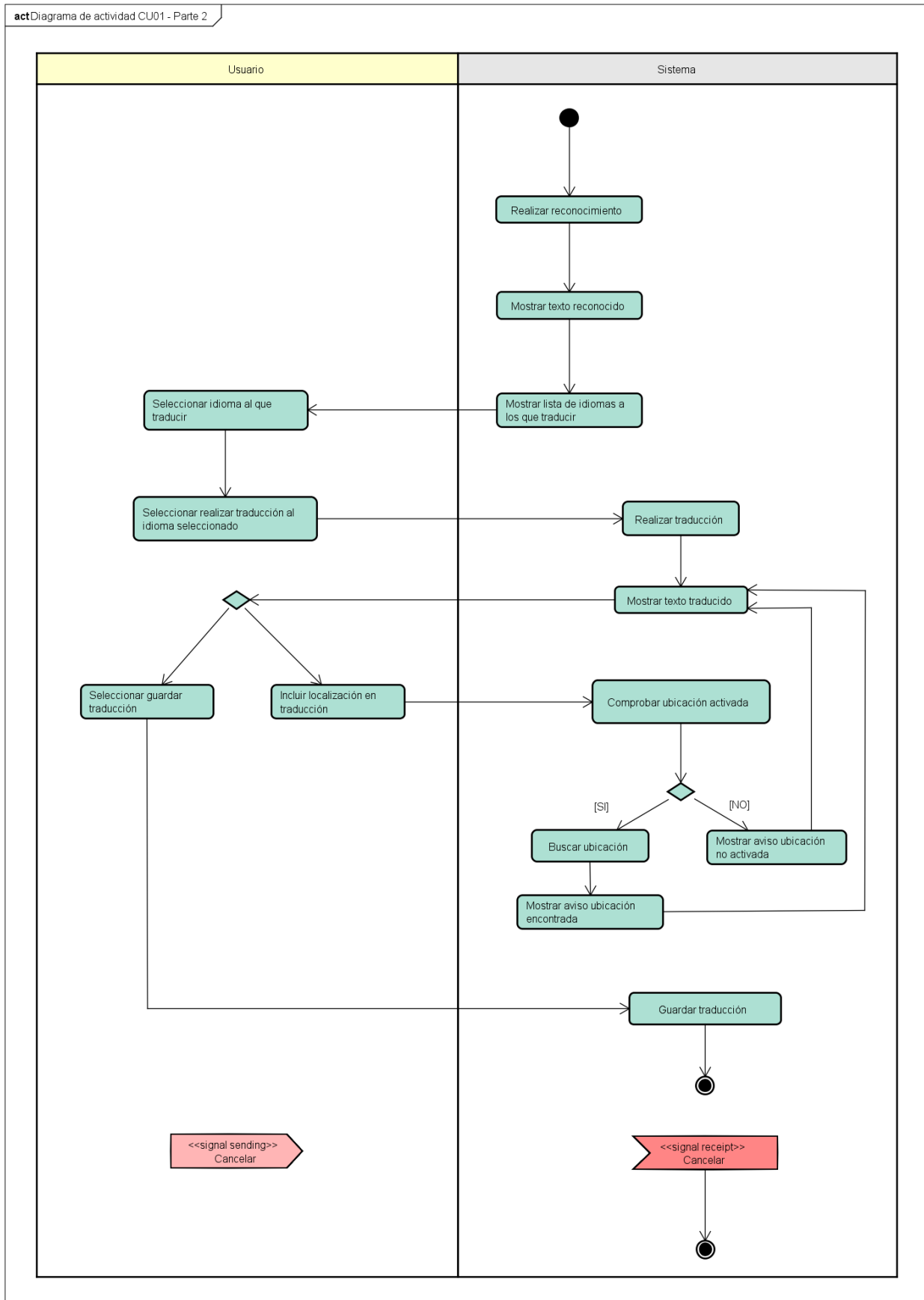


Figura 6.4: Diagrama de actividad del CU01. Parte 2

6.2.3.2. CU02: Visualizar traducciones

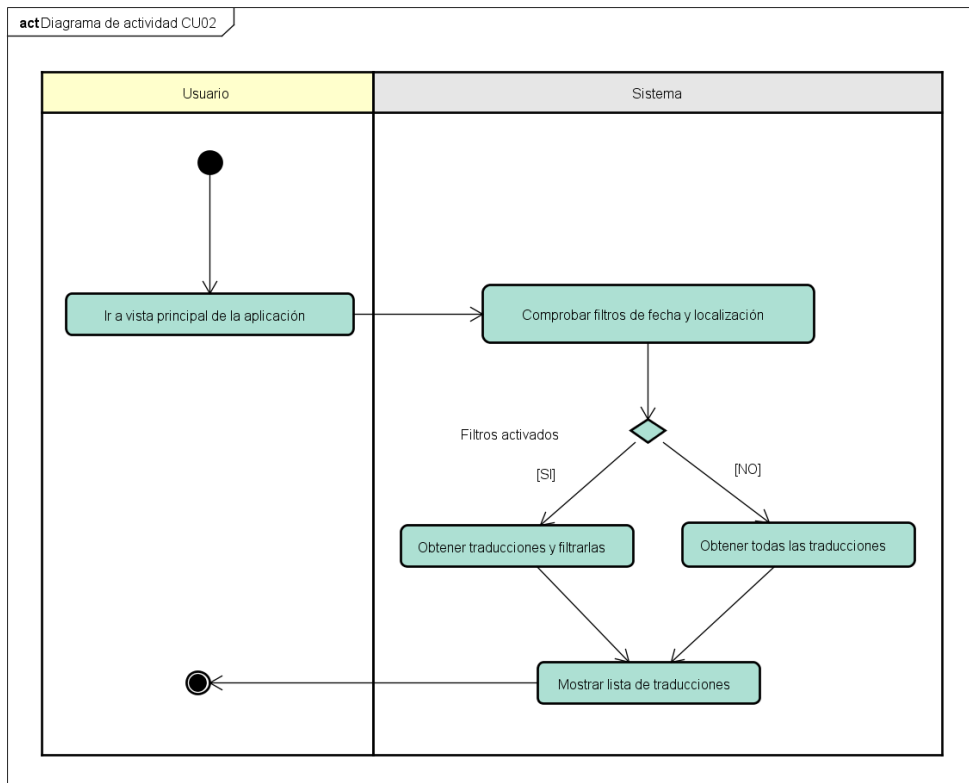


Figura 6.5: Diagrama de actividad del CU02

6.2.3.3. CU03: Consultar detalle de traducción

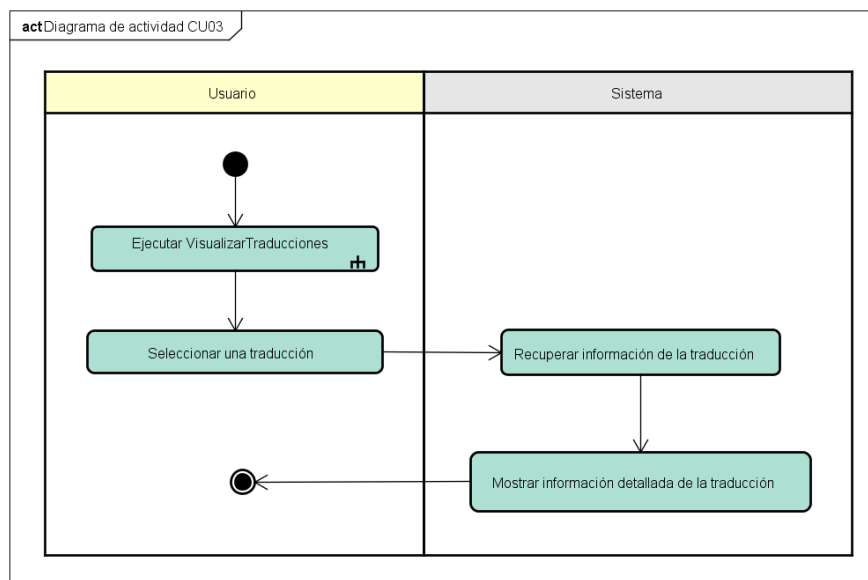


Figura 6.6: Diagrama de actividad del CU03

6.2.3.4. CU04: Borrar traducción

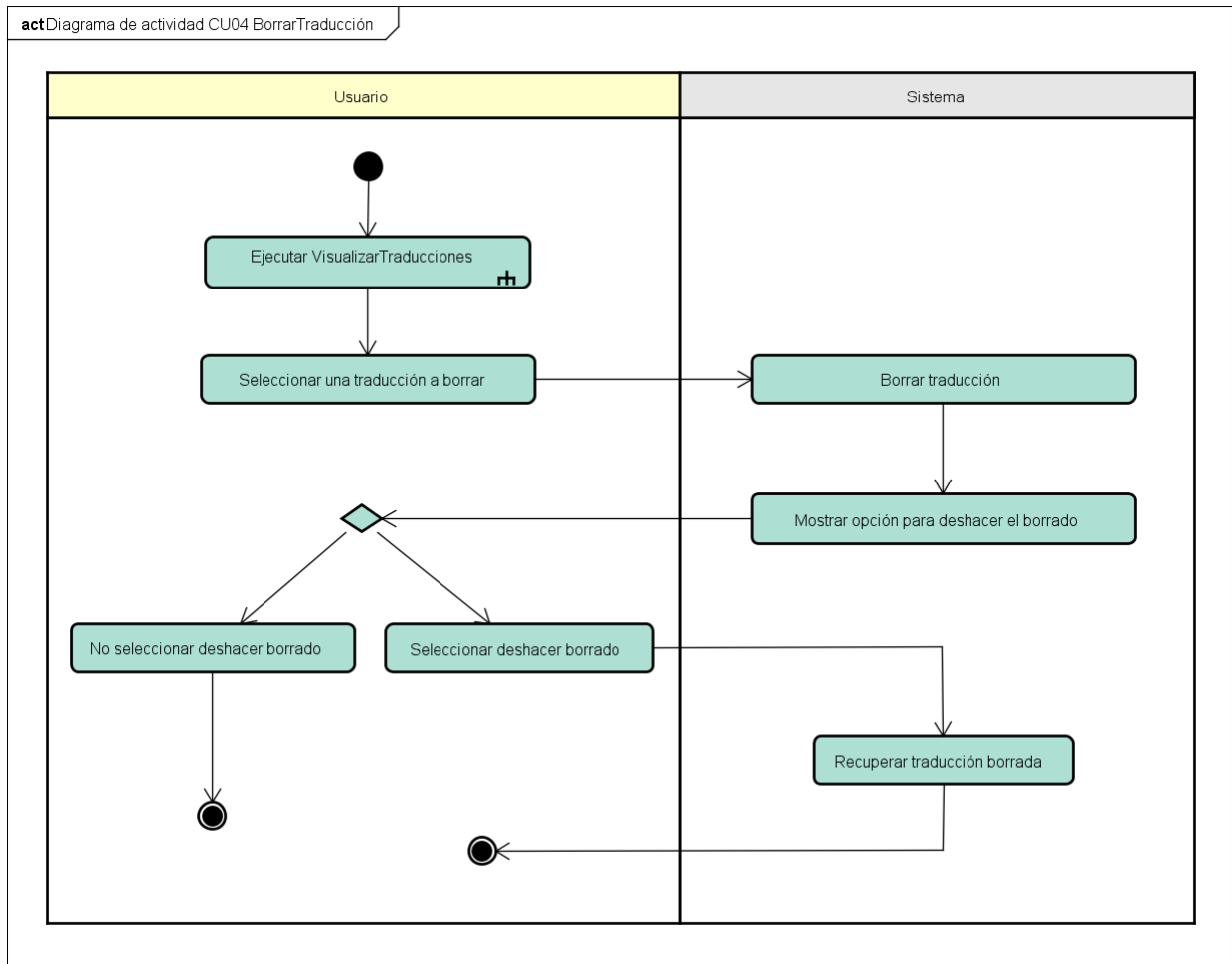


Figura 6.7: Diagrama de actividad del CU04

6.2.3.5. CU05: Buscar traducción

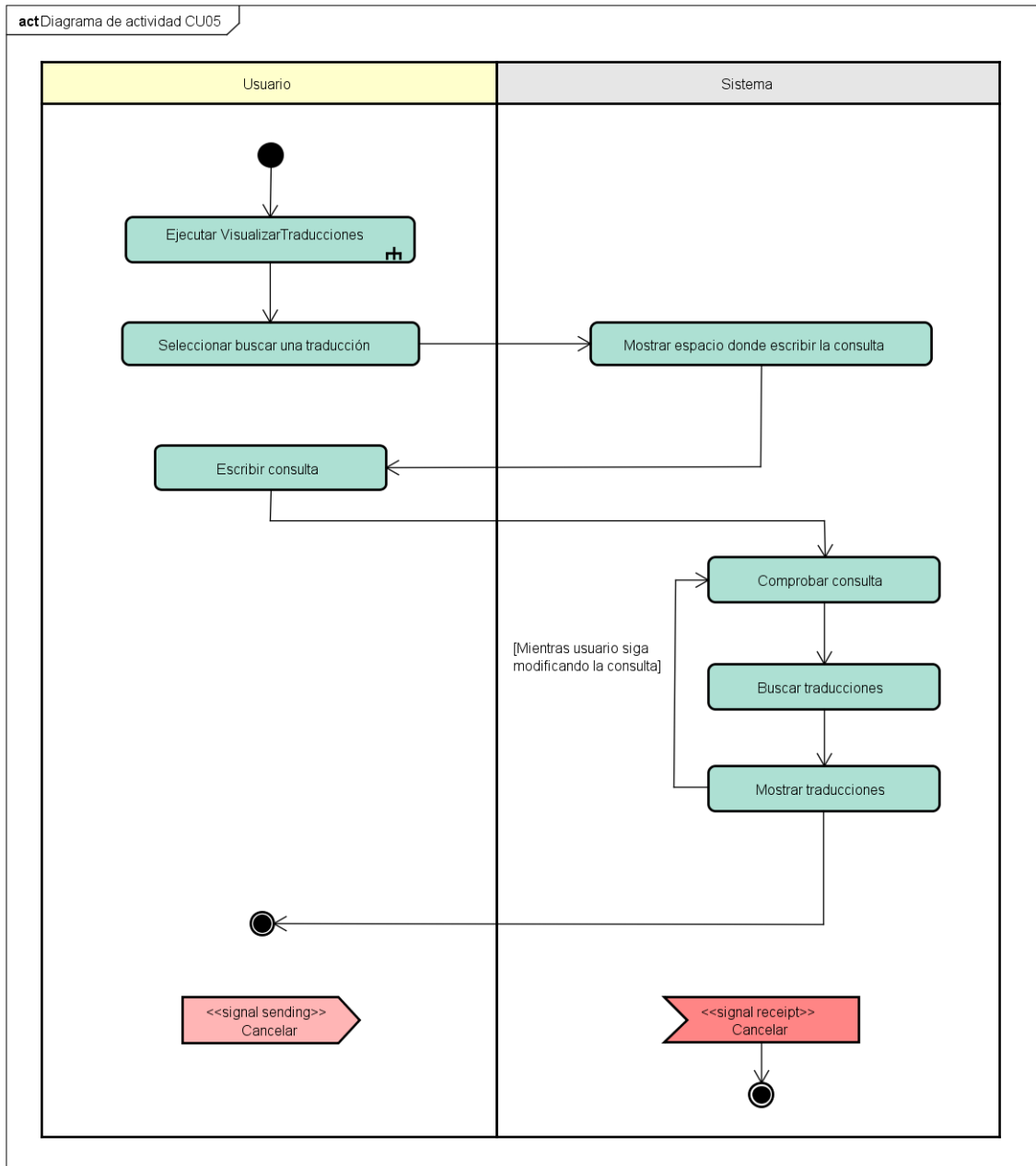


Figura 6.8: Diagrama de actividad del CU05

6.2.3.6. CU06: Filtrar traducciones

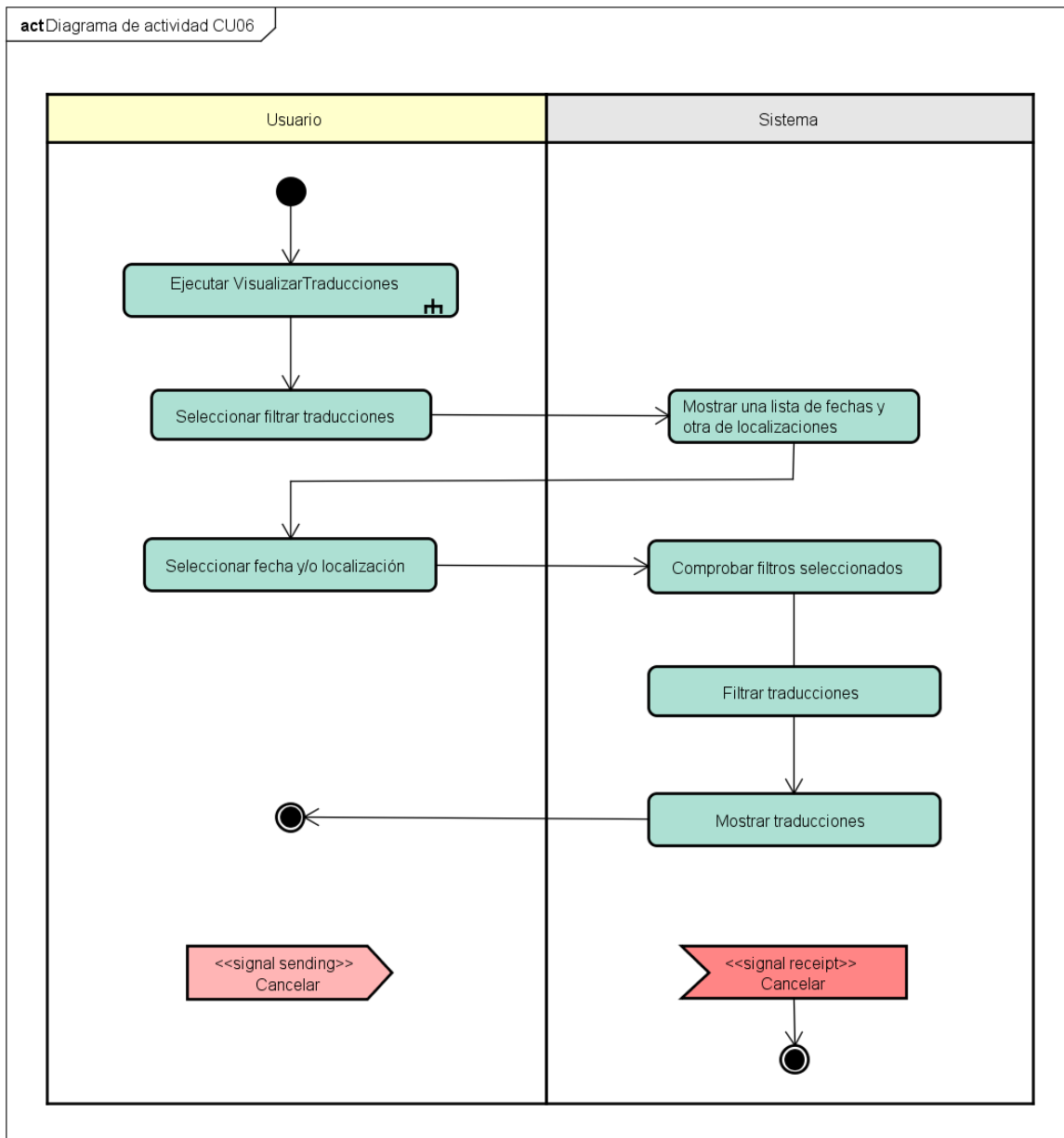


Figura 6.9: Diagrama de actividad del CU06

Capítulo 7

Diseño

7.1. Diseño de la arquitectura

7.1.1. Patrón arquitectónico MVP

El patrón Modelo-Vista-Presentador [17] o MVP es un patrón derivado del conocido MVC o Modelo-Vista-Controlador y actualmente está cobrando una gran importancia en el desarrollo de aplicaciones Android.

En la figura 7.1 se puede ver las diferencias entre MVC y MVP. La diferencia fundamental entre estos dos patrones es la dependencia entre el modelo y la vista, presente en MVC pero no en MVP.

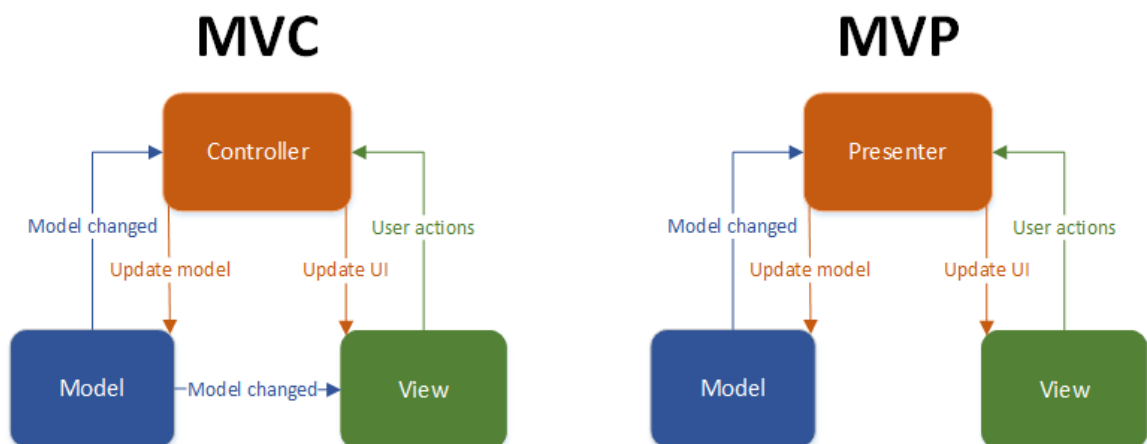


Figura 7.1: Diferencias entre MVC y MVP

El patrón MVP permite separar la capa de presentación de la lógica de la misma, de tal forma que todo lo relacionado con cómo funciona la interfaz queda separado de cómo se representarlo en pantalla. Por este motivo es recomendable su uso en Android ya que en el desarrollo para esta plataforma es muy común tener actividades en las que se mezcla interfaz, lógica, acceso a datos y otros, lo cual dificulta su entendimiento.

Este patrón consiste en tres componentes:

- **Vista:** habitualmente implementada por una Activity o un Fragment. Es la encargada de mostrar los datos en la interfaz de usuario. Contiene una referencia al presentador, invocándolo cada vez que se realiza una acción sobre la interfaz.
- **Presentador:** encargado de intermediar entre la vista y el modelo y responsable de la lógica operacional de negocio. Decide qué ocurre cuando se interactúa con la vista. Además, recupera los datos del modelo y se los devuelve a la vista formateados.
- **Modelo:** contiene la lógica de negocio, el dominio de la aplicación.

En la figura 7.2 se puede ver la interacción entre estos componentes.

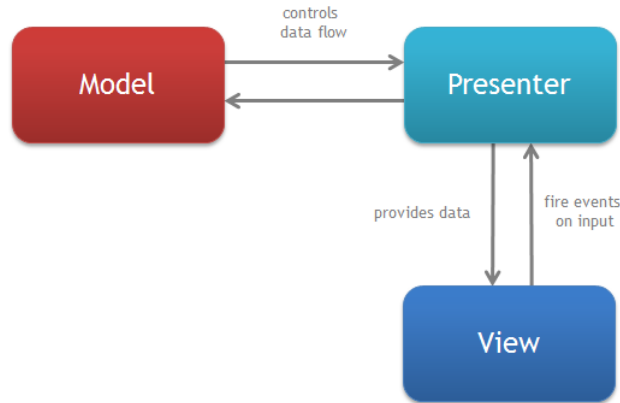


Figura 7.2: Patrón MVP

7.1.2. Patrones de diseño utilizados

7.1.2.1. Patrón Agente o Broker

El patrón Agente o *Broker* [18] tiene como objetivo separar los objetos del dominio de la implementación del almacenamiento de datos.

Un Agente o Broker es una clase encargada de dar acceso a la base de datos a los objetos de la capa de dominio, de manera que se consigue un cierto desacoplamiento de éstos. Esta clase es la única que conoce directamente las operaciones.

Para cada clase del dominio que necesite ser persistente existirá una clase asociada (broker, intermediario) que proporcione los mecanismos para materializar objetos desde la base de datos y desmaterializarlos de nuevo en la base de datos.

En la figura 7.3 se muestra un ejemplo de este patrón, con una clase del dominio y su correspondiente broker.

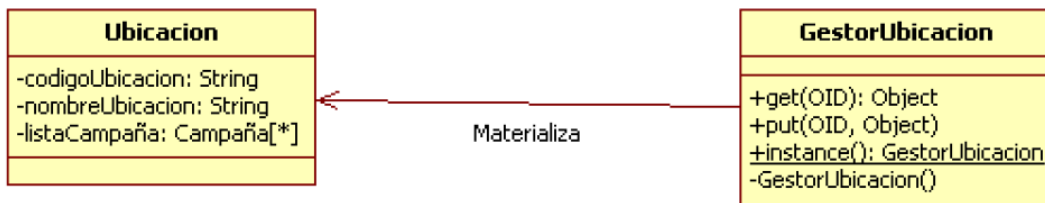


Figura 7.3: Ejemplo de patrón Agente o Broker

GestorUbicacion es responsable del almacenamiento y recuperación de las instancias del objeto Ubicacion.

En nuestro caso, existen dos clases de dominio, Language y Translation, asociadas, cada una de ellas, a su correspondiente Gestor de Persistencia (Agente o Broker).

7.1.2.2. Patrón Singleton

El patrón *Singleton* [19] tiene como objetivo asegurarse de que una clase tiene una sola instancia, y proporcionar un punto de acceso global a ella, evitando la necesidad de utilizar variables globales.

La propia clase que hace uso de este patrón es la encargada de devolver su instancia o crearla en el caso de que no exista.

En la figura 7.4 aparece representado este patrón.

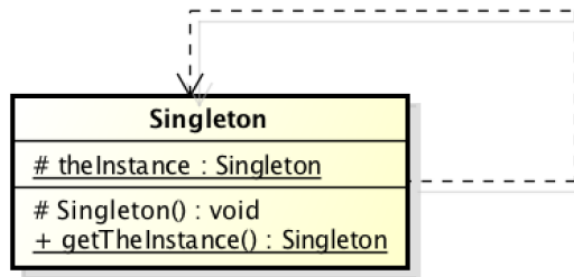


Figura 7.4: Ejemplo de patrón singleton

En nuestro caso, este patrón se está utilizando para obtener una única instancia de acceso a la base de datos. Cuando otra clase necesita acceder a la base de datos, accede a la única instancia de DBManager.

7.1.3. Arquitectura general

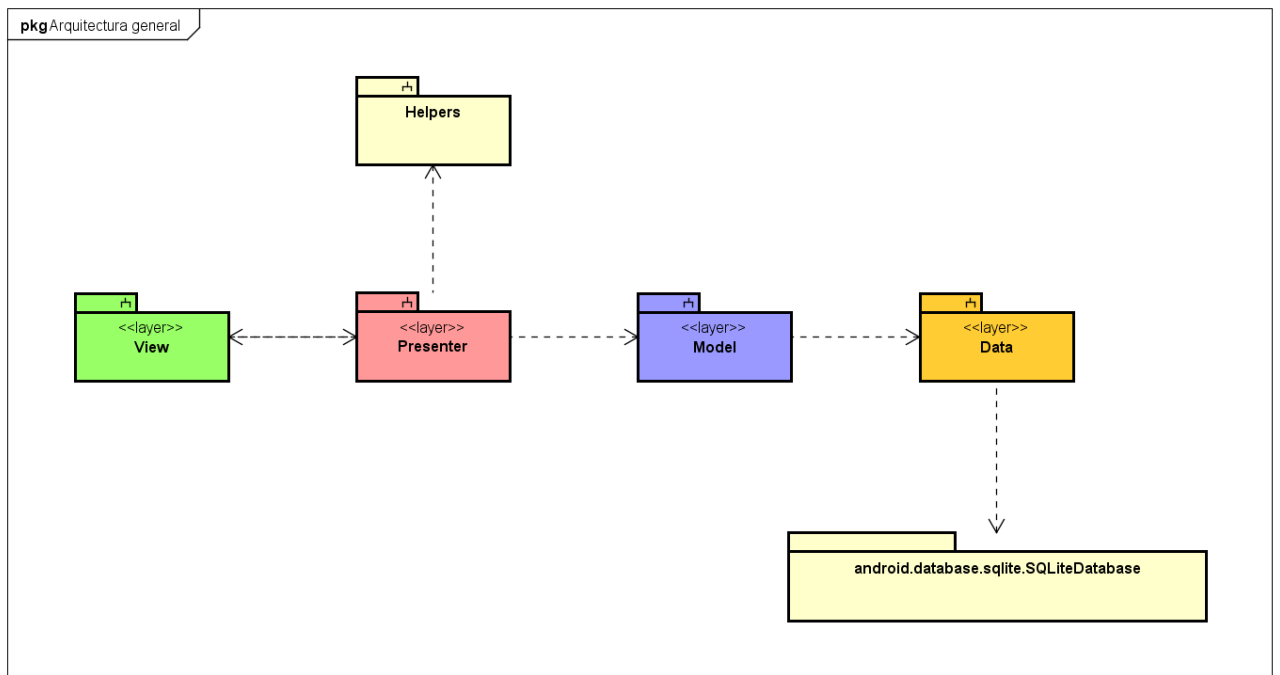


Figura 7.5: Modelo general de la arquitectura

7.1.3.1. Descripción del modelo general de la arquitectura

Como se puede ver en la figura 7.5, se ha establecido una arquitectura de cuatro capas:

- **Vista:** correspondiente al componente *View* de MVP.
- **Presentador:** correspondiente al componente *Presenter* de MVP.
- **Modelo:** correspondiente al componente *Model* de MVP.
- **Datos:** acceso a la base de datos.

En otros casos, la capa View y Presenter podrían haberse agrupado en una única capa de Presentación y la lógica de negocio operacional podría haberse llevado a la capa Modelo.

Al estar trabajando con una aplicación Android, se ha podido observar que haciéndolo de la forma anteriormente citada, el código se complicaba demasiado. Por esta razón, se ha decidido dejar la lógica operacional en los Presentadores y que el Modelo contenga únicamente el dominio de la aplicación.

También se representa un paquete Helpers, que incluye la clase encargada de conectar con la biblioteca nativa y realizar el reconocimiento de caracteres y otras clases que sirven de soporte.

Por último, también se representa la dependencia de la capa de acceso a datos con una base de datos SQLite.

7.1.4. Descomposición modular

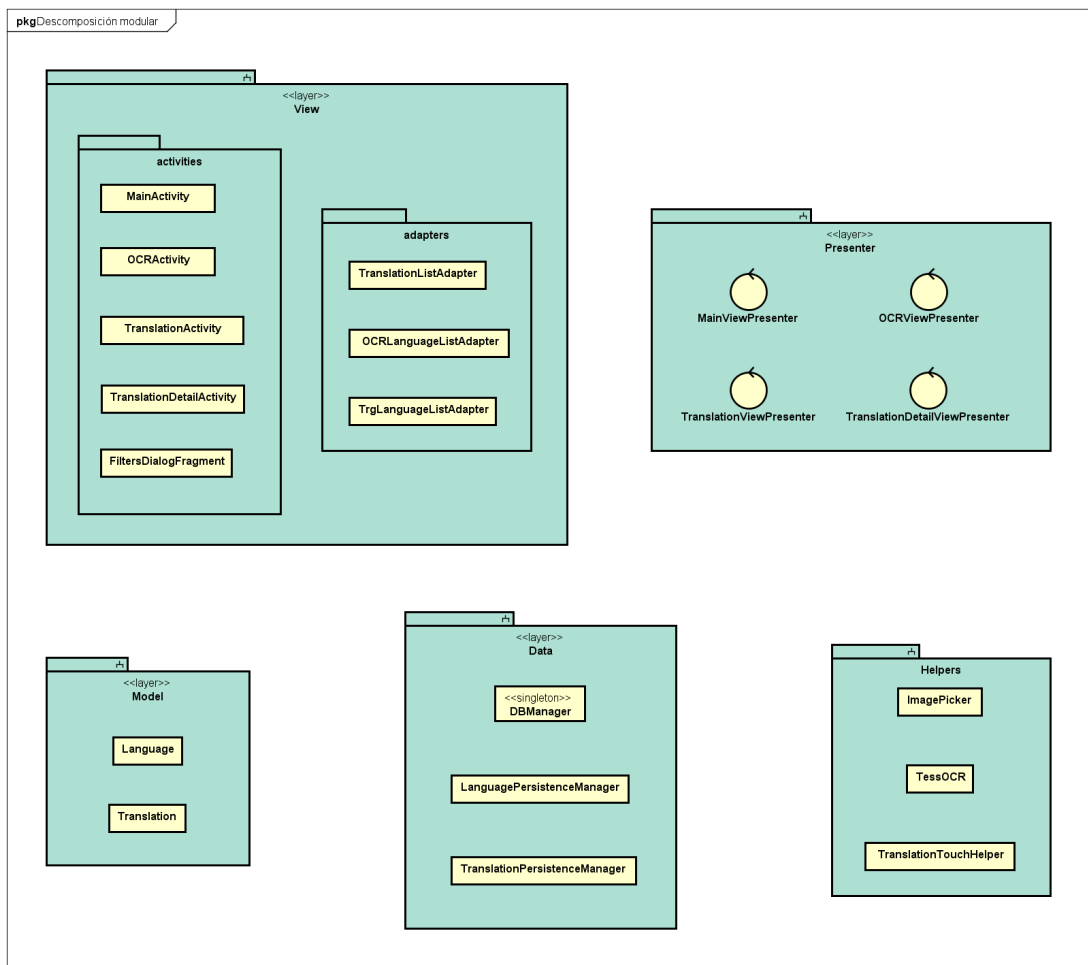


Figura 7.6: Descomposición modular

7.1.4.1. Descripción de la descomposición modular

El objetivo del diagrama de la figura 7.6 es representar la división del sistema en partes diferenciadas, enumerando las clases implicadas en cada una de las partes, sin entrar en detalle ni en las relaciones existentes entre ellas.

En los dos siguientes apartados se detallan estos módulos, mostrando un diagrama con los atributos, operaciones y relaciones entre las clases de cada uno de ellos y otro en el que aparecen las relaciones entre clases de diferentes módulos.

7.1.5. Diseño de los módulos o subsistemas

En este apartado se detallan cada uno de los módulos identificados en la figura 7.6, con las relaciones existentes entre las clases que los forman.

7.1.5.1. Diseño detallado del módulo View

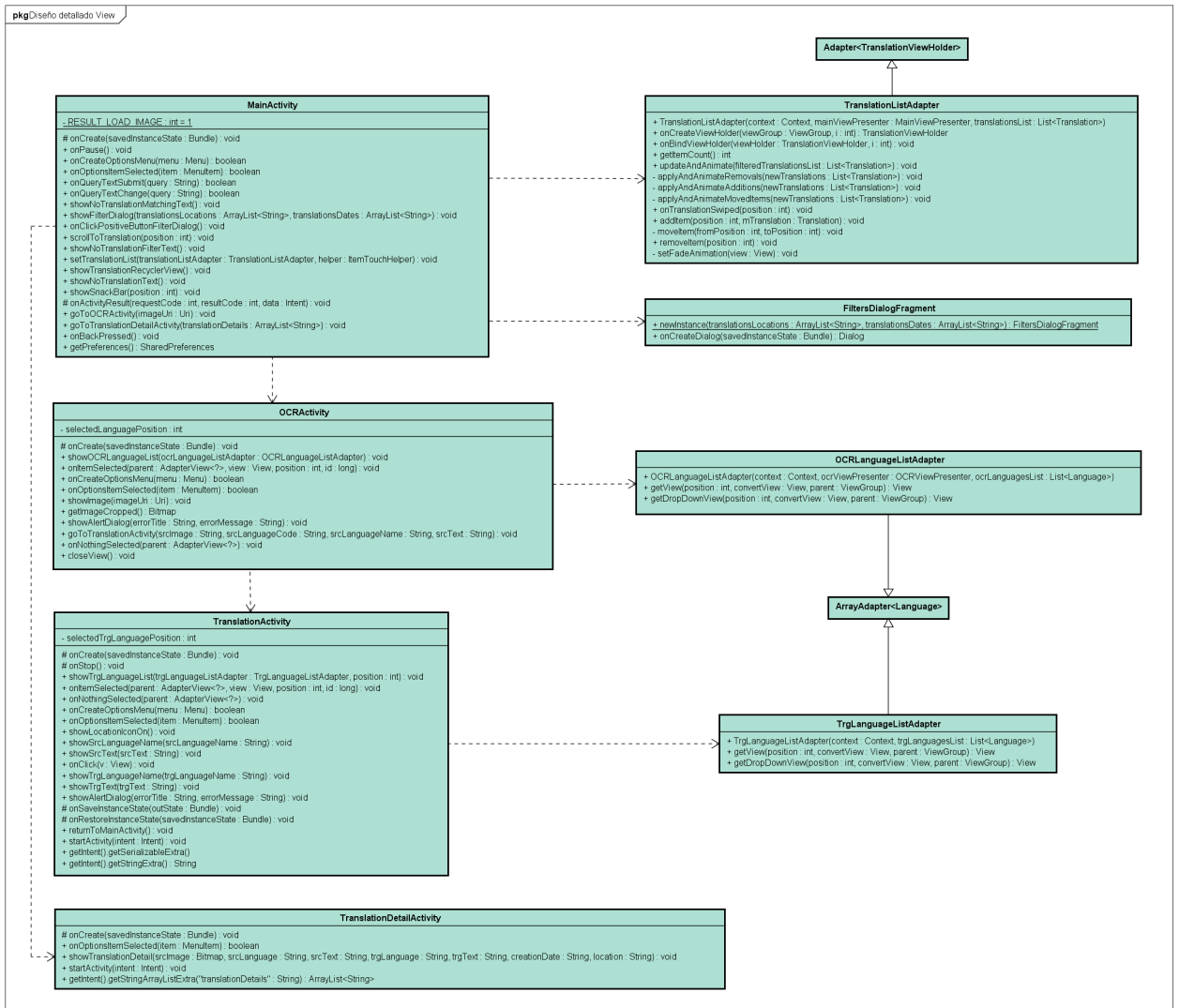


Figura 7.7: Diseño detallado del módulo Vista

7.1.5.2. Diseño detallado del módulo *Presenter*

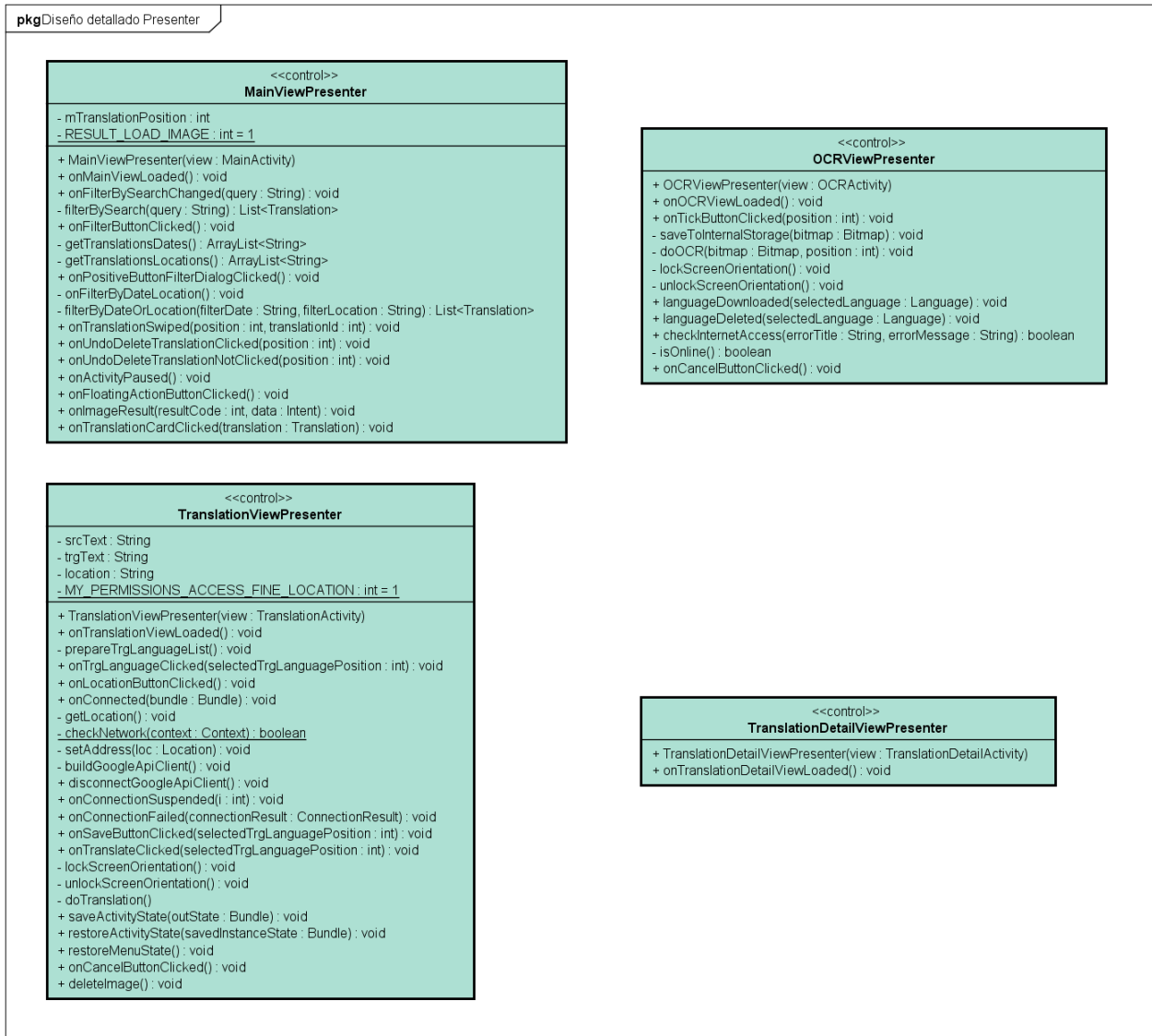


Figura 7.8: Diseño detallado del módulo Presentador

7.1.5.3. Diseño detallado del módulo *Model*

Figura 7.9: Diseño detallado del módulo Modelo

7.1.5.4. Diseño detallado del módulo *Data*

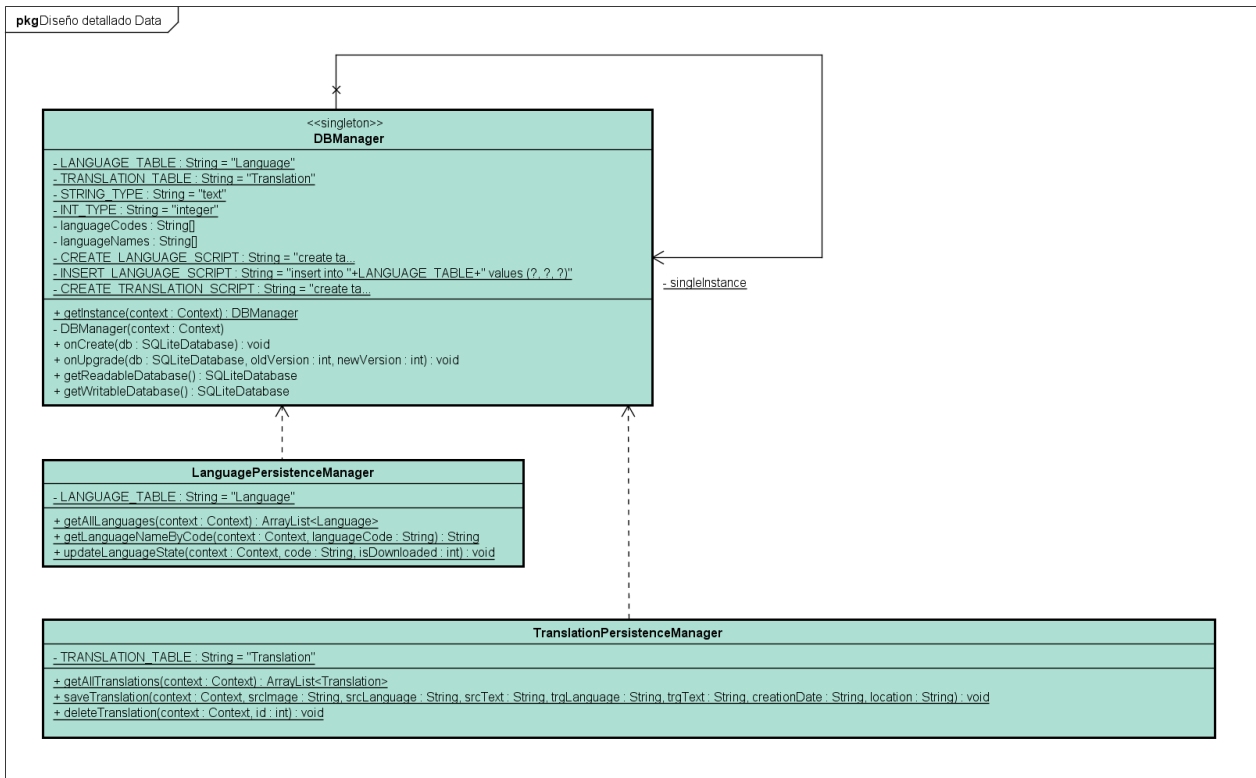


Figura 7.10: Diseño detallado del módulo Datos

7.1.5.5. Diseño detallado del módulo *Helpers*

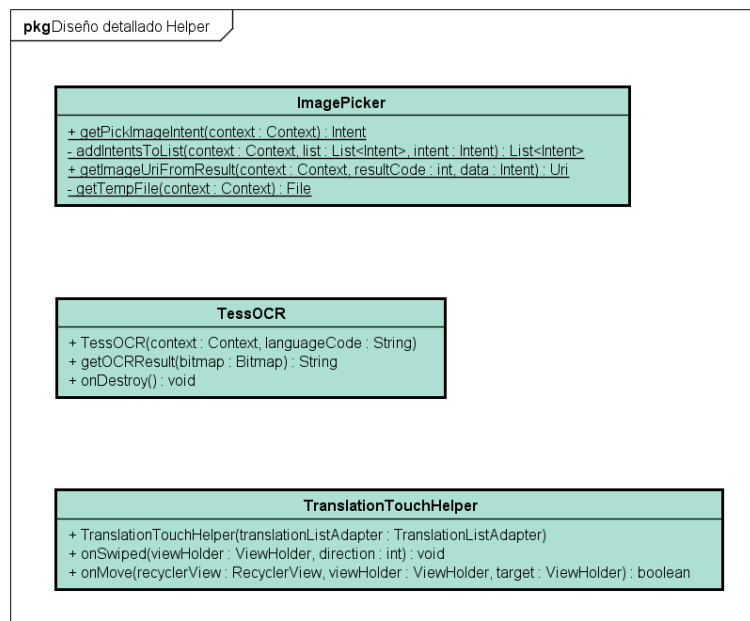


Figura 7.11: Diseño detallado del módulo Helpers

7.1.6. Relación entre módulos

En este apartado se muestran las relaciones entre las clases de diferentes módulos. Se ha decidido dividir los diagramas por casos de uso, para facilitar su entendimiento.

7.1.6.1. CU01: Añadir traducción

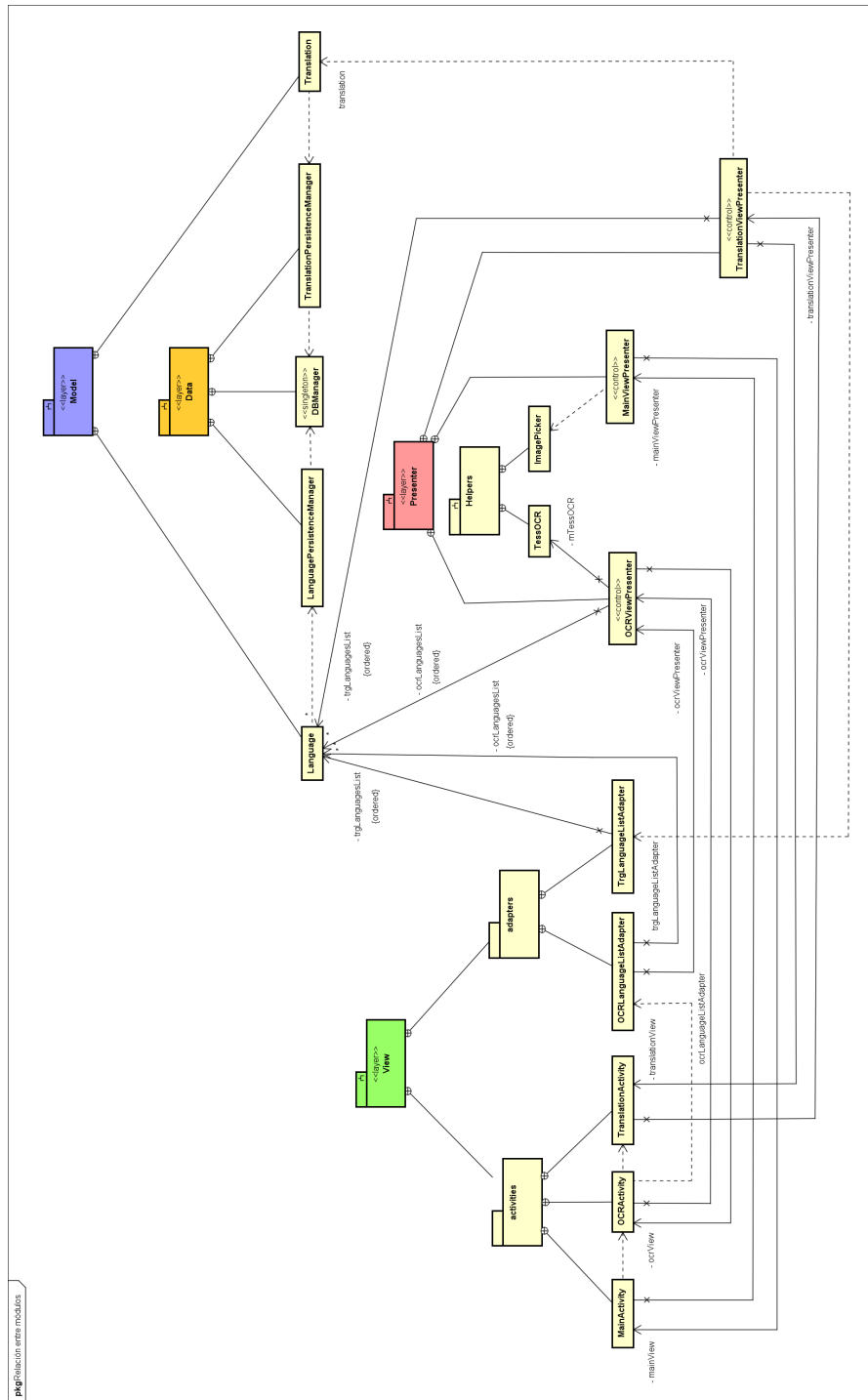


Figura 7.12: Relación entre módulos CU01

7.1.6.2. CU02: Visualizar traducciones

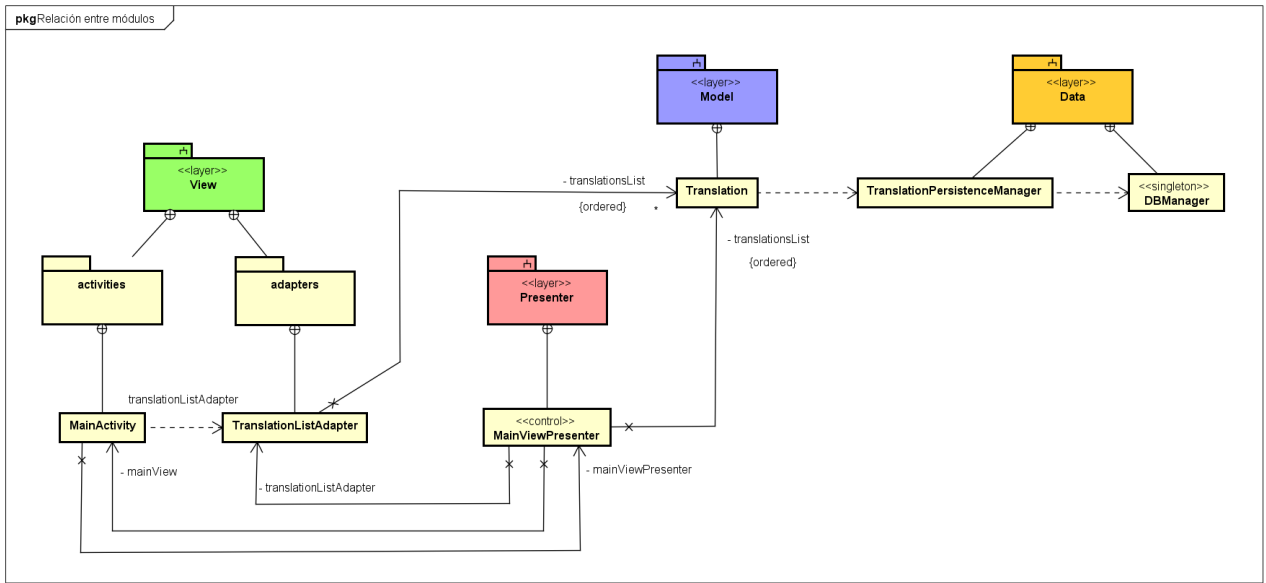


Figura 7.13: Relación entre módulos CU02

7.1.6.3. CU03: Consultar detalle de traducción

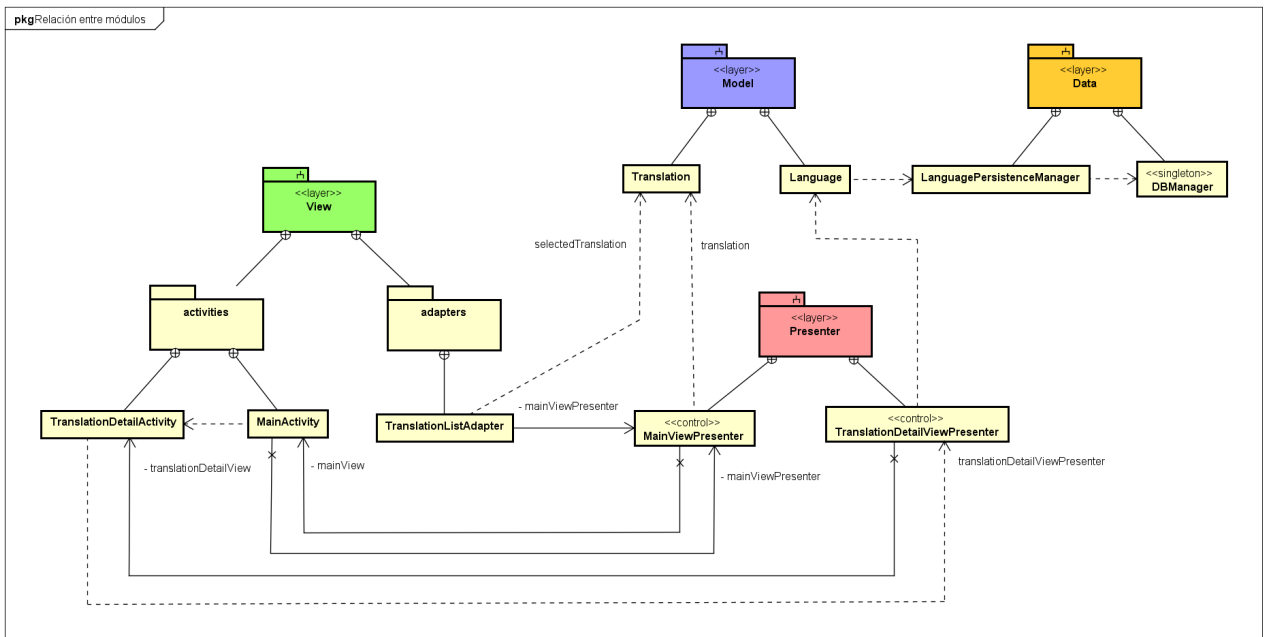


Figura 7.14: Relación entre módulos CU03

7.1.6.4. CU04: Borrar traducción

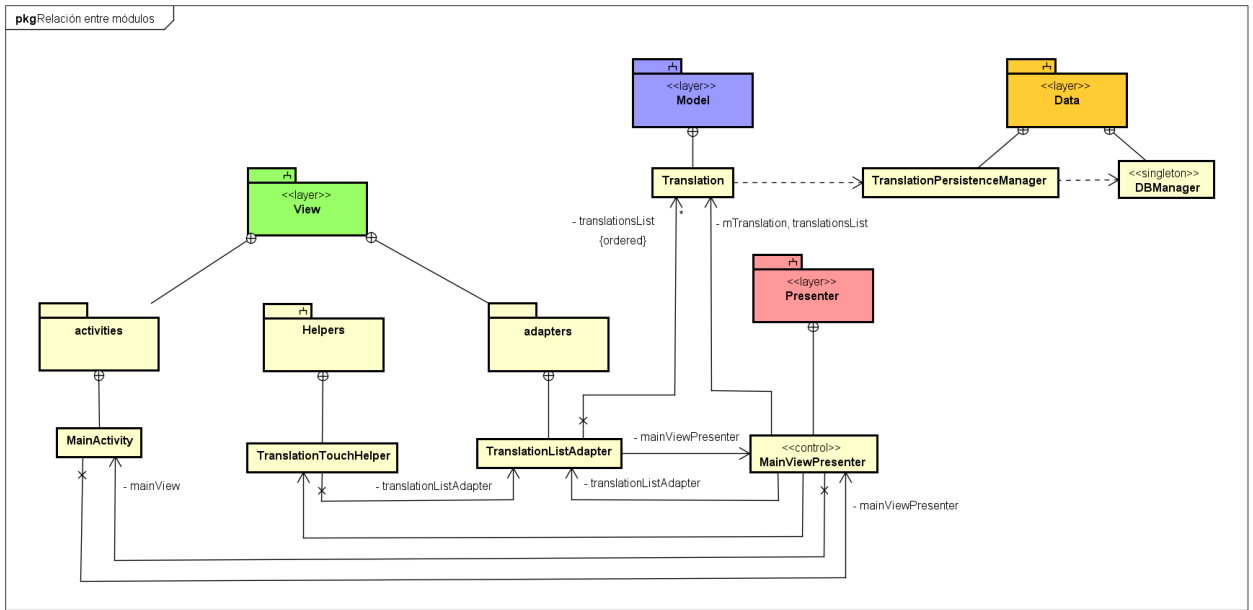


Figura 7.15: Relación entre módulos CU04

7.1.6.5. CU05: Buscar traducción

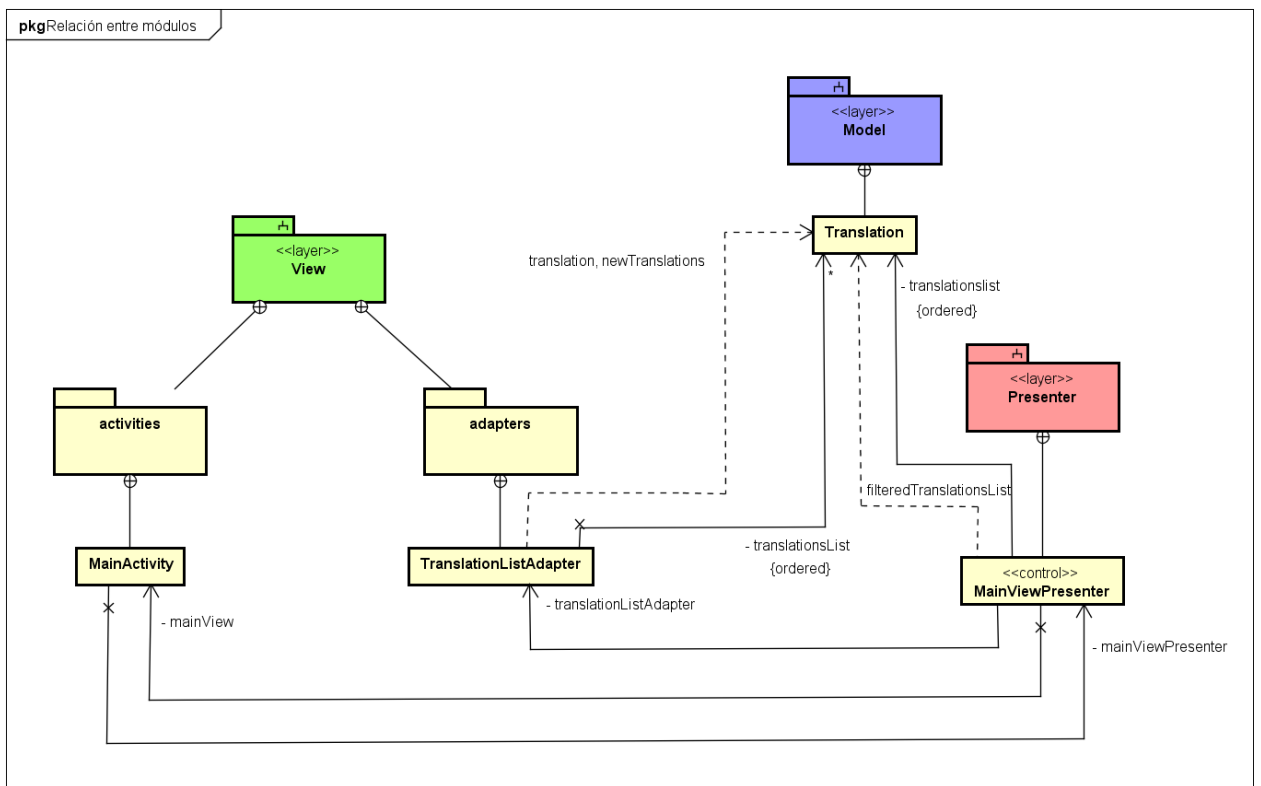


Figura 7.16: Relación entre módulos CU05

7.1.6.6. CU06: Filtrar traducciones

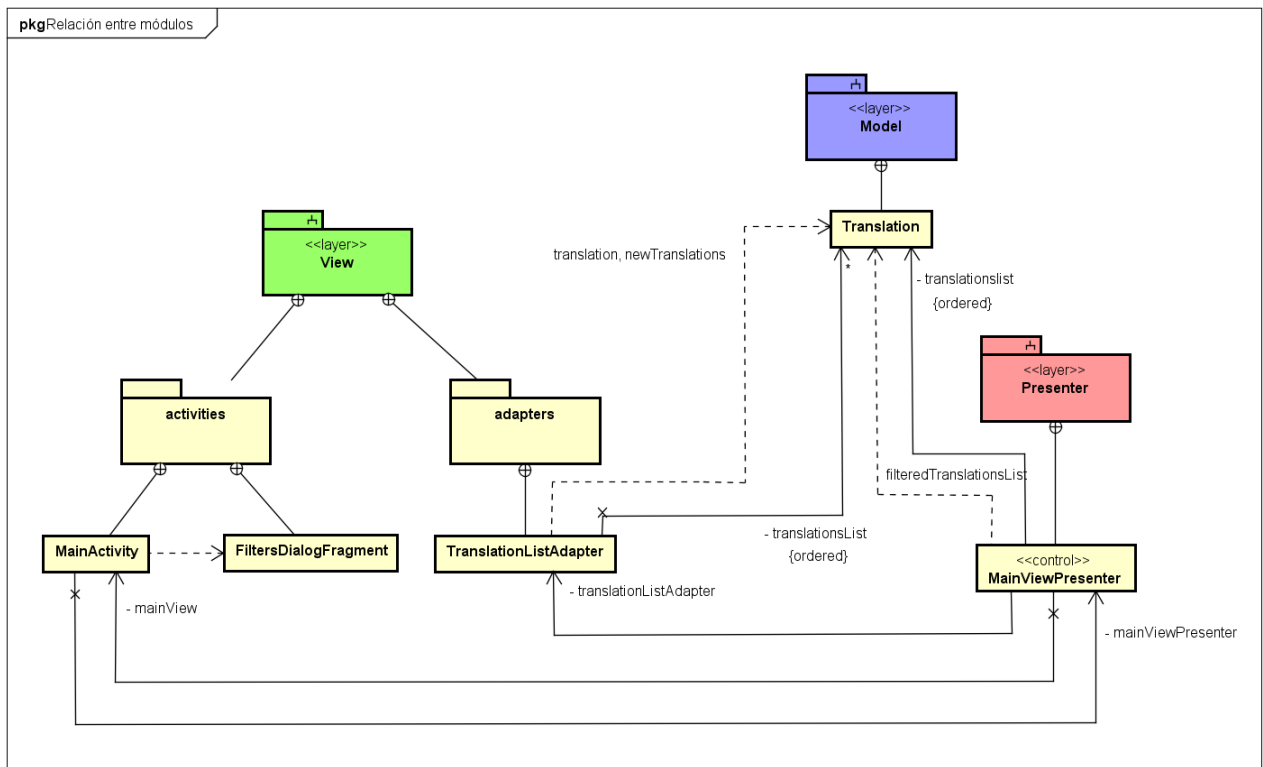


Figura 7.17: Relación entre módulos CU06

Como se puede observar en todos los diagramas de este apartado, existe relación entre los adaptadores que forman parte de la Vista y las clases del Modelo.

Según el patrón MVP, la vista no debería conocer el modelo directamente pero existe un extenso debate sobre en qué componente del patrón MVP deberían ir este tipo de adaptadores. En nuestro caso se ha decidido incluirlos en la vista porque reciben eventos y avisan a los presentadores, pero necesitan la dependencia con el modelo, lo cual contradice el patrón MVP.

7.2. Modelo de Datos

7.2.1. Esquema relacional de la base de datos

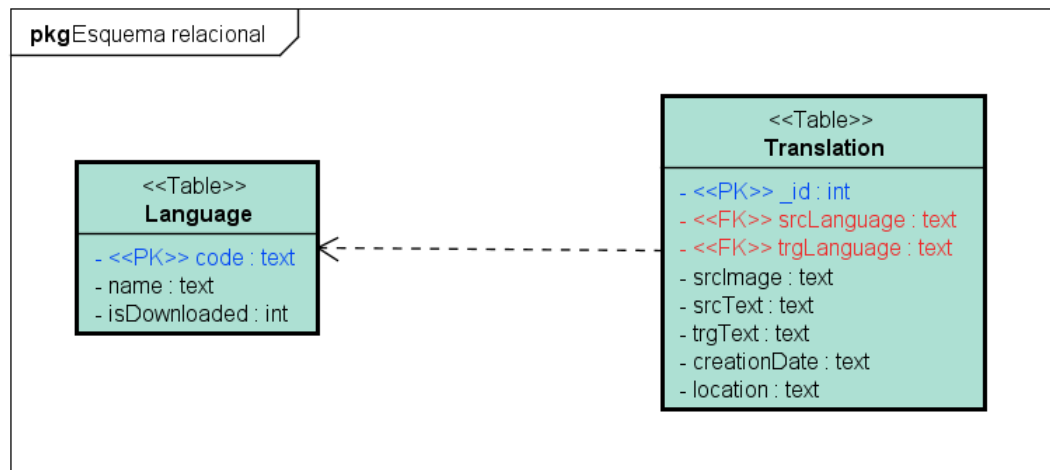


Figura 7.18: Esquema relacional de la base de datos

7.2.2. Consideraciones de las tablas del esquema relacional

7.2.2.1. Language

- **Población:** fija, no modificable, salvo en caso de que Tesseract o Microsoft añadan nuevos idiomas.
- **Columnas:**
 - *isDownloaded*: SQLite no tiene un tipo de datos destinado a almacenar booleanos, por lo que se ha decidido usar el formato integer de SQLite, representando 0 como falso (idioma no descargado) y 1 como verdadero (idioma descargado).

7.2.2.2. Translation

- **Población:** modificable, se pueden añadir y borrar traducciones, pero no actualizarlas.
- **Columnas:**
 - *creationDate*: SQLite no tiene un tipo de datos destinado a almacenar fechas, por lo que se ha decidido usar el formato text de SQLite.

7.3.2. CU01: Añadir traducción. Parte 2

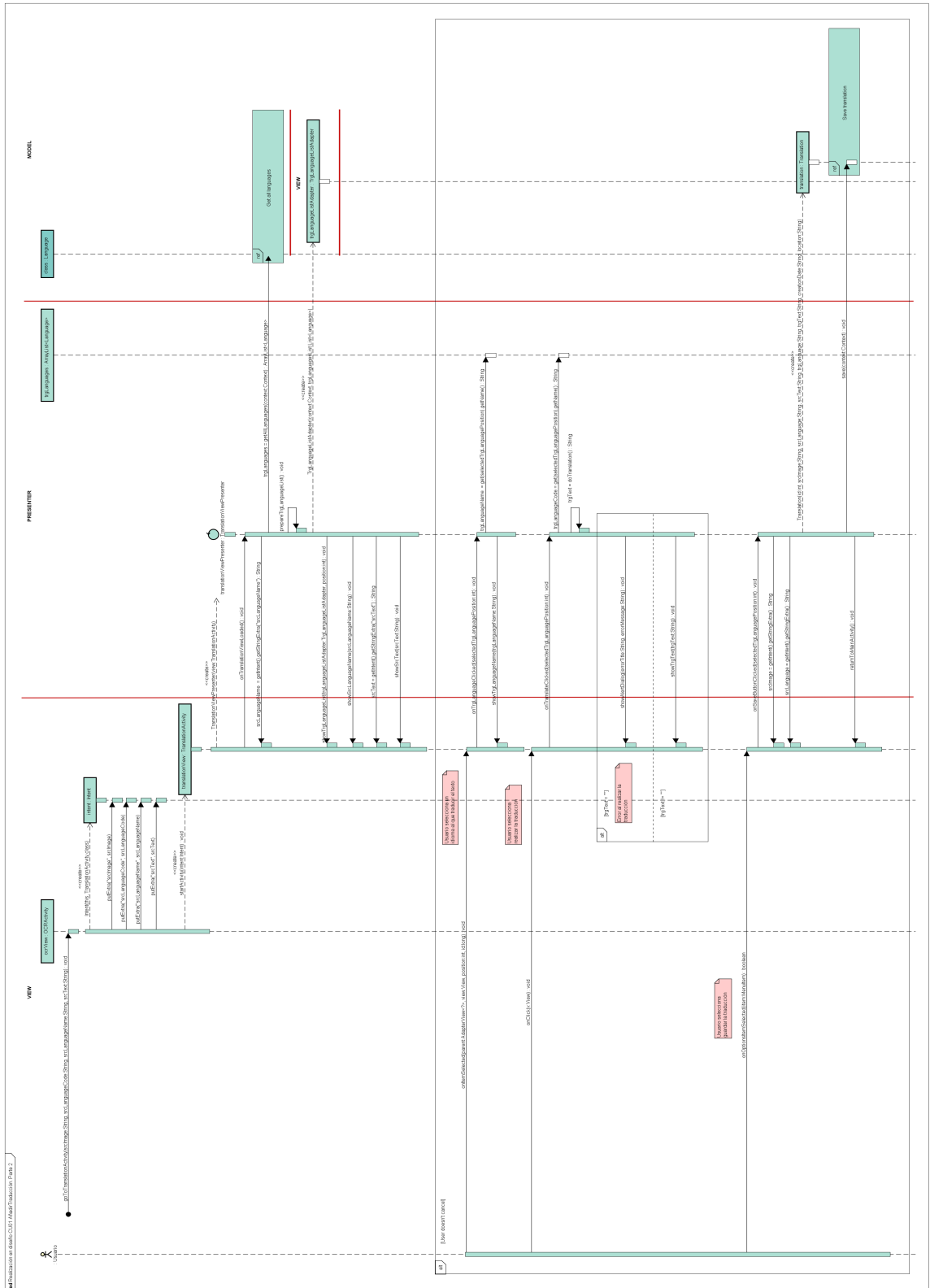


Figura 7.20: Diagrama de secuencia del CU01. Parte 2

7.3.3. Detalle de *Obtener todos los idiomas*

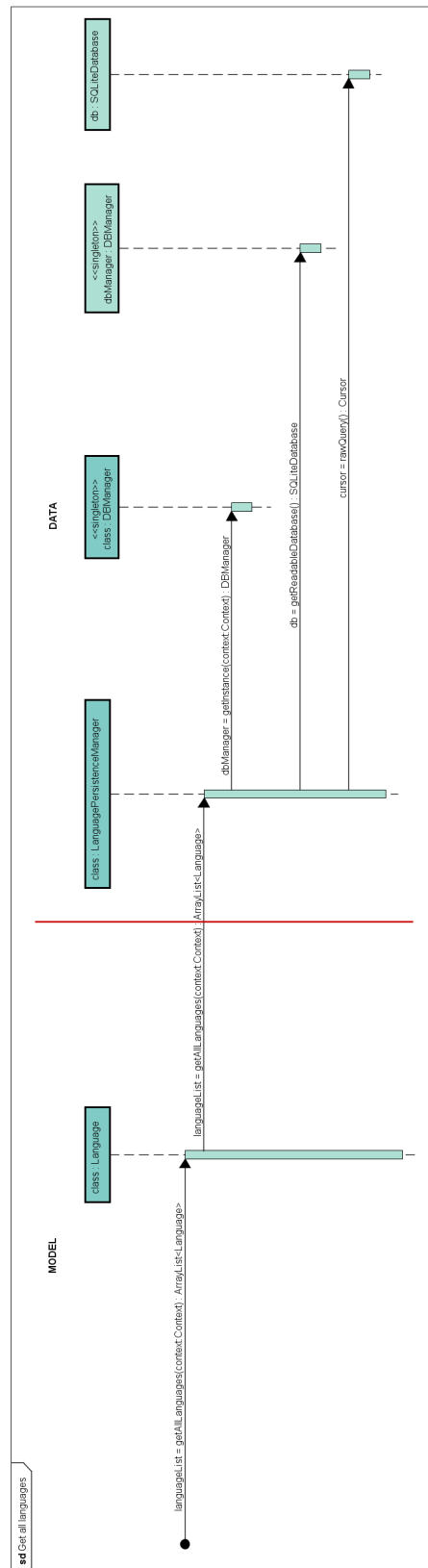


Figura 7.21: Diagrama de secuencia de *Obtener todos los idiomas*

7.3.4. Detalle de *Guardar traducción*

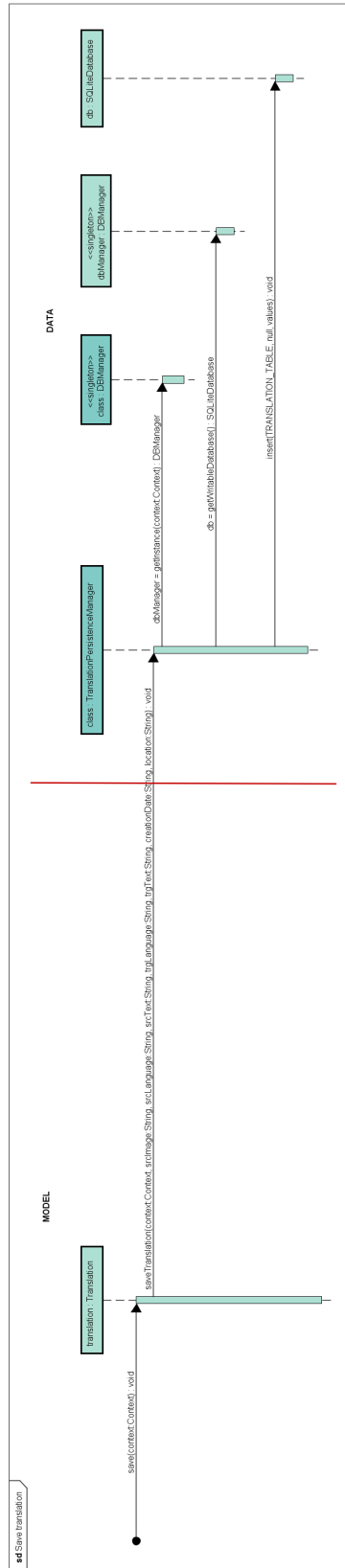


Figura 7.22: Diagrama de secuencia de *Guardar traducción*

7.3.5. CU02: Visualizar traducciones

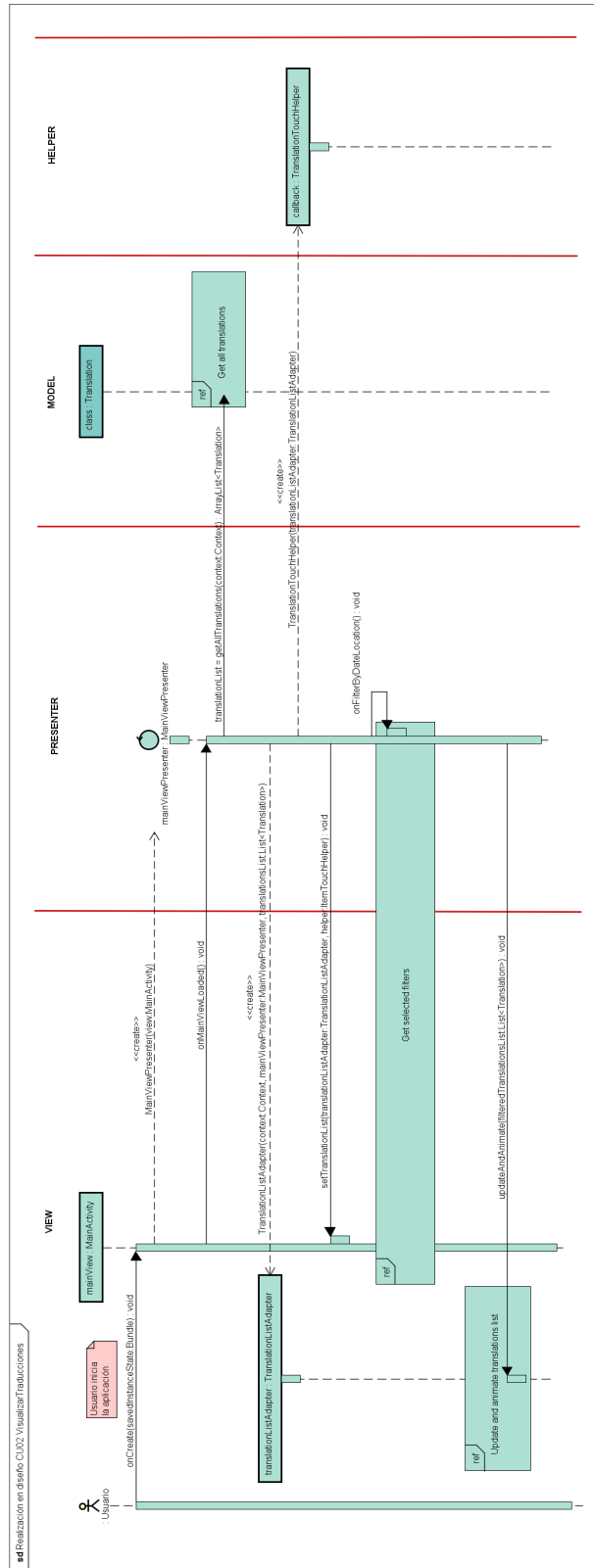


Figura 7.23: Diagrama de secuencia del CU02

7.3.6. Detalle de *Obtener todas las traducciones*

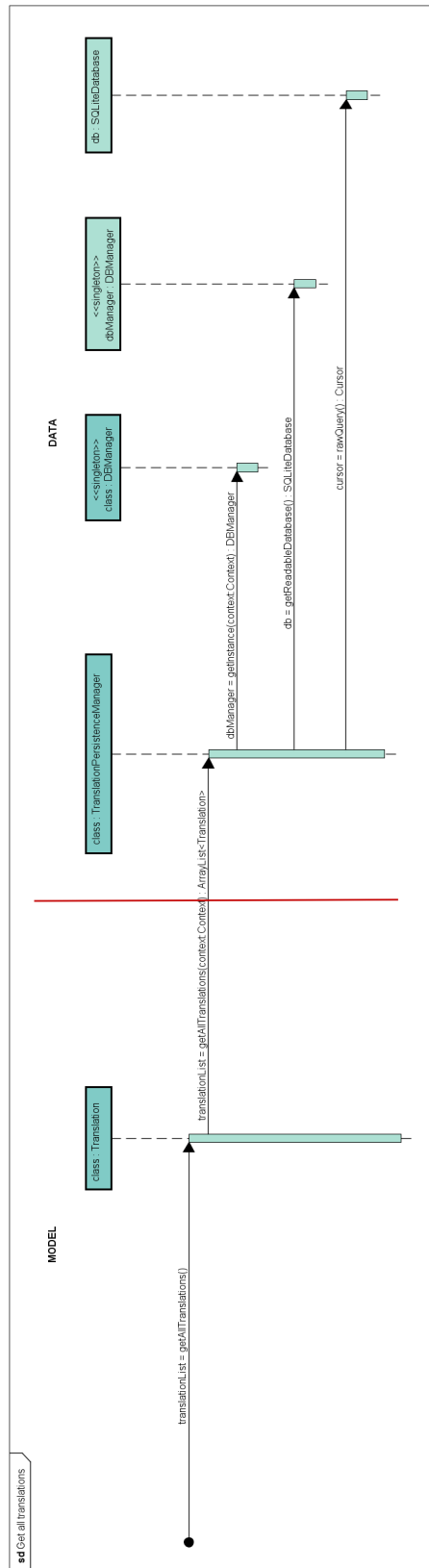


Figura 7.24: Diagrama de secuencia de *Obtener todos las traducciones*

7.3.7. Detalle de *Obtener los filtros seleccionados*

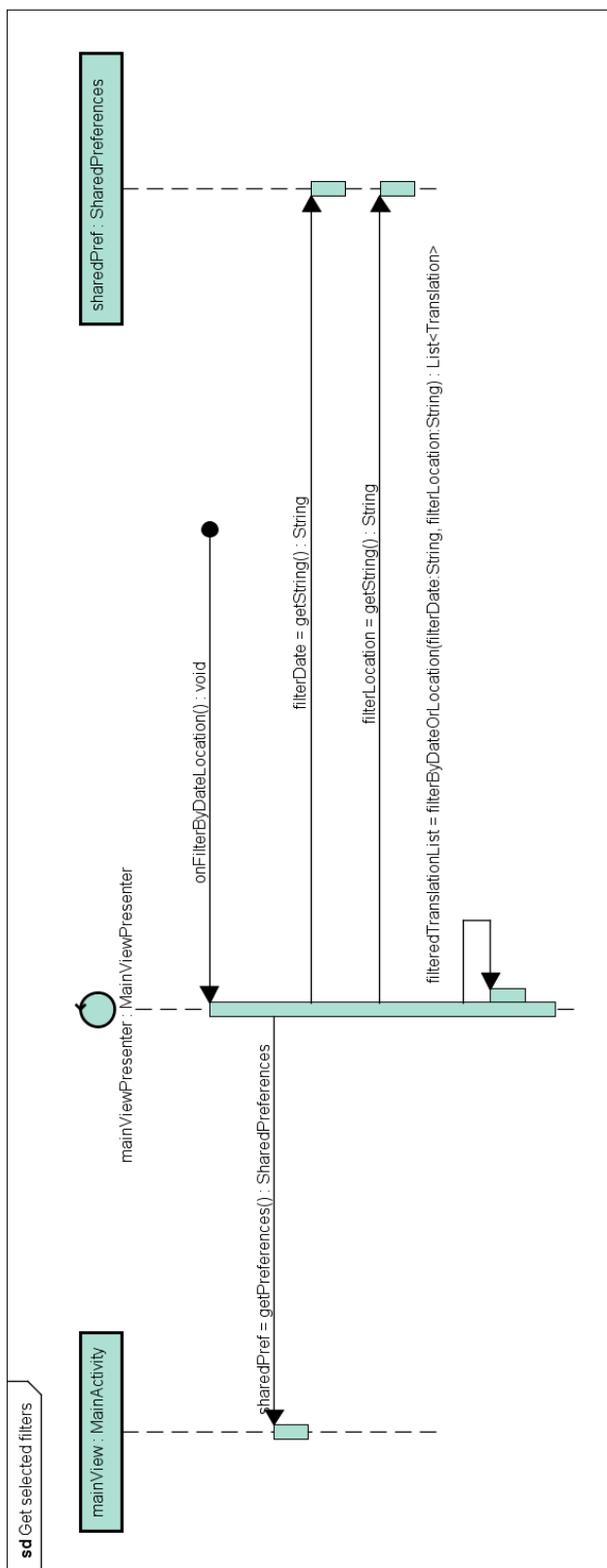


Figura 7.25: Diagrama de secuencia de *Obtener los filtros seleccionados*

7.3.8. Detalle de *Actualizar y animar la lista de traducciones*

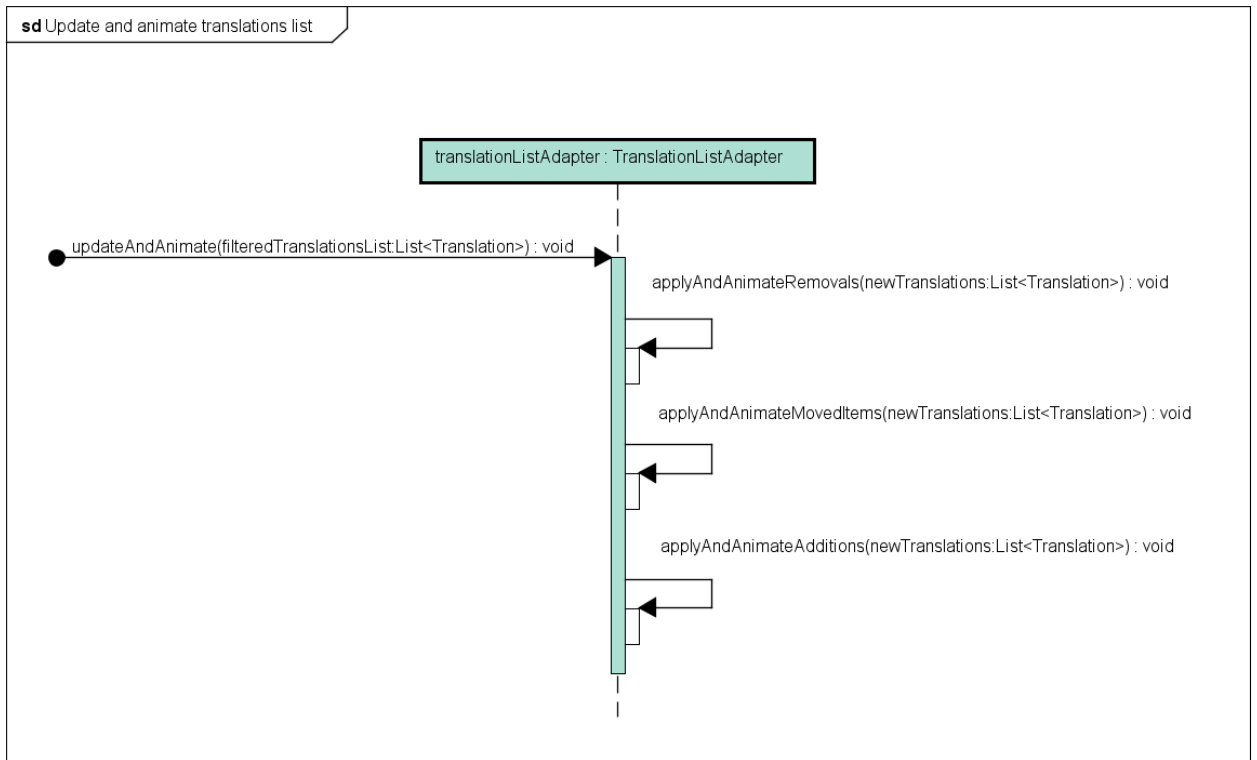


Figura 7.26: Diagrama de secuencia de *Actualizar y animar la lista de traducciones*

7.3.10. Detalle de *Obtener nombre del idioma a partir de su código*

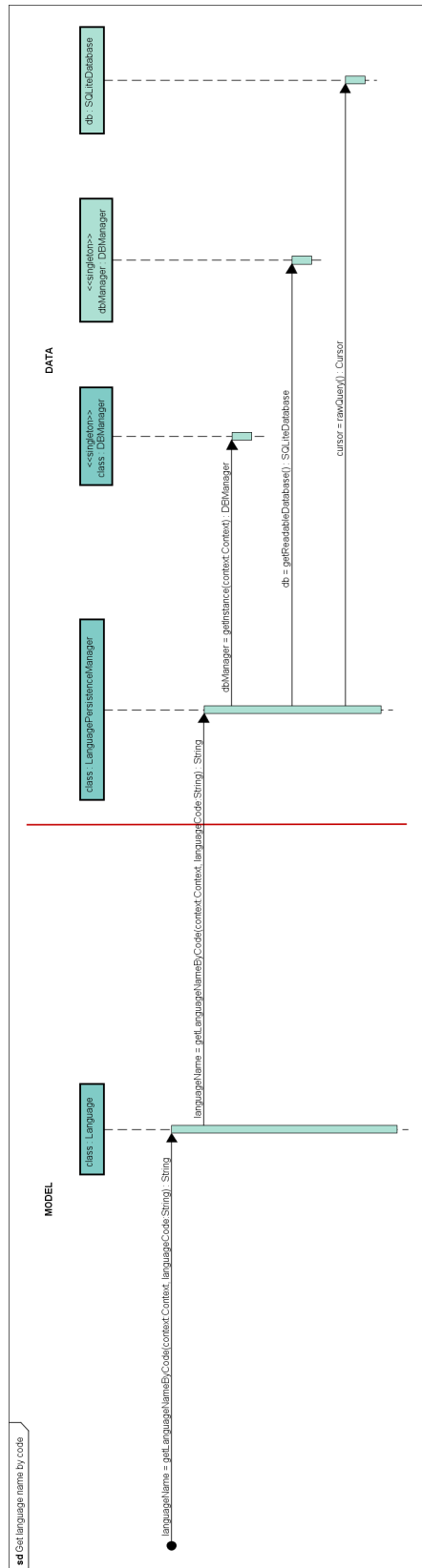


Figura 7.28: Diagrama de secuencia de *Obtener nombre del idioma a partir de su código*

7.3.11. CU04: Borrar traducción

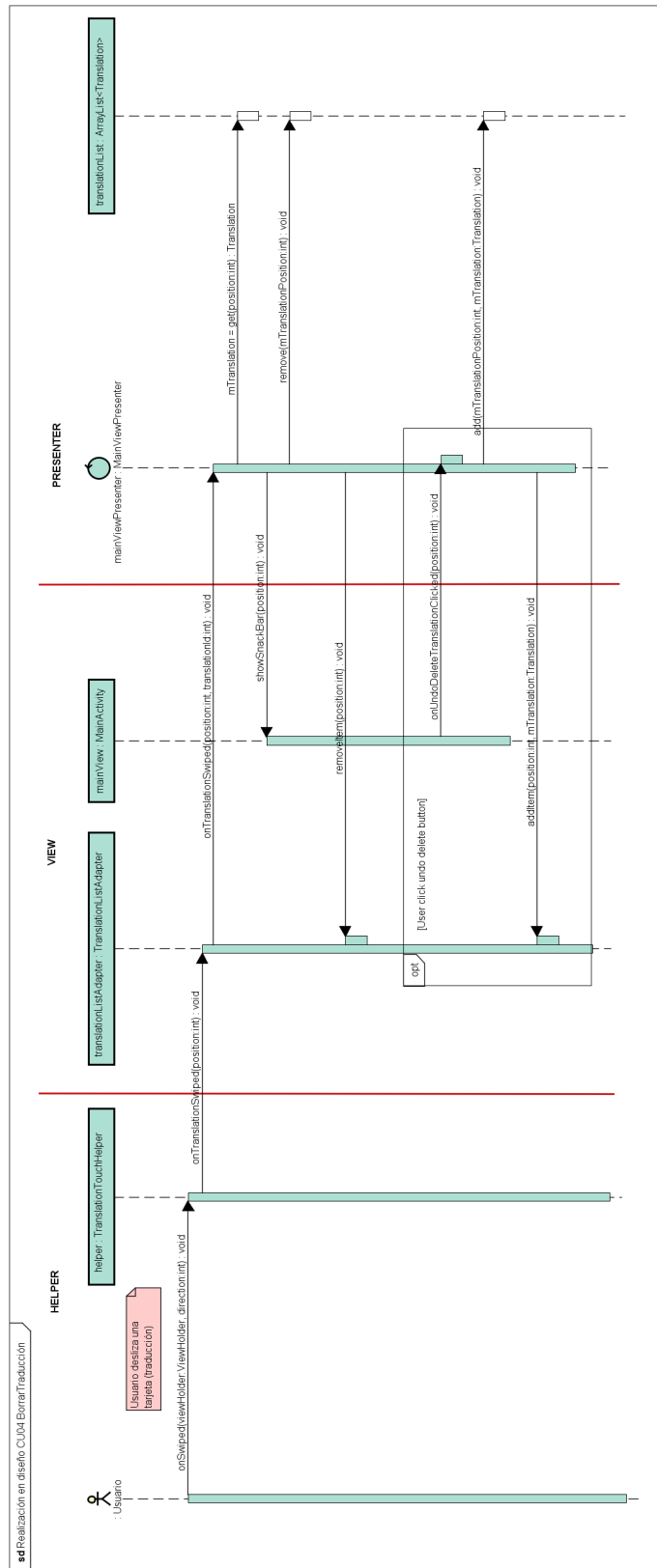


Figura 7.29: Diagrama de secuencia del CU04

7.3.12. CU05: Buscar traducción

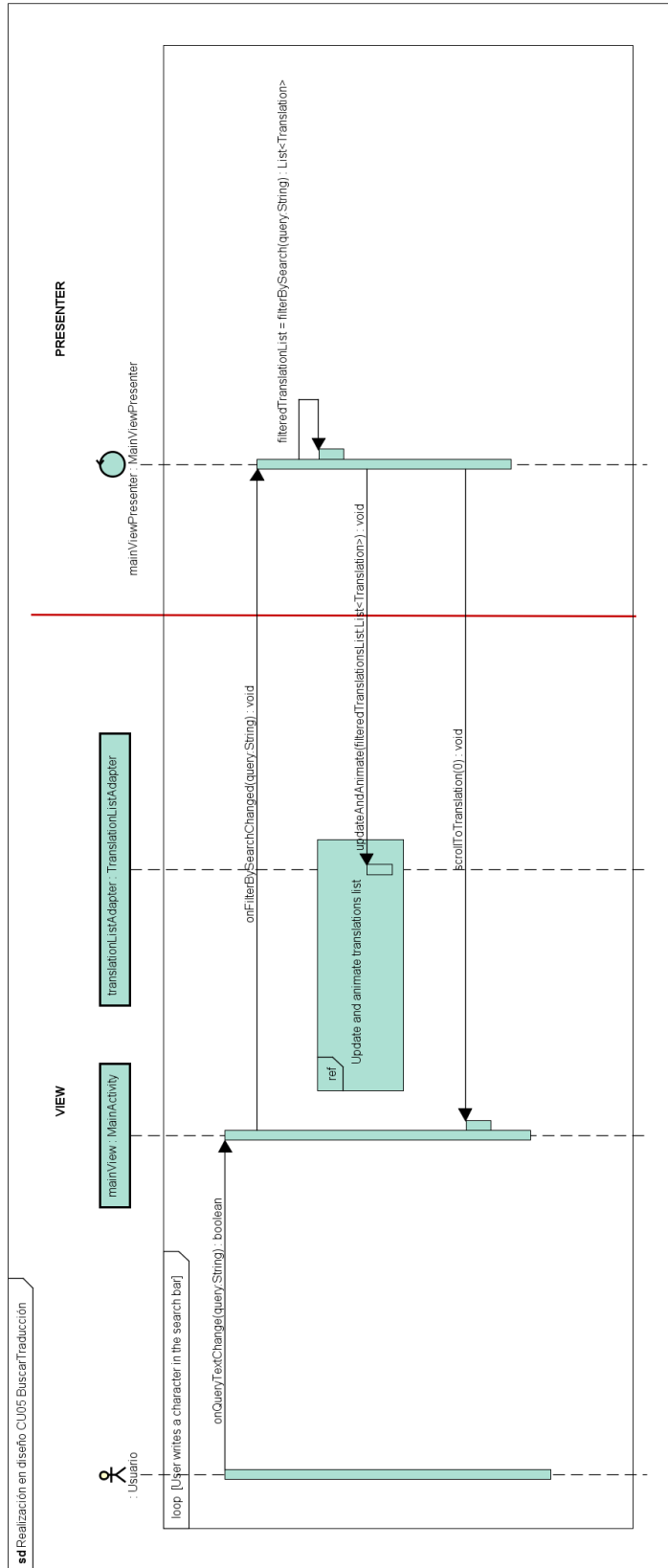


Figura 7.30: Diagrama de secuencia del CU05

7.3.13. CU06: Filtrar traducciones

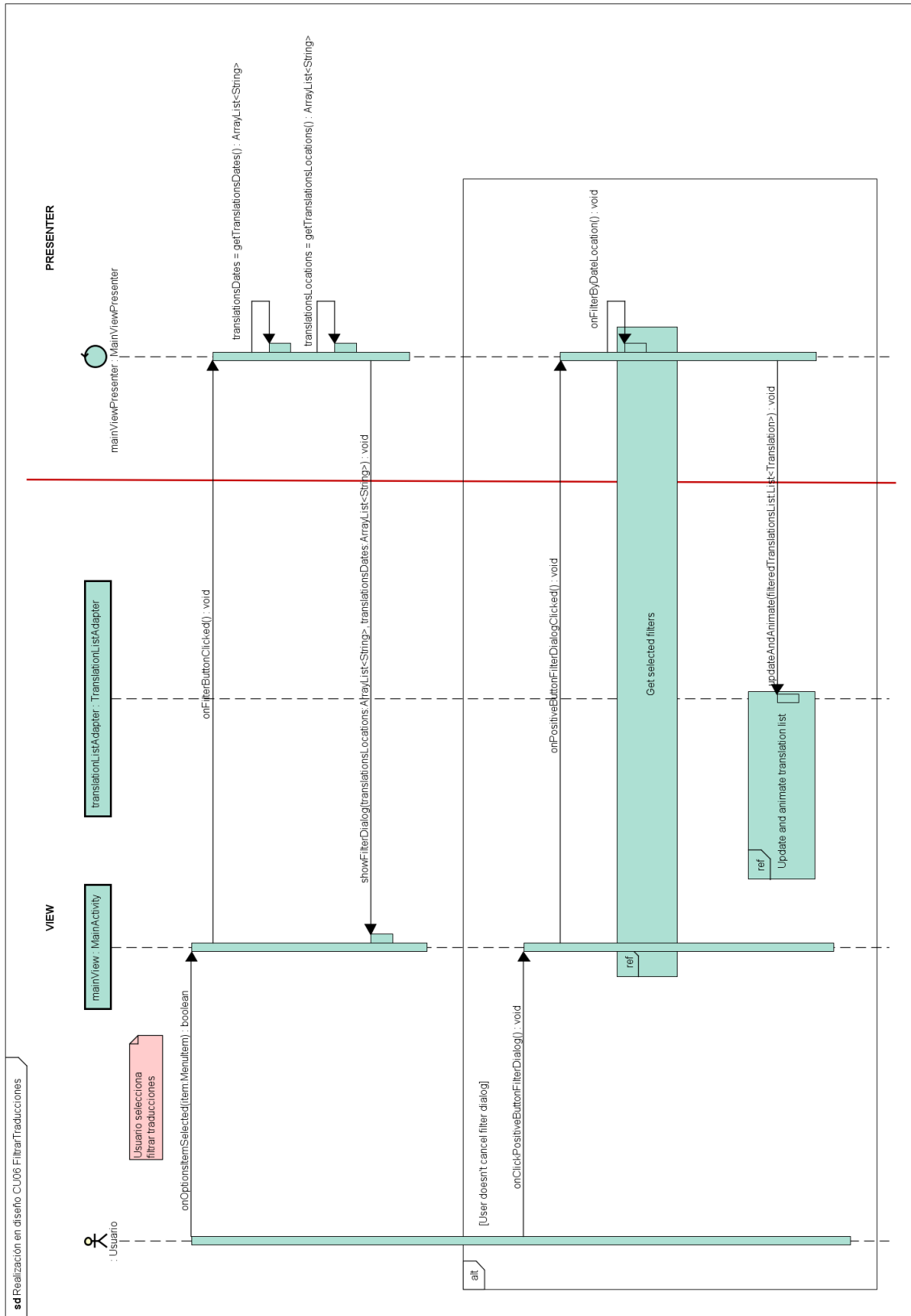


Figura 7.31: Diagrama de secuencia del CU06

7.4. Diseño de la interfaz de usuario

Los diseños de la interfaz de usuario han sido realizados con la herramienta online *NinjaMock* [20] y siguen las directivas de Material Design.

Estos diseños han sido mostrados a diferentes personas de distintas edades y perfiles, para que interactúen y opinen sobre la facilidad de uso, la claridad de los iconos utilizados y otros aspectos relevantes.

Los diseños mostrados en el siguiente apartado son los diseños UI finales, que se han obtenido después de realizar varias modificaciones con las sugerencias de los usuarios y el Product Owner.

7.4.1. CU01. Añadir traducción

Como se puede observar en los diseños de este caso de uso, el proceso de añadir una traducción se compone de una serie de pasos guiados por los que va avanzando el usuario. Se ha pretendido que la interfaz de usuario sea lo más intuitiva y clara posible.

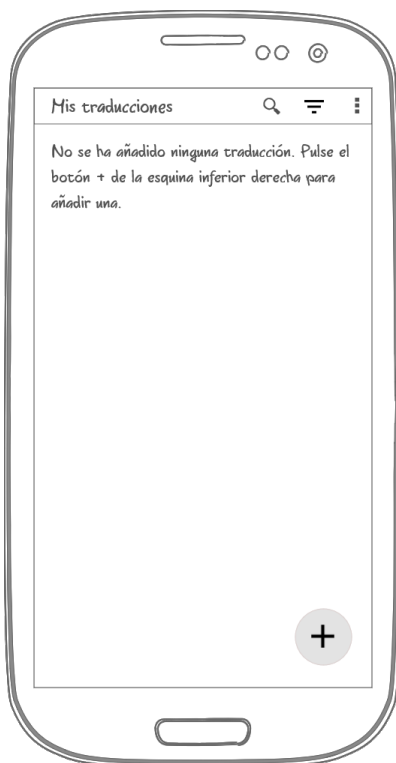


Figura 7.32: Boceto de la vista principal sin traducciones



Figura 7.33: Boceto de la vista principal: seleccionar imagen



Figura 7.34: Boceto de la vista de reconocimiento: seleccionar/descargar idioma de reconocimiento

En la figura 7.32 aparece un botón en la esquina inferior derecha, que se utiliza para añadir una nueva traducción.

En la figura 7.32 y 7.33 aparecen tres iconos en la barra superior:

- **Icono de búsqueda:** se utiliza para filtrar las traducciones según el criterio de búsqueda introducido por el usuario. Ver apartado 7.4.5.
- **Icono de filtros:** abre un diálogo en el que se pueden seleccionar fecha y localización para filtrar las traducciones. Ver apartado 7.4.6.
- **Icono de más información:** aparece información sobre el proyecto, el alumno y otros.

En la figura 7.34 aparece una lista desplegable con los idiomas de reconocimiento, y un botón a la derecha de cada uno de ellos indicando aquellos que están descargados y ofreciendo la posibilidad de descargar los que no aún no lo están.

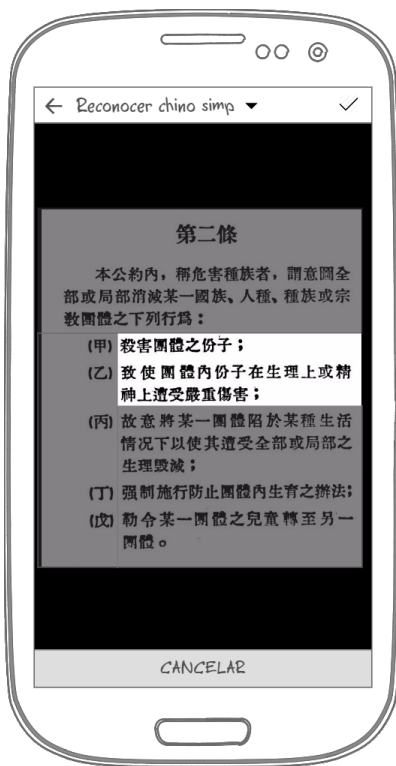


Figura 7.35: Boceto de la vista de reconocimiento: seleccionar parte de la imagen y aceptar



Figura 7.36: Boceto de la vista de traducción: seleccionar idioma de llegada en la traducción

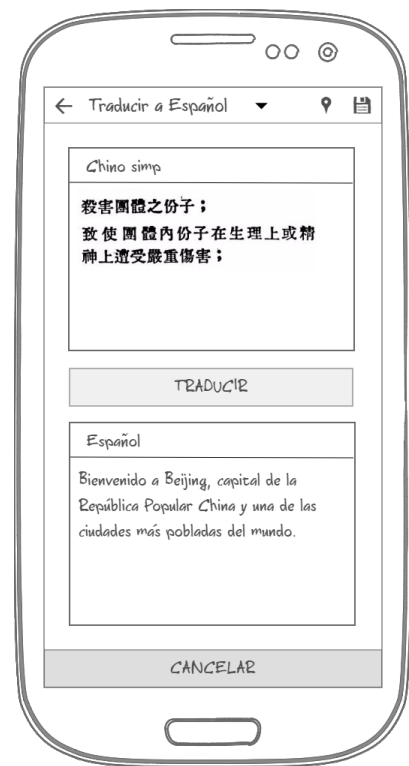


Figura 7.37: Boceto de la vista de traducción: realizar traducción y guardarla

En la figura 7.35 aparece un rectángulo ajustable para seleccionar la parte de la imagen a reconocer y un botón en la esquina derecha de la barra superior para iniciar el reconocimiento. También aparece un botón en la parte inferior para cancelar.

En la figura 7.36 aparece una lista desplegable con los idiomas a los que traducir el texto reconocido.

En la figura 7.37 aparece lo siguiente:

- Idioma del texto reconocido y el texto reconocido como tal.
- Botón para realizar la traducción.
- Idioma al que traducir y el texto traducido.
- Botón para incluir la localización en la traducción a guardar.
- Botón para guardar la traducción.
- Botón para cancelar.

7.4.2. CU02. Visualizar traducciones

En la figura 7.38 aparecen las traducciones realizadas por el usuario representadas en forma de tarjetas.

En cada una de las tarjetas que representan una traducción aparece la siguiente información:

- Miniatura de la imagen que contiene el texto a reconocer.
- Texto traducido.
- Fecha en la que se ha realizado la traducción.
- Localización donde se ha realizado la traducción.

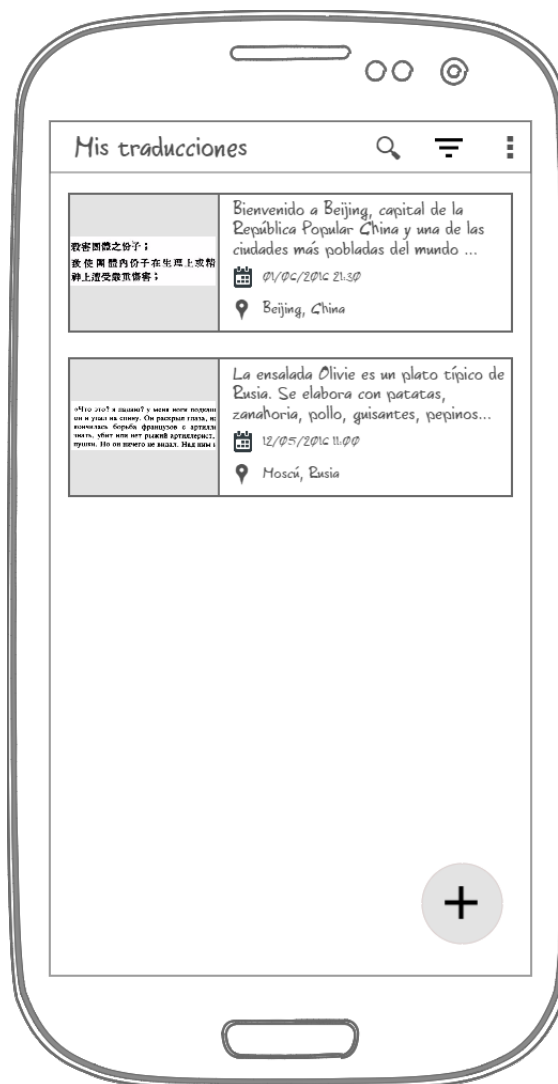


Figura 7.38: Boceto de la vista principal con traducciones

7.4.3. CU03. Consultar detalle de traducción

En la figura 7.39 aparece el detalle de una traducción que contiene lo siguiente:

- Fecha y localización
- Texto reconocido y su idioma
- Texto traducido y su idioma

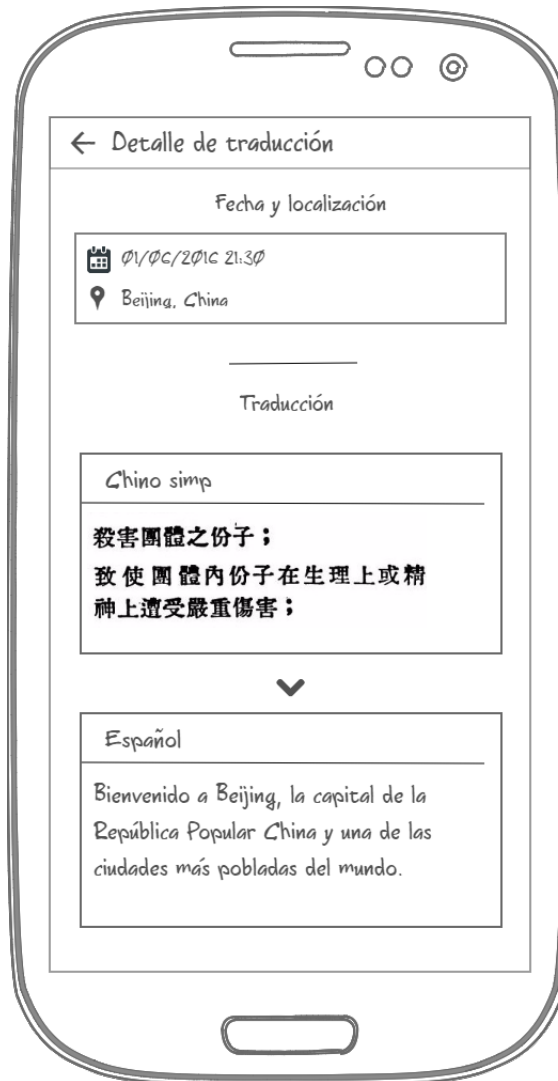


Figura 7.39: Boceto de la vista de detalle de traducción

7.4.4. CU04. Borrar traducción

En la figura 7.40 se representa el deslizamiento de una tarjeta para borrar la traducción correspondiente.

En la figura 7.41 la tarjeta deslizada desaparece y aparece una barra negra en la parte inferior con un botón para deshacer el borrado.

En la figura 7.42 aparece la vista principal después de **no** haber pulsado el botón de deshacer el borrado.

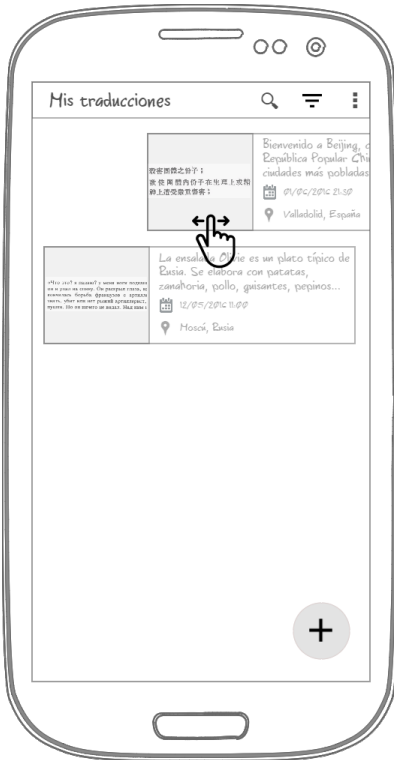


Figura 7.40: Boceto de la vista de borrar traducción: deslizar tarjeta para borrar la traducción

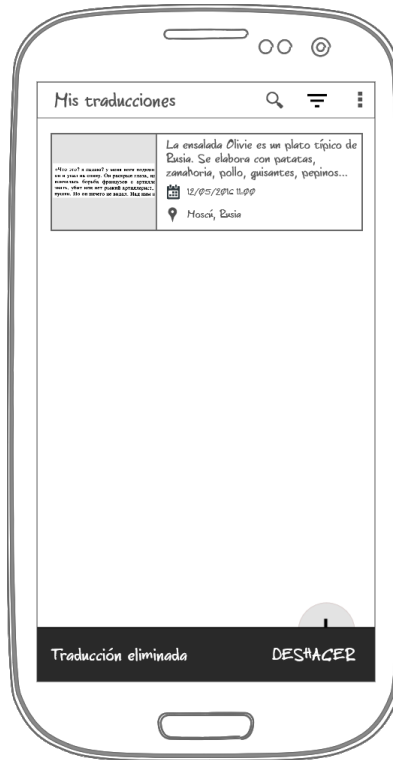


Figura 7.41: Boceto de la vista de borrar traducción: opción para deshacer el borrado

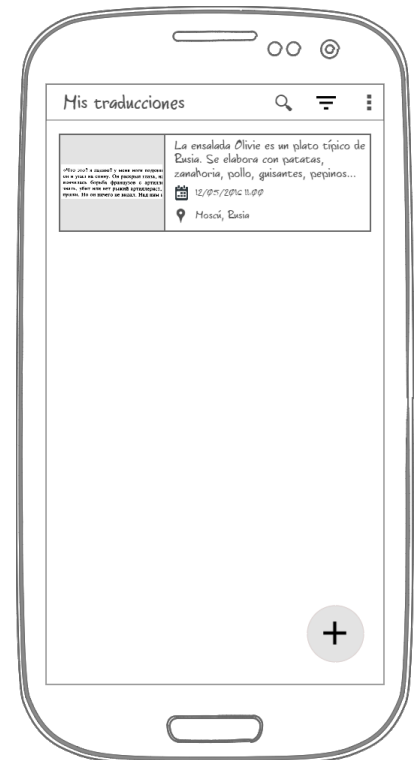


Figura 7.42: Boceto de la vista de borrar traducción: traducción borrada

7.4.5. CU05. Buscar traducción

En la figura 7.43 aparece una barra de búsqueda y las traducciones correspondientes al criterio de búsqueda introducida. También aparece una X para cancelar la búsqueda.

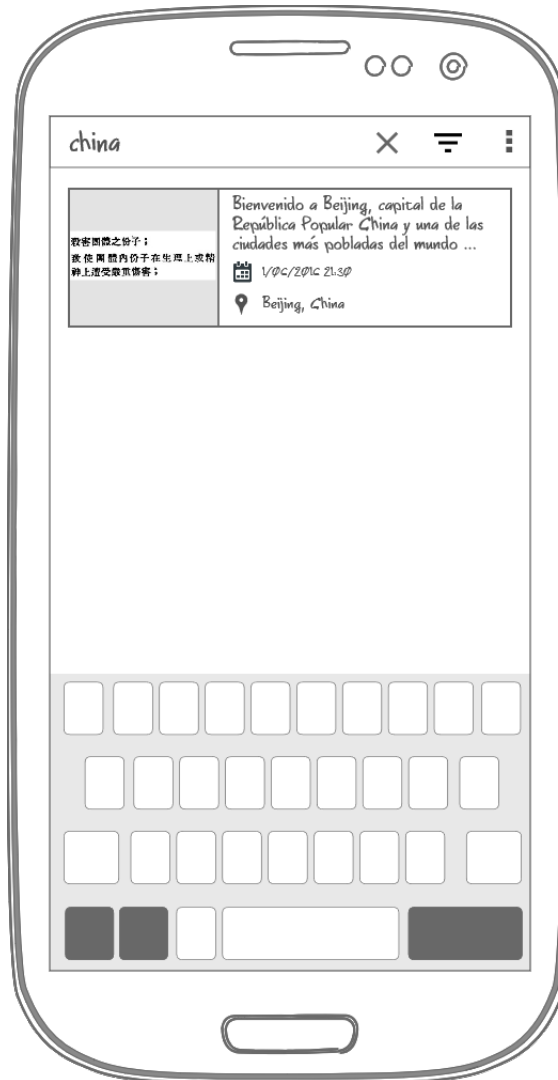


Figura 7.43: Boceto de la vista principal realizando una búsqueda

7.4.6. CU06. Filtrar traducciones

En las figuras 7.44, 7.45 y 7.46 aparece un diálogo para que el usuario seleccione los filtros que desee.

En la figura 7.44 aparece una lista de fechas con las que filtrar las traducciones.

En la figura 7.45 aparece una lista de localizaciones con las que filtrar las traducciones.

En la figura 7.46 aparecen los filtros que haya seleccionado el usuario y dos botones para aceptar o cancelar los filtros seleccionados.

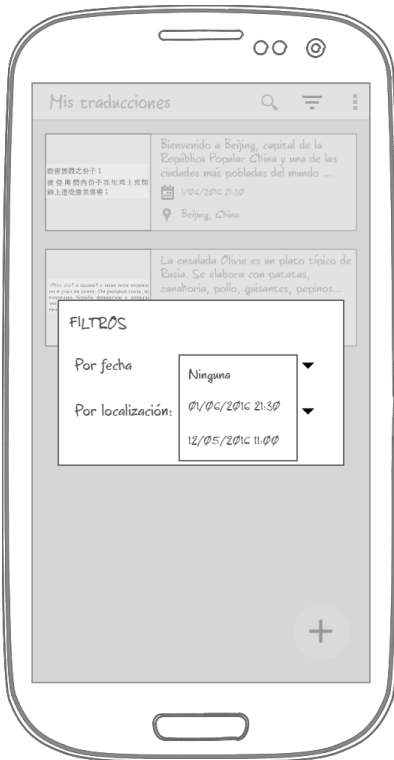


Figura 7.44: Boceto de la vista de filtrar traducciones: seleccionar fecha

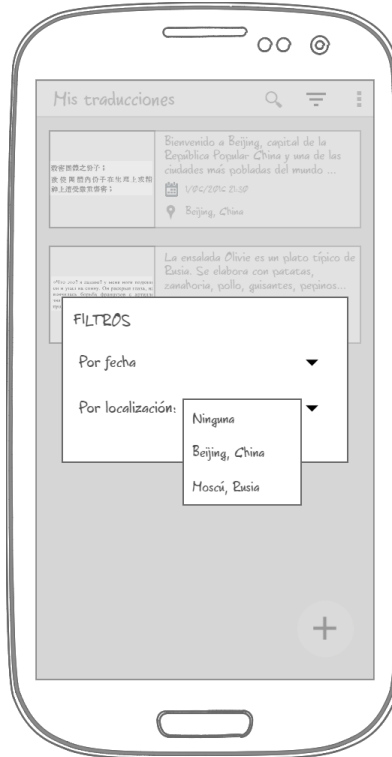


Figura 7.45: Boceto de la vista de filtrar traducciones: seleccionar localización



Figura 7.46: Boceto de la vista de filtrar traducciones: aceptar filtros seleccionados

En la figura 7.47 aparecen las traducciones correspondientes a los filtros seleccionados por el usuario.

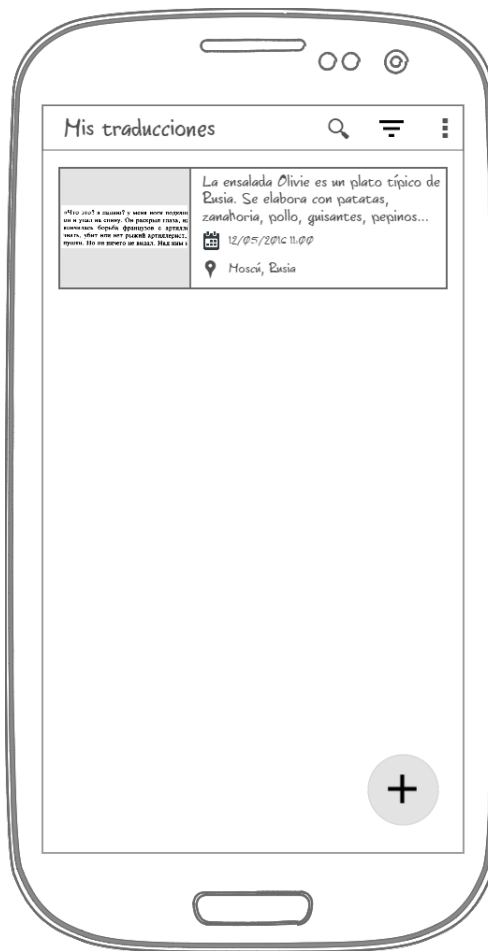


Figura 7.47: Boceto de la vista de filtrar traducciones: traducciones filtradas

Capítulo 8

Implementación y pruebas

8.1. Implementación

La implementación de *Asistev* se ha realizado en varias fases:

- **Versión 0.1 ó Alpha:** permite seleccionar parte de la imagen sobre la que realizar el reconocimiento, completa el reconocimiento de caracteres de un texto en inglés contenido en una imagen, obtenida desde la cámara o desde la galería, y lo muestra por pantalla.
- **Versión 0.2 ó Beta:** además de la funcionalidad de la versión Alpha, permite seleccionar, descargar y/o borrar idiomas de reconocimiento y seleccionar un idioma al que traducir el texto reconocido.
- **Versión 1.0 o Release:** añade soporte para 39 idiomas, tanto para el reconocimiento como para la traducción. Además, añade la persistencia de los idiomas y las traducciones e implementa el resto de la funcionalidad (historial de traducciones, búsqueda y filtrado).

El código fuente del proyecto está dividido en dos módulos:

- **app:** es el módulo de la aplicación como tal, contiene cinco paquetes con 20 clases Java, que se corresponden a los de la figura 7.6. También contiene 19 layouts para las vistas, iconos, colores, strings y otros ficheros. Este módulo implementa toda la funcionalidad de la aplicación excepto el reconocimiento de caracteres.
- **tess-two:** implementa el motor de reconocimiento de caracteres u OCR. Está formado por las bibliotecas de Tesseract y *Leptonica* y otras clases escritas en lenguaje C o C++. El módulo app se conecta con este módulo para realizar el reconocimiento óptico de caracteres.

Nota importante: el proyecto con el código fuente incluido en el CD contiene un único módulo app en vez de dos. El módulo de más se debe a que para aprender a utilizar Android NDK, se han construido las bibliotecas de reconocimiento a mano, y se han añadido a un nuevo módulo, el cual ocupa casi 2GB de espacio [21]. Por otro lado, el proyecto entregado contiene, en el propio módulo app, las bibliotecas precompiladas en un AAR, ocupando mucho menos espacio.

8.2. Plan de pruebas

8.2.1. Introducción

En un proyecto de desarrollo de software la realización de pruebas es fundamental para garantizar el correcto funcionamiento de la aplicación, así como su fiabilidad, consistencia y estabilidad.

Es importante realizar pruebas durante todas las fases de desarrollo de la aplicación. De esta forma, si aparece un fallo, se puede solucionar lo antes posible y evitar que perjudique al resto de sistema según va avanzando el desarrollo.

En todo proyecto, un fallo no detectado a tiempo implica un coste, tanto de recursos como económico, ya que supone que un desarrollador dedique tiempo a solucionarlo, tiempo que implica un coste económico.

En este plan, se definen los diferentes tipos de pruebas realizadas y se explica, mediante casos de prueba, todas las pruebas realizadas sobre la aplicación desarrollada.

8.2.2. Tipos de pruebas realizadas

8.2.2.1. Pruebas unitarias

Este tipo de pruebas se utiliza para comprobar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado.

Son llevadas a cabo durante el desarrollo y el encargado de realizarlas puede ser un testeador como tal o el propio desarrollador del código, al ser el que mejor lo conoce.

En nuestro caso, estas pruebas se han ido realizando a medida que se terminaba un método. Para cada método finalizado, el testeador escogía distintos valores de entrada para examinar cada uno de los posibles flujos de ejecución del método y cerciorarse de que se devuelven los valores de salida adecuados. Esto es comúnmente conocido como prueba de caja blanca.

8.2.2.2. Pruebas de integración

Las pruebas integrales o pruebas de integración son aquellas que se realizan una vez aprobadas las pruebas unitarias. Consisten en combinar módulos individuales de software y probarlos como un grupo.

Por lo general, este tipo de pruebas suelen ser de caja negra, es decir, se estudia el sistema desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno.

En nuestro caso, estas pruebas se han ido realizando para cada funcionalidad, combinando los módulos de código implicados en cada una de ellas y llevando a cabo una prueba conjunta.

8.2.2.3. Pruebas de sistema

Este tipo de pruebas se realizan una vez terminadas las pruebas de integración. El objetivo principal es probar todo el sistema completo e integrado.

Al igual que las pruebas de integración, este tipo de pruebas también suelen ser de caja negra.

8.2.3. Casos de Pruebas

En este apartado se detallan los casos de prueba, guiados por casos de uso, realizados para evaluar la aplicación desarrollada. Para cada una de los casos de prueba se explicará una breve descripción del caso, la acción a realizar por el usuario, el resultado esperado, el resultado obtenido y la solución aplicada en el caso de que no se haya obtenido el resultado esperado.

8.2.3.1. CU01. Añadir traducción

CP01	Obtención de imagen a reconocer desde la cámara
Descripción	Un Usuario selecciona la cámara para obtener la imagen que será usada en el reconocimiento y realiza una foto.
Acción	Ir a la vista principal de la aplicación, pulsar el botón de añadir traducción (+), seleccionar la cámara, realizar la foto y aceptar.
Resultado esperado	La foto se recupera correctamente y se muestra en la siguiente vista.
Resultado obtenido	Correcto

Tabla 8.1: Descripción del CP01

CP02	Obtención de imagen a reconocer desde la galería
Descripción	Un Usuario selecciona la galería para obtener la imagen que será usada en el reconocimiento.
Acción	Ir a la vista principal de la aplicación, pulsar el botón de añadir traducción (+), seleccionar la galería y seleccionar una imagen.
Resultado esperado	La imagen se recupera correctamente y se muestra en la siguiente vista.
Resultado obtenido	Correcto

Tabla 8.2: Descripción del CP02

CP03	Obtención de imagen a reconocer desde la cámara y girando el dispositivo
Descripción	Un Usuario selecciona la cámara para obtener la imagen que será usada en el reconocimiento y realiza la foto en modo apaisado.
Acción	Ir a la vista principal de la aplicación, pulsar el botón de añadir traducción (+), seleccionar la cámara, girar el dispositivo y realizar la foto.
Resultado esperado	La foto se recupera correctamente y se muestra tal y como se ha realizado en la siguiente vista.
Resultado obtenido	Correcto

Tabla 8.3: Descripción del CP03

CP04	Descarga de un idioma de reconocimiento
Descripción	Un Usuario descarga un idioma de reconocimiento.
Acción	Ir a la vista de reconocimiento, abrir la lista de idiomas de reconocimiento y pulsar sobre el botón de descargar un idioma específico.
Resultado esperado	Se inicia la descarga y al finalizar se muestra un mensaje de validación indicando que la descarga del idioma se ha realizado con éxito.
Resultado obtenido	Correcto

Tabla 8.4: Descripción del CP04

CP05	Descarga de un idioma de reconocimiento sin conexión a internet
Descripción	Un Usuario intenta descargar un idioma de reconocimiento sin tener conexión a internet.
Acción	Ir a la vista de reconocimiento, abrir la lista de idiomas de reconocimiento y pulsar sobre el botón de descargar un idioma específico sin tener conexión a internet.
Resultado esperado	Se muestra un mensaje de error indicando que la descarga ha fallado y que se compruebe la conexión a internet.
Resultado obtenido	Correcto

Tabla 8.5: Descripción del CP05

CP06	Interrupción de la descarga de un idioma de reconocimiento por la pérdida repentina de la conexión a internet
Descripción	Un Usuario inicia la descarga de un idioma de reconocimiento pero ésta se ve interrumpida por la pérdida repentina de la conexión a internet.
Acción	Ir a la vista de reconocimiento, abrir la lista de idiomas de reconocimiento y pulsar sobre el botón de descargar un idioma específico. De repente, se pierde la conexión a internet.
Resultado esperado	Se muestra un mensaje de error indicando que se ha interrumpido la descarga y que se compruebe la conexión a internet.
Resultado obtenido	Correcto

Tabla 8.6: Descripción del CP06

CP07	Inicio del reconocimiento de caracteres
Descripción	Un Usuario inicia el reconocimiento de caracteres.
Acción	Ir a la vista de reconocimiento, seleccionar uno de los idiomas de la lista de idiomas de reconocimiento. A continuación, seleccionar la parte de la imagen a reconocer y pulsar el botón de iniciar reconocimiento.
Resultado esperado	Se realiza el reconocimiento y se muestra el texto reconocido en la siguiente vista.
Resultado obtenido	Correcto

Tabla 8.7: Descripción del CP07

CP08	Inicio del reconocimiento seleccionando un idioma no descargado
Descripción	Un Usuario inicia el reconocimiento de caracteres habiendo seleccionado un idioma que no se ha descargado.
Acción	En la vista de reconocimiento, seleccionar de la lista de idiomas de reconocimiento uno que no está descargado. A continuación, seleccionar la parte de la imagen a reconocer y pulsar el botón de iniciar reconocimiento.
Resultado esperado	Se muestra un mensaje de error indicando que el idioma seleccionado no está disponible y que es necesario descargarlo para reconocer texto en ese idioma
Resultado obtenido	Correcto

Tabla 8.8: Descripción del CP08

CP09	Inicio del reconocimiento seleccionando un idioma cuya descarga se ha interrumpido
Descripción	Un Usuario inicia el reconocimiento de caracteres habiendo seleccionado un idioma que no se ha descargado completamente por una pérdida de la conexión a internet o por la interrupción voluntaria de la descarga.
Acción	En la vista de reconocimiento, seleccionar de la lista de idiomas de reconocimiento uno cuya descarga ha sido interrumpida. A continuación, seleccionar la parte de la imagen a reconocer y pulsar el botón de iniciar reconocimiento.
Resultado esperado	Se muestra un mensaje de error indicando que el idioma seleccionado no está disponible y que es necesario descargarlo para reconocer texto en ese idioma.
Resultado obtenido	Primera ejecución: Incorrecto Causa: la aplicación se cierra inesperadamente porque se intenta iniciar el reconocimiento con un fichero de un idioma incompleto. Acciones: comprobar que el idioma seleccionado está descargado en su totalidad. En caso positivo, iniciar el reconocimiento. Segunda ejecución: Correcto

Tabla 8.9: Descripción del CP09

CP10	Obtención de idiomas de llegada para la traducción, excluyendo el idioma del texto reconocido
Descripción	Un Usuario visualiza la lista con los idiomas disponibles a los que traducir el texto reconocido. Esta lista no debería incluir el idioma del texto reconocido ya que no tiene sentido traducir un texto a su mismo idioma.
Acción	En la vista de traducción, abrir la lista desplegable de idiomas de traducción.
Resultado esperado	La lista de idiomas de traducción no incluye el idioma del texto reconocido.
Resultado obtenido	Correcto

Tabla 8.10: Descripción del CP10

CP11	Traducción de un texto
Descripción	Un Usuario realiza la traducción de un texto.
Acción	En la vista de traducción, seleccionar un idioma al que realizar la traducción y pulsar el botón Traducir.
Resultado esperado	El texto de partida se traduce al idioma seleccionado.
Resultado obtenido	Primera ejecución: Incorrecto Causa: se producen errores de autenticación con el API de traducción debido a una configuración errónea. Acciones: corregir la configuración del API y volver a realizar la autenticación. Segunda ejecución: Correcto

Tabla 8.11: Descripción del CP11

CP12	Traducción de un texto con saltos de línea
Descripción	Un Usuario realiza la traducción de un texto que contiene saltos de línea.
Acción	En la vista de traducción, seleccionar un idioma al que realizar la traducción y pulsar el botón Traducir. El texto a traducir contiene varios saltos de línea.
Resultado esperado	El texto resultante de la traducción incluye los saltos de línea del texto de partida.
Resultado obtenido	Primera ejecución: Incorrecto Causa: el API de traducción utilizada no detecta los saltos de línea por lo que el texto resultante se muestra sin saltos de línea. Acciones: detectar los saltos de línea y enviar varias peticiones de traducción al API, uno por cada frase. Segunda ejecución: Correcto

Tabla 8.12: Descripción del CP12

CP13	Guardado de una traducción
Descripción	Un Usuario guarda una traducción
Acción	En la vista de traducción, seleccionar el botón de guardar traducción.
Resultado esperado	Se guarda la traducción en la base de datos y se vuelve a la vista principal
Resultado obtenido	Correcto

Tabla 8.13: Descripción del CP13

CP14	Guardado de una traducción sin haberla realizado
Descripción	Un Usuario intenta guardar una traducción sin haberla realizado.
Acción	En la vista de traducción, seleccionar el botón de guardar traducción sin haber realizado la traducción.
Resultado esperado	Se muestra un mensaje de error indicando que ha habido un error al guardar la traducción y que para guardarla es necesario traducir el texto usando el botón correspondiente.
Resultado obtenido	Correcto

Tabla 8.14: Descripción del CP14

8.2.3.2. CU02. Visualizar traducciones

CP15	Visualizado de las traducciones realizadas
Descripción	Un Usuario visualiza las traducciones que ha añadido al sistema.
Acción	Ir la vista principal al iniciar la aplicación, después de guardar una traducción o al cancelar el proceso de añadir una nueva traducción. En esta vista se muestran las traducciones añadidas al sistema.
Resultado esperado	Se muestra una lista de traducciones en forma de tarjetas, filtrada en función de la fecha y localización seleccionadas en los filtros y ordenada de fecha más próxima a más lejana.
Resultado obtenido	Primera ejecución: Incorrecto Causa: las traducciones recientemente creadas se añaden al final de la lista debido a como se guardan y recuperan de la base de datos. Acciones: invertir la lista de traducciones obtenida de la base de datos. Segunda ejecución: Correcto

Tabla 8.15: Descripción del CP15

8.2.3.3. CU03. Consultar detalle de traducción

CP16	Consulta del detalle de la traducción
Descripción	Un Usuario consulta el detalle de una traducción seleccionada.
Acción	Ir la vista de detalle de traducción pulsando sobre una traducción de la lista mostrada en la vista principal.
Resultado esperado	Se muestra información detallada de la traducción, incluyendo una barra superior con la imagen utilizada en el reconocimiento.
Resultado obtenido	Primera ejecución: Incorrecto Causa: el título de la vista de detalle de traducción no se entiende si la imagen utilizada en el reconocimiento es de un color claro. Acciones: incluir un gradiente que oscurezca la parte de la imagen cuando coincida con el título. Segunda ejecución: Correcto

Tabla 8.16: Descripción del CP16

8.2.3.4. CU04. Borrar traducción

CP17	Borrado de una traducción
Descripción	Un Usuario borra una traducción.
Acción	Ir la vista principal y deslizar una tarjeta de traducción.
Resultado esperado	La traducción desaparece de la lista y se muestra una opción temporal para deshacer el borrado.
Resultado obtenido	Correcto

Tabla 8.17: Descripción del CP17

CP18	Recuperación de traducción borrada
Descripción	Un Usuario deshace el borrado de una traducción.
Acción	Ir la vista principal, deslizar una tarjeta de traducción y pulsar sobre la opción de deshacer que aparece.
Resultado esperado	La traducción vuelve aparecer en la lista y no se borra de la base de datos.
Resultado obtenido	<p>Primera ejecución: Incorrecto</p> <p>Causa: la traducción borrada y posteriormente recuperada aparece dos veces si se pulsa por error el botón deshacer dos veces.</p> <p>Acciones: comprobar si la traducción a recuperar ya se ha añadido a la lista. En caso positivo, no se vuelve a añadir.</p> <p>Segunda ejecución: Correcto</p>

Tabla 8.18: Descripción del CP18

CP19	Borrado definitivo de una traducción
Descripción	Un Usuario borra una traducción y no selecciona la opción de deshacer.
Acción	Ir la vista principal, deslizar una tarjeta de traducción y dejar que la opción de deshacer el borrado desaparezca.
Resultado esperado	La traducción desaparece de la lista y se borra de la base de datos.
Resultado obtenido	Correcto

Tabla 8.19: Descripción del CP19

CP20	Borrado de traducción con filtros
Descripción	Un Usuario borra una traducción sobre una lista filtrada.
Acción	Ir la vista principal, donde se muestra una lista de traducciones filtrada y deslizar una tarjeta de traducción para borrarla. A continuación, quitar los filtros para mostrar la lista completa de traducciones.
Resultado esperado	La lista con todas las traducciones aparece actualizada con los borrados realizados sobre la lista filtrada.
Resultado obtenido	<p>Primera ejecución: Incorrecto</p> <p>Causa: solo se borra la traducción de la lista de traducciones mostrada, no de la lista original de traducciones. Es necesario mantener dos listas de traducciones diferentes, una en el adaptador, que se muestra al usuario, y la lista original de traducciones, que se actualiza con las acciones realizadas sobre la lista que se muestra.</p> <p>Acciones: borrar de la lista original la traducción borrada de la lista mostrada.</p> <p>Segunda ejecución: Correcto</p>

Tabla 8.20: Descripción del CP20

8.2.3.5. CU05. Buscar traducción

CP21	Búsqueda de traducciones
Descripción	Un Usuario busca una traducción.
Acción	Ir la vista principal, pulsar el botón de buscar y escribir una consulta en la barra de búsqueda.
Resultado esperado	Lista de traducciones que se correspondan con el criterio de búsqueda del Usuario.
Resultado obtenido	<p>Primera ejecución: Incorrecto</p> <p>Causa: según se van introduciendo caracteres en la barra de búsqueda llega un momento en el que la lista de traducciones aparece vacía aunque alguna coincida con el criterio de búsqueda utilizado. Esto se debe a que la lista se va filtrando sobre si misma hasta no quedar nada.</p> <p>Acciones: utilizar una copia de la lista de traducciones original, en vez de una referencia a la misma, de tal forma que se distinga la lista de traducciones original de la lista que se va filtrando.</p> <p>Segunda ejecución: Correcto</p>

Tabla 8.21: Descripción del CP21

8.2.3.6. CU06. Filtrar traducción

CP22	Filtrado de traducciones
Descripción	Un Usuario busca una traducción.
Acción	Ir la vista principal, pulsar el botón de filtrar y seleccionar una fecha y localización.
Resultado esperado	Lista de traducciones que se correspondan con los filtros seleccionados.
Resultado obtenido	Correcto

Tabla 8.22: Descripción del CP22

CP23	Opciones de filtrado después de borrar una traducción
Descripción	Un Usuario filtra las traducciones y borra una traducción de forma que ya no exista ninguna que cumpla alguno de los filtros seleccionados.
Acción	Ir la vista principal, pulsar el botón de filtrar, seleccionar una fecha y/o localización y borrar la última traducción que cumpla alguno de los filtros seleccionados.
Resultado esperado	Los filtros afectados se reinician a la opción por defecto, ya que no existe ninguna traducción que los cumplan.
Resultado obtenido	<p>Primera ejecución: Incorrecto</p> <p>Causa: los filtros muestran sus opciones a partir de las fechas y localizaciones de las traducciones guardadas. Ejemplo: el filtro de localizaciones muestra las opciones "ninguna", "Valladolid" y "Madrid" y el usuario selecciona "Valladolid" (posición 1 en la lista de localizaciones). Se muestra una traducción, que el usuario borra, y la opción Valladolid desaparece del filtro de localización. El filtro seleccionado pasa a ser el que ocupa la posición 1, es decir "Madrid". Este filtro no ha sido seleccionado por el usuario. Si no existiera Madrid, la aplicación se detendría, ya que intentaría ponerse la posición 1 de la lista de localizaciones en el filtro, pero esta ya no existe al haberse borrado la única traducción que la cumplía.</p> <p>Acciones: establecer en los filtros afectados la opción "ninguna" cuando la traducción borrada provoque que ya no exista ninguna que cumpla con alguno de los filtros seleccionados.</p> <p>Segunda ejecución: Correcto</p>

Tabla 8.23: Descripción del CP23

Capítulo 9

Conclusiones

9.1. Objetivos alcanzados

Una vez finalizado el proyecto, es interesante enumerar una serie de objetivos que se han alcanzado:

- Se ha realizado un estudio sobre las diferentes bibliotecas de reconocimiento óptico de caracteres y traducción.
- Se ha aprendido a utilizar la biblioteca de OCR Tesseract, implementada en multitud de aplicaciones Android, iOS y páginas web que hacen uso de reconocimiento óptico de caracteres.
- Se ha utilizado y entendido el funcionamiento de Android NDK.
- Se ha aprendido a utilizar una API de traducción, como es la de Microsoft.
- Se han afianzado los conocimientos de Android del alumno.
- Se ha aprendido a utilizar el patrón MVP en Android, permitiendo obtener un código más claro, legible y mantenible gracias a la modularidad de la cual se caracteriza este patrón.
- Se ha diseñado una interfaz basada en Material Design de Android, que actualmente es utilizada por muchas aplicaciones, al ser atractiva e intuitiva.
- Se ha conseguido realizar una aplicación capaz de reconocer caracteres en 39 idiomas y traducir el texto reconocido al mismo número de idiomas, lo cual ha sido un gran logro ya que la idea inicial no contemplaba tantos idiomas.
- Se ha conseguido crear un sistema de descarga y borrado de idiomas de reconocimiento, evitando problemas de memoria al usuario y permitiéndole descargarse los idiomas que vaya a necesitar en su próximo viaje.
- Se ha añadido una funcionalidad que no poseen los traductores convencionales, como es el historial de traducciones, con búsqueda y filtrado por fecha y/o localización.
- Se ha aprendido a utilizar herramientas de prototipado web, que facilitan y agilizan el proceso de diseño de interfaz.
- Se han afianzado los conocimientos del alumno en el modelo de desarrollo ágil Scrum, actualmente muy utilizado en el entorno empresarial.
- Se ha planificado un proyecto software y se han gestionado los riesgos producidos.
- Se han afianzado los conocimientos de LaTeX del alumno, elaborando la memoria del proyecto.

9.2. Valoración personal

La realización de este TFG me ha permitido adquirir nuevos conocimientos y afianzar los que ya había obtenido, tanto en el Grado de Ingeniería Informática como por mi cuenta.

Antes de iniciar el proyecto, se presentaba como un trabajo complicado ya que sería necesario utilizar técnicas de reconocimiento óptico de caracteres y bibliotecas nativas. Según iba avanzando el proyecto, surgieron problemas relacionados con la biblioteca de reconocimiento pero fueron solucionados con éxito.

Durante el desarrollo del proyecto ha sido necesario realizar cambios para adaptarse a las necesidades que iban surgiendo. Al hacer uso de una metodología como es Scrum, me ha sido relativamente sencillo gestionar los cambios y replanificar.

Por otra parte, las técnicas OCR han avanzado considerablemente en los últimos años y cada vez se utilizan más en multitud de aplicaciones. No obstante, aún falta mucho camino por recorrer y estoy convencido de que serán algo cotidiano en un futuro no muy lejano.

En definitiva, se han alcanzado los objetivos propuestos y el resultado del proyecto ha sido satisfactorio.

9.3. Líneas futuras

Como trabajo futuro se podrían realizar los siguientes puntos:

- Añadir reconocimiento y traducción a un mayor número de idiomas.
- Mejorar el reconocimiento óptico de caracteres.
- Mejorar la traducción realizada.
- Añadir dictados de voz de los textos traducidos.
- Permitir descargar paquetes de idiomas en función de la región a visitar. Por ejemplo: paquete Europa Occidental con los idiomas inglés, francés, alemán, español, italiano u otros.
- Mejorar algunos aspectos de la interfaz gráfica e incluso añadir animaciones en algunas vistas de la misma.
- Añadir la posibilidad de compartir traducciones con redes sociales.

Bibliografía

- [1] Wikipedia. «Optical character recognition». [En línea]. Disponible en: https://en.wikipedia.org/wiki/Optical_character_recognition [Accedido: 05-Febrero-2016].
- [2] ——. «Scrum (desarrollo de software)». [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Scrum_\(desarrollo_de_software\)](https://es.wikipedia.org/wiki/Scrum_(desarrollo_de_software)) [Accedido: 08-Febrero-2016].
- [3] Dropbox. «Dropbox funciona como tú». [En línea]. Disponible en: <https://www.dropbox.com/> [Accedido: 09-Febrero-2016].
- [4] Github. «How people build software». [En línea]. Disponible en: <http://github.com/> [Accedido: 09-Febrero-2016].
- [5] ShareLaTeX. «The easy to use, online, collaborative LaTeX editor». [En línea]. Disponible en: <https://www.sharelatex.com/> [Accedido: 09-Febrero-2016].
- [6] Wikipedia. «Desarrollo en cascada». [En línea]. Disponible en: https://es.wikipedia.org/wiki/Desarrollo_en_cascada [Accedido: 10-Febrero-2016].
- [7] ——. «Proceso unificado». [En línea]. Disponible en: https://es.wikipedia.org/wiki/Proceso_unificado [Accedido: 10-Febrero-2016].
- [8] Scrum Manager. «Gestión de proyectos Scrum Manager». [En línea]. Disponible en: http://www.scrummanager.net/files/sm_proyecto.pdf [Accedido: 11-Febrero-2016].
- [9] Wolf Paulus. «Android and OCR». [En línea]. Disponible en: <https://wolfpaulus.com/journal/android-journal/android-and-ocr/> [Accedido: 12-Febrero-2016].
- [10] Wikipedia. «Tesseract OCR». [En línea]. Disponible en: https://es.wikipedia.org/wiki/Tesseract_OCR [Accedido: 12-Febrero-2016].
- [11] Robert Theis. «tess-two». [En línea]. Disponible en: <https://github.com/rmtheis/tess-two> [Accedido: 15-Febrero-2016].
- [12] Free OCR. «Bienvenido a Free OCR». [En línea]. Disponible en: <http://www.free-ocr.com/es.html> [Accedido: 15-Febrero-2016].
- [13] Apertium. «Plataforma libre para la traducción automática». [En línea]. Disponible en: <https://www.apertium.org> [Accedido: 15-Febrero-2016].
- [14] Microsoft. «Microsoft Translator Text API». [En línea]. Disponible en: <https://www.microsoft.com/en-us/translator/translatorapi.aspx> [Accedido: 17-Febrero-2016].
- [15] Proyectos Ágiles. «Cliente (Product Owner)». [En línea]. Disponible en: <https://proyectosagiles.org/cliente-product-owner/> [Accedido: 19-Febrero-2016].
- [16] Elaine M. Hall, «*Managing Risk: Methods for Software Systems Development*». Addison-Wesley, 1998.
- [17] Antonio Leiva. «MVP for Android: how to organize the presentation layer». [En línea]. Disponible en: <http://antonioleiva.com/mvp-android/> [Accedido: 4-Abril-2016].
- [18] Yania Crespo González, *Apuntes de la asignatura Diseño de Software del Grado en Ingeniería Informática de la Universidad de Valladolid. Tema 5. Patrones de diseño-Singleton*. Universidad de Valladolid, 2014-2015.
- [19] —, *Apuntes de la asignatura Diseño de Software del Grado en Ingeniería Informática de la Universidad de Valladolid. Tema 5. Patrones de acceso a datos. Más*. Universidad de Valladolid, 2014-2015.

- [20] NinjaMock. «Slice your work in half». [En línea]. Disponible en: <http://ninjamock.com/> [Accedido: 20-Abril-2016].
- [21] Robert Theis. «Building Tess Two». [En línea]. Disponible en: <https://github.com/rmtheis/tess-two/blob/master/BUILDING.md> [Accedido: 15-Febrero-2016].
- [22] Taiga. «Love your project». [En línea]. Disponible en: <https://taiga.io> [Accedido: 17-Febrero-2016].
- [23] Wikipedia. «Experiencia de usuario». [En línea]. Disponible en: https://es.wikipedia.org/wiki/Experiencia_de_usuario [Accedido: 17-Febrero-2016].
- [24] ——. «Front-end y back-end». [En línea]. Disponible en: https://es.wikipedia.org/wiki/Front-end_y_back-end [Accedido: 17-Febrero-2016].
- [25] Taiga Blog. «User stories demystified – User Story Points». [En línea]. Disponible en: <https://blog.taiga.io/user-stories-demystified.html> [Accedido: 18-Febrero-2016].
- [26] ——. «User stories demystified – User stories points vs man hours». [En línea]. Disponible en: <https://blog.taiga.io/user-stories-demystified.html> [Accedido: 18-Febrero-2016].
- [27] Github. «tesseract-ocr/tessdata». [En línea]. Disponible en: <https://github.com/tesseract-ocr/tessdata> [Accedido: 8-Marzo-2016].
- [28] Desmond Shaw. «Making an Android OCR Application with Tesseract». [En línea]. Disponible en: <http://www.codepool.biz/making-an-android-ocr-application-with-tesseract.html> [Accedido: 9-Marzo-2016].
- [29] Arthur Hub. «ArthurHub/Android-Image-Cropper». [En línea]. Disponible en: <https://github.com/ArthurHub/Android-Image-Cropper> [Accedido: 24-Marzo-2016].
- [30] Github. «Tesseract OCR-TessData». [En línea]. Disponible en: <https://github.com/tesseract-ocr/tessdata> [Accedido: 25-Marzo-2016].
- [31] Prateek Srivastava. «ProgressBar». [En línea]. Disponible en: <https://github.com/f2prateek/progressbar> [Accedido: 28-Abril-2016].
- [32] Android Developers. «Getting the Last Known Location». [En línea]. Disponible en: <https://developer.android.com/training/location/retrieve-current.html> [Accedido: 18-Mayo-2016].
- [33] Xaver Kapeller. «How to filter a recyclerview with a searchview». [En línea]. Disponible en: <http://stackoverflow.com/questions/30398247/how-to-filter-a-recyclerview-with-a-searchview> [Accedido: 6-Junio-2016].

Apéndice A

Glosario de términos

- **AAR**: distribución binaria de un proyecto Android. Son ficheros comprimidos que contienen, precompilados, todos los archivos que incluiría un proyecto Android como tal. Es similar a un fichero .jar, solo que los AAR también incluyen componentes visuales como layouts o drawables.
- **Android**: sistema operativo basado en el núcleo Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tablets o tablefonos; y también para relojes inteligentes, televisores y automóviles.
- **Android NDK**: conjunto de herramientas que permiten al desarrollador implementar partes de una aplicación Android utilizando código escrito en lenguajes nativos como C o C++.
- **API**: siglas de “interfaz de programación de aplicaciones”. Conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.
- **Asistev**: acrónimo de “asistente Android para traducción de textos escritos en viajes”.
- **COTS**: acrónimo de “Commercial Off-The-Shelf” o producto de caja en español. Un COTS es un producto manufacturado estándar en lugar de personalizado. COTS típicamente requiere una configuración que se adapta para usos específicos.
- **Eclipse**: conjunto de herramientas de programación de código abierto multiplataforma para el desarrollo de aplicaciones.
- **Fork**: creación de un proyecto en una dirección distinta de la principal u oficial tomando el código fuente del proyecto ya existente.
- **HTML5**: quinta revisión importante del lenguaje básico de la World Wide Web, HTML. HTML5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos.
- **IDE**: siglas de “Integrated Development Environment” o entorno de desarrollo integrado en español. Es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.
- **LaTeX**: sistema de composición de textos, orientado a la creación de documentos escritos que presenten una alta calidad tipográfica.
- **Leptonica**: software de código abierto ampliamente utilizado para aplicaciones de análisis y procesamiento de imágenes.
- **Licencia Apache**: licencia de software libre permisiva creada por la Apache Software Foundation (ASF).
- **Material Design**: normativa de diseño de Google enfocada en la visualización del sistema operativo Android y aplicable a la web y cualquier otra plataforma. Material se trata de un diseño más limpio, en el que predominan animaciones y transiciones de respuesta, el relleno y los efectos de profundidad tales como la iluminación y las sombras.
- **Pila del producto o Product Backlog**: lista de los requisitos desde el punto de vista del cliente. Está formada por las funcionalidades o historias de usuario que desea obtener el cliente, ordenadas por la prioridad que el mismo le otorga a cada una.

- **Scrum:** modelo de desarrollo ágil basado en una colaboración estrecha con el cliente y una predisposición y respuesta ante los cambios.
- **Sprint:** período de un proyecto Scrum en el cual se lleva a cabo el trabajo en sí. Es recomendado que la duración de los sprints sea constante y definida por el equipo en base a su propia experiencia (2 o 3 semanas).
- **SQLite:** sistema de gestión de bases de datos relacional compatible con ACID, contenida en una biblioteca relativamente pequeña (275KB) escrita en C.
- **Stack Overflow:** sitio web utilizado por una comunidad de desarrolladores informáticos, en la cual otros desarrolladores pueden encontrar soluciones a problemas de programación en diferentes lenguajes.
- **T:** tarea. Representa las tareas en las que se divide cada una de las historias de usuario, seguidas de un identificador.
- **Tesseract Android Tools:** conjunto de herramientas que permiten compilar las bibliotecas Tesseract y Leptonica y utilizarlas en la plataforma Android. Proporciona un API de Java que permite acceder a las APIs nativamente compiladas de Tesseract y Leptonica.
- **TFG:** siglas de "trabajo fin de grado". Proyecto o trabajo exigido como condición para obtener finalmente una titulación universitaria.
- **US:** user story. Representan las historias de usuario o user stories, seguidas de un identificador.

Apéndice B

Manual de usuario

B.1. Introducción

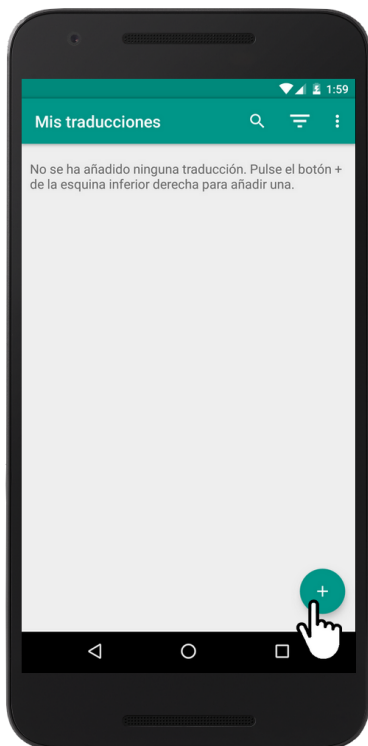
Este documento detalla la forma de utilizar la aplicación *Asistev* por parte de un usuario.

Se detallará cada una de las funcionalidades de la aplicación junto con una serie de imágenes ilustrativas.

B.2. Funcionalidades

B.2.1. Añadir traducción

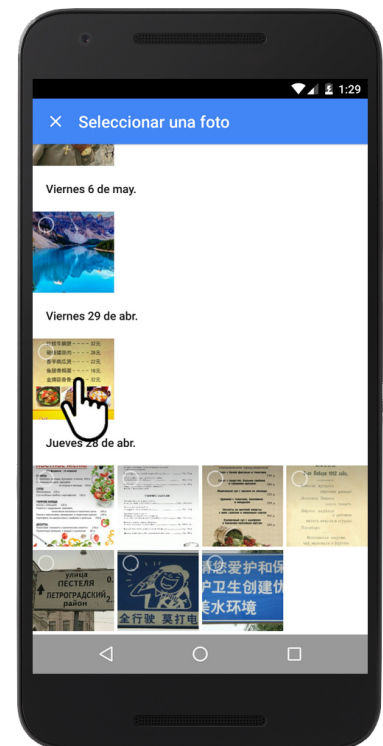
B.2.1.1. Paso 1, alternativa 1: Obtener imagen de la galería



(a) Añadir nueva traducción



(b) Seleccionar la opción de la galería



(c) Seleccionar imagen de la galería

Figura B.1: Instrucciones para obtener imagen de la galería

B.2.1.2. Paso 1, alternativa 2: Obtener imagen desde la cámara



Figura B.2: Instrucciones para obtener imagen desde la cámara

B.2.1.3. Paso 2: Descargar idioma de reconocimiento



Figura B.3: Instrucciones para descargar un idioma de reconocimiento

B.2.1.4. Paso 2.1 (opcional): Borrar idioma de reconocimiento



(a) Borrar idioma de reconocimiento



(b) Idioma de reconocimiento borrado

Figura B.4: Instrucciones para borrar un idioma de reconocimiento

B.2.1.5. Paso 3: Realizar el reconocimiento óptico de caracteres



(a) Seleccionar idioma de reconocimiento



(b) Seleccionar la parte de la imagen a reconocer



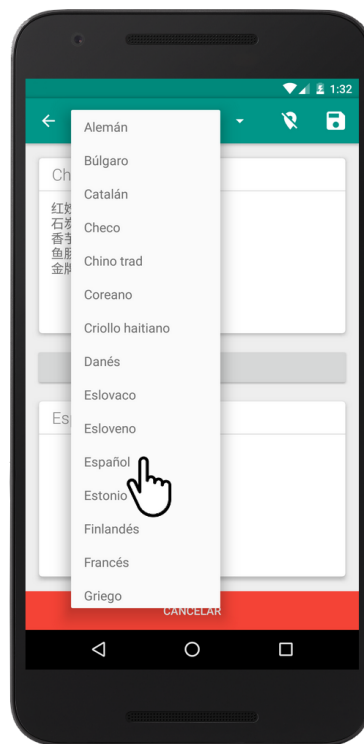
(c) Iniciar reconocimiento de caracteres

Figura B.5: Instrucciones para realizar el reconocimiento óptico de caracteres

B.2.1.6. Paso 4: Realizar la traducción



(a) Abrir lista de idiomas de traducción



(b) Seleccionar idioma al que traducir el texto



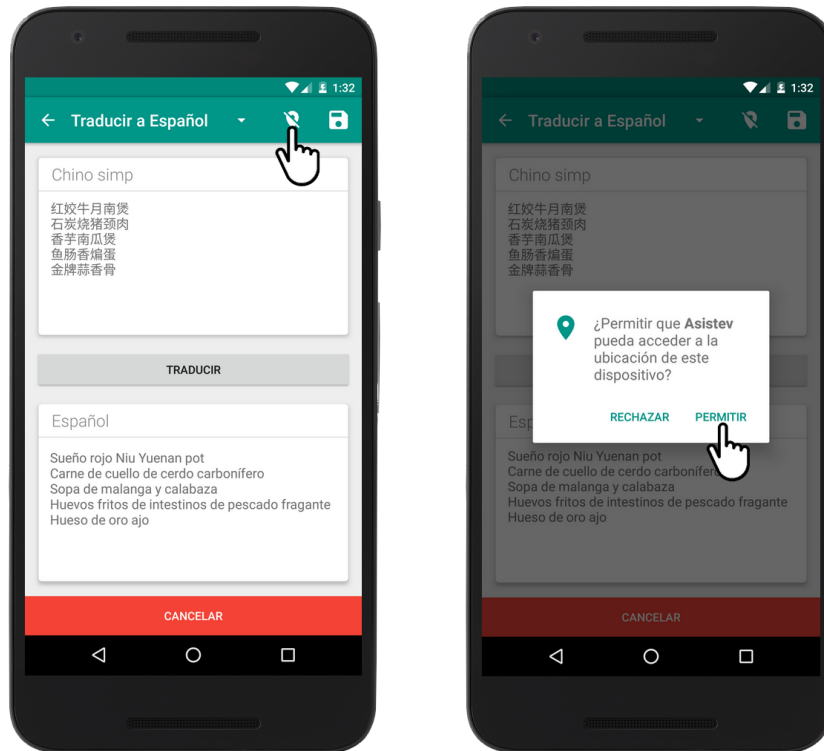
(c) Traducir



(d) Texto traducido correctamente

Figura B.6: Instrucciones para realizar la traducción

B.2.1.7. Paso 5: Guardar la traducción



(a) Incluir localización en la traducción (opcional)

(b) Aceptar permisos de localización (opcional)



(c) Guardar traducción

Figura B.7: Instrucciones para guardar la traducción

B.2.2. Visualizar traducciones

La siguiente vista se muestra después de las siguientes acciones:

- Abrir la aplicación.
- Guardar una traducción. Ver figura B.7c.
- Volver atrás desde el detalle de una traducción.
- Cancelar en cualquier punto cuando se está añadiendo una traducción. Para cancelar, pulsar el botón rojo en la barra inferior de la app.



Figura B.8: Visualizar traducciones

B.2.3. Consultar detalle de traducción



(a) Seleccionar traducción

(b) Detalles de la traducción



(c) Detalles de la traducción al deslizar hacia arriba

Figura B.9: Instrucciones para consultar el detalle de una traducción

B.2.4. Borrar traducción

B.2.4.1. Deshaciendo el borrado



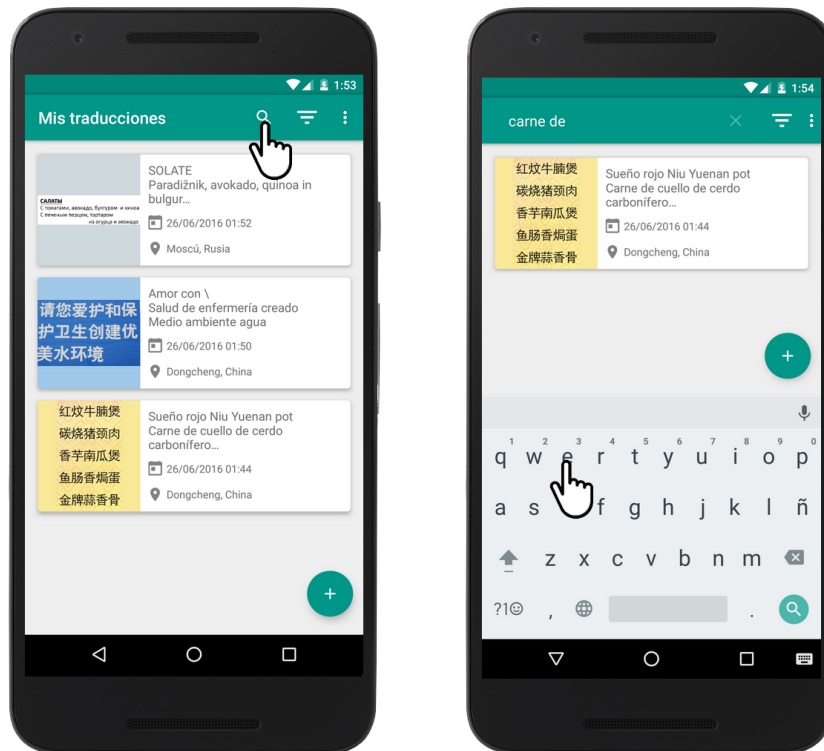
Figura B.10: Instrucciones para borrar una traducción y deshacer su borrado

B.2.4.2. Sin deshacer el borrado



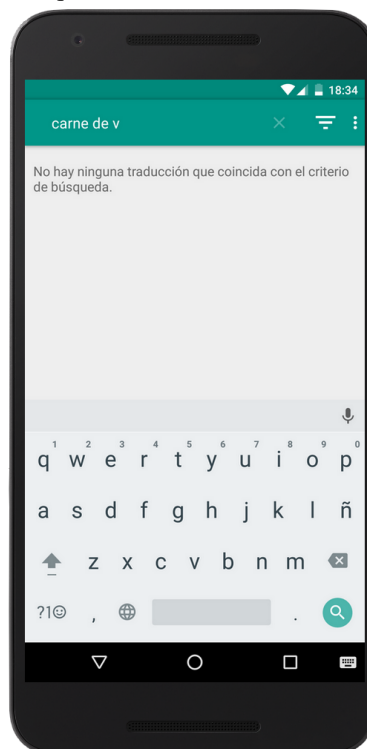
Figura B.11: Instrucciones para borrar una traducción definitivamente

B.2.5. Buscar traducción



(a) Abrir barra de búsqueda

(b) Escribir criterio de búsqueda



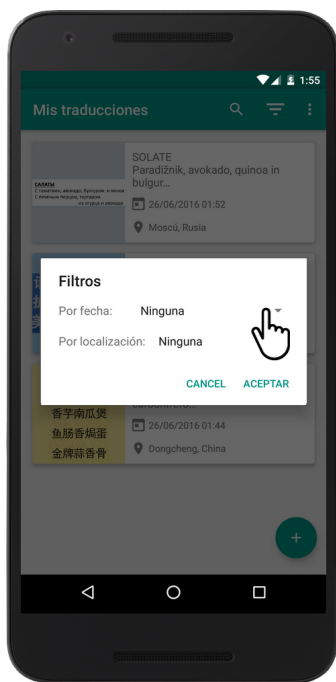
(c) No existen traducciones coincidentes

Figura B.12: Instrucciones para buscar traducciones

B.2.6. Filtrar traducciones



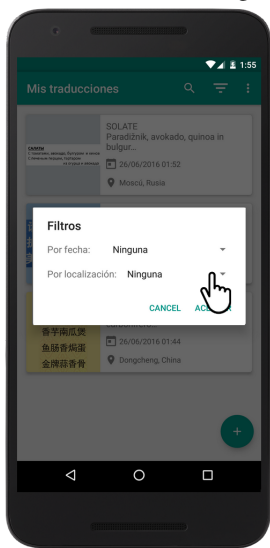
(a) Abrir diálogo de filtros



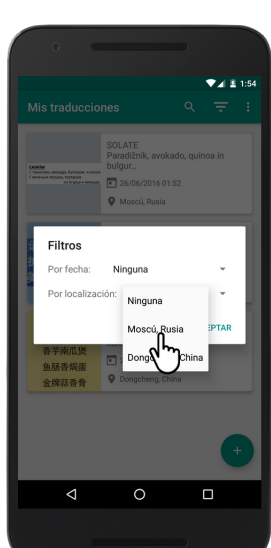
(b) Abrir lista de fechas



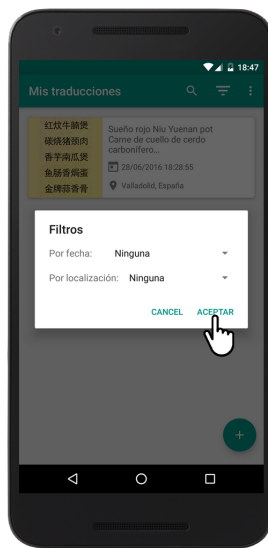
(c) Seleccionar "Ninguna"



(d) Abrir lista de localizaciones



(e) Seleccionar "Moscú"



(f) Aceptar filtros seleccionados



(g) Traducciones filtradas

Figura B.13: Instrucciones para filtrar traducciones

Apéndice C

Plan de seguimiento

C.1. Introducción

Una de las partes fundamentales para que un proyecto alcance su cometido cumpliendo con requisitos de tiempo, alcance y calidad, es el seguimiento de todo trabajo llevado a cabo durante el mismo.

La elaboración de este documento está apoyada en Taiga, una plataforma open source de gestión de proyectos ágiles [22]. No obstante, se han añadido más detalles de seguimiento de los que la propia plataforma ofrece.

Para entender el plan de trabajo seguido en este proyecto es necesario introducir unos conceptos fundamentales de Scrum. Para ello, usaremos la figura C.1, que representa el flujo de trabajo de la metodología Scrum.

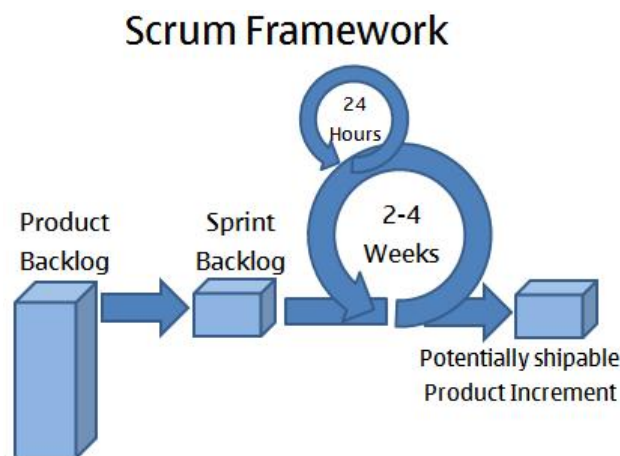


Figura C.1: Estructura de Scrum

Como podemos ver en la figura C.1, la forma de trabajar con Scrum contiene los siguientes pasos:

1. Se parte del *Product Backlog* o pila del producto, que registra una lista de los requisitos desde el punto de vista del cliente y está formada por las funcionalidades o historias de usuario que desea obtener el cliente, ordenadas por la prioridad que el mismo le otorga a cada una.
2. De esta pila, se seleccionan las historias de usuario de mayor prioridad para empezar una iteración o *sprint*.
3. Una vez seleccionadas todas las historias de usuario a desarrollar en el siguiente sprint, se descompone cada una de ellas en tareas a llevar a cabo por el equipo de desarrollo. Esta lista de tareas es lo que forma el *Sprint Backlog* o pila del sprint.
4. Se inicia una iteración o sprint, que dura entorno a 2-4 semanas, en las cuales se trabaja sobre todas las tareas de la pila del sprint.
5. Al inicio de cada jornada se realiza una reunión breve conocida como *daily* donde se revisa en conjunto el trabajo realizado por cada miembro el día anterior, y el previsto para el día en curso.
6. Cada ciclo de desarrollo o iteración (sprint) finaliza con la entrega de una parte operativa del producto o incremento.

C.1.1. Propósito

Este documento incluye el detalle de las diferentes iteraciones o sprints que se llevarán a cabo durante el desarrollo del proyecto. En nuestro caso, se han estimado 7 sprints con una duración de 3 semanas cada uno, exceptuando el último que posee 4 días a mayores. En cada semana se trabajará únicamente de lunes a viernes.

Para cada sprint, se identificarán las historias de usuario que se van a desarrollar, así como las tareas a realizar para completar cada una de ellas.

Se realizarán estimaciones de esfuerzo basadas en:

- **Puntos de historias de usuario:** estimación del tamaño o complejidad de las historias de usuario. A mayor complejidad y tamaño de la historia, más puntos. Para cada historia de usuario, se asignan puntos a los siguientes aspectos.
 - **UX (User Experience):** factores y elementos relativos a la interacción del usuario con la aplicación, cuyo resultado es la generación de una percepción positiva o negativa acerca de la aplicación en sí. [23]
 - **Design:** diseño de la interfaz de usuario y flujo de navegación entre las pantallas de la aplicación.
 - **Front:** parte del software que interactúa con los usuarios, responsable de recolectar los datos de la entrada del usuario y transformarlos según las especificaciones que demanda el back-end para procesarlos y devolver una respuesta. Representa la capa de presentación. [24]
 - **Back:** parte del software que procesa la entrada desde el front-end. Representa la capa de acceso a datos. [24]
 - **Documentation:** incluye todo lo relativo a documentación del proyecto.
- **Puntos de tareas:** subdivisión de los puntos de historias de usuario en las diferentes tareas a realizar para completar cada una de ellas.

Una vez realizados 2 o 3 sprints, estos puntos se utilizan para estimar los esfuerzos que el equipo es capaz de realizar en cada sprint. Considerando el siguiente ejemplo [25]:

- **Sprint 1:** 16 puntos con 1 historia de usuario.
- **Sprint 2:** 14 puntos con 2 historias de usuario.
- **Sprint 3:** 15 puntos de usuario con 2 historias de usuario.

La media de los puntos de historias de usuario realizados en los 3 sprints es 15. Si se estima que en el sprint 4 se van a utilizar 25 puntos, esta media nos indica que será necesario otro sprint, ya que la *velocidad media del equipo* es inferior, de 15 puntos por sprint en este caso.

C.1.1.1. Puntos de historias de usuario vs horas/hombre

En las metodologías tradicionales y en algunas metodologías ágiles es muy habitual el uso de las horas/hombre como medida de estimación del esfuerzo necesario para finalizar una tarea. No obstante, cada vez es más frecuente encontrar proyectos en los que se sustituye esta medida por los *puntos de historia de usuario*. Esto se debe a que las personas no solemos ser muy buenas estimando tiempo y menos cuando carecemos de la experiencia necesaria para basarnos en proyectos anteriores y realizar estimaciones realistas. [26]

La naturaleza del desarrollo, especialmente software, es que gran parte del tiempo estimado se utiliza depurando para corregir fallos. Los puntos de historias de usuario ofrecen otro punto de vista, en el que solo es necesario estimar la complejidad de la tarea y el esfuerzo se calcula posteriormente a través de la velocidad media del equipo, como se ha explicado en el apartado anterior.

C.1.2. Alcance

El seguimiento de las tareas se realiza durante todas las etapas del proyecto. Durante todo el desarrollo del proyecto se anotan las tareas realizadas, las tareas a realizar y los comentarios correspondientes.

Este documento permite obtener una visión clara y detallada de todo lo que sucede en el proyecto y en qué momento.

C.1.3. Códigos de colores

- Historia de usuario a realizar en el sprint actual.

C.2. Sprint 0: 1 de febrero – 21 de febrero 2016

Este primer sprint o sprint 0 adquiere este nombre porque no forma parte de la implementación de la funcionalidad de la aplicación, sino más bien de la preparación de la documentación y arquitectura necesaria para el desarrollo del proyecto.

C.2.1. Pila del producto

ID	Prioridad	Descripción
US01	Muy alta	Como Product Owner quiero tener disponible la documentación inicial del proyecto para poder tener una visión global del mismo.
US06	Muy alta	Como Usuario quiero que se reconozcan los caracteres contenidos en una imagen.
US07	Muy alta	Como Usuario quiero realizar una foto con la cámara del dispositivo, que posteriormente será utilizada en el reconocimiento de caracteres.
US08	Alta	Como Usuario quiero seleccionar una imagen de la galería de fotos, que posteriormente será utilizada en el reconocimiento de caracteres.
US09	Alta	Como Usuario quiero seleccionar en qué parte de la imagen se encuentra el texto a reconocer.
US10	Alta	Como Usuario quiero que se traduzcan los caracteres reconocidos de una imagen, para poder entender el contenido del texto.
US11	Media	Como Usuario quiero seleccionar un idioma de reconocimiento.
US12	Media	Como Usuario quiero seleccionar un idioma al que traducir el texto reconocido.
US13	Baja	Como Usuario deseo disponer del manual de usuario.

Tabla C.1: Pila del producto inicial del Sprint 0

C.2.2. US01 - Como product owner quiero tener disponible la documentación inicial del proyecto para poder tener una visión global del proyecto

C.2.2.1. Descripción

La documentación se va desarrollando a lo largo de todo el proyecto. En esta historia de usuario se va a realizar la documentación inicial del proyecto.

C.2.2.2. Puntos

- UX: -
- Design: -
- Front: -
- Back: -
- Documentation: 14

C.2.2.3. Tareas (Pila del sprint)

- **T02** - Búsqueda, valoración y selección de bibliotecas o servicios OCR y de traducción. – 3 puntos
- **T03** - Plan de Desarrollo del Software. – 4 puntos
- **T04** - Especificación de requisitos y modelo de análisis: definición de actores, diagrama inicial de casos de uso, especificación de casos de uso – 4 puntos
- **T05** - Plan de gestión de riesgos: introducción e identificación de riesgos. – 3 puntos

C.2.2.4. Comentarios

En T02, la selección del API de traducción se ha demorado ya que no ha sido sencillo encontrar un servicio de traducción “gratuito” y de calidad. Se han realizado pruebas con diferentes servicios de traducción para comprobar los idiomas disponibles, así como la calidad de las traducciones.

C.2.3. Revisión de Sprint

C.2.3.1. Incremento

- Puntos de historias de usuario planificados: **14**
- Puntos de historias de usuario realizados: **14**
- Velocidad media: **14 puntos/sprint**

En este sprint 0 no ha habido un incremento de funcionalidad como tal ya que aún no hay “software funcionando”.

En este sprint se ha elaborado el Plan de Desarrollo del Software, el cual ha conllevado una tarea de búsqueda intensiva de información sobre diferentes alternativas para realizar el reconocimiento de caracteres y la traducción.

Por otra parte, también se han establecido los requisitos iniciales del proyecto, los actores y los casos de uso iniciales.

Por último, se ha realizado la identificación de riesgos y se ha comenzado a redactar el presente documento.

C.2.3.2. Riesgos

En este sprint no se ha disparado ningún riesgo.

C.3. Sprint 1: 22 de febrero – 10 de marzo 2016

C.3.1. Pila del producto

ID	Prioridad	Descripción
US14	Muy alta	Como Product Owner quiero tener un modelo de dominio de la aplicación y un plan de gestión de riesgos, que garantice un control de los riesgos y las medidas de corrección oportunas.
US06	Muy alta	Como Usuario quiero que se reconozcan los caracteres contenidos en una imagen.
US07	Muy alta	Como Usuario quiero realizar una foto con la cámara del dispositivo, que posteriormente será utilizada en el reconocimiento de caracteres.
US08	Alta	Como Usuario quiero seleccionar una imagen de la galería de fotos, que posteriormente será utilizada en el reconocimiento de caracteres.
US09	Alta	Como Usuario quiero seleccionar en qué parte de la imagen se encuentra el texto a reconocer.
US10	Alta	Como Usuario quiero que se traduzcan los caracteres reconocidos de una imagen, para poder entender el contenido del texto.
US11	Media	Como Usuario quiero seleccionar un idioma de reconocimiento.
US12	Media	Como Usuario quiero seleccionar un idioma al que traducir el texto reconocido.
US13	Baja	Como Usuario deseo disponer del manual de usuario.

Tabla C.2: Pila del producto inicial del Sprint 1

En este sprint se ha incluido una nueva historia de usuario US14 y se ha priorizado por encima de todas las demás. Esto se debe a que es necesario tener parte de la documentación de análisis y un plan de gestión de riesgos listo antes de comenzar con el desarrollo.

C.3.2. US14 - Como Product Owner quiero tener un modelo de dominio de la aplicación y un plan de gestión de riesgos completo, que garantice un control de los riesgos y las medidas de corrección oportunas

C.3.2.1. Descripción

Se continúa elaborando la documentación del proyecto.

C.3.2.2. Puntos

- UX: -
- Design: -
- Front: -
- Back: -
- Documentation: 9

C.3.2.3. Tareas (Pila del sprint)

- **T15** - Modelo de análisis: modelo de dominio. – 4 puntos
- **T16** - Plan de gestión de riesgos: análisis, impacto y planes de acción. – 5 puntos

C.3.2.4. Comentarios

En T15 han aparecido dificultades al realizar el modelo de dominio, ya que es necesario tener un conocimiento más detallado del funcionamiento de las bibliotecas OCR y traducción para realizar el modelo definitivo. No obstante, se ha realizado una versión preliminar del modelo (2 puntos), que será actualizada en futuros sprints.

La finalización del modelo de dominio se aplaza a otro sprint.

C.3.3. US06 - Como Usuario quiero que se reconozcan los caracteres contenidos en una imagen

C.3.3.1. Descripción

Reconocer los caracteres incluidos en una imagen mediante el uso de técnicas OCR y transformarlos a formato texto.

Para esta historia de usuario se utilizará una imagen guardada en el propio proyecto Android, aún no se usará la cámara o la galería para obtener la imagen.

C.3.3.2. Puntos

- **UX:** -
- **Design:** -
- **Front:** 7
- **Back:** -
- **Documentation:** -

C.3.3.3. Tareas (Pila del sprint)

- **T17** - Implementar biblioteca OCR Tesseract en el proyecto Android. – 4 puntos
- **T18** - Realizar el reconocimiento óptico de caracteres de una imagen por defecto. – 2 puntos
- **T19** - Validar los caracteres reconocidos. – 1 punto

C.3.3.4. Comentarios

Implementar la biblioteca OCR Tesseract en el proyecto Android ha necesitado varios intentos, pero finalmente se ha conseguido realizar con éxito. Todos las bibliotecas y ficheros necesarios para realizar OCR han sido añadidos al proyecto y a partir de ahora, ya no será necesario compilarlas cada vez que se ejecute la aplicación.

También se ha conseguido realizar el reconocimiento óptico de caracteres contenidos en una imagen guardada dentro del proyecto Android. El reconocimiento se ha realizado con varias palabras en inglés, siendo necesario descargarse el fichero del idioma correspondiente, en este caso “eng.traineddata”. [27]

C.3.4. Revisión de Sprint

C.3.4.1. Incremento

- Puntos de historias de usuario planificados: **16**
- Puntos de historias de usuario realizados: **14**
- Velocidad media: **14 puntos/sprint**

En este sprint sí que ha habido un incremento de funcionalidad ya que el reconocimiento de caracteres ya funciona con una imagen predefinida que contenga palabras en inglés. Para que OCR funcione rápido y correctamente, la imagen debe contener el texto perfectamente centrado, sin ningún margen adicional.

También se ha continuado con la elaboración de la documentación, en este caso el análisis e impacto de los riesgos y los planes de acción.

Por otro lado, no ha sido posible finalizar el modelo de dominio completo, lo que ha llevado a acortar la duración del sprint en un día, siendo la fecha fin del sprint el jueves 10 de marzo.

C.3.4.2. Riesgos

Se ha producido el riesgo **R04**, que trata de la falta de experiencia en el uso de bibliotecas OCR, en este caso Tesseract.

El riesgo se ha producido porque implementar Tesseract y conseguir que funcione no ha sido tarea fácil y ha habido momentos de incertidumbre y muchas dudas debidas a la falta de experiencia en la materia. Para solucionar esto, se ha llevado a cabo el plan de acción correctora de consultar documentación, tutoriales, foros especializados y ejemplos sobre el uso de la biblioteca Tesseract. [28]

C.4. Sprint 2: 11 de marzo – 31 de marzo 2016

C.4.1. Pila del producto

ID	Prioridad	Descripción
US07	Muy alta	Como Usuario quiero realizar una foto con la cámara del dispositivo, que posteriormente será utilizada en el reconocimiento de caracteres.
US08	Muy alta	Como Usuario quiero seleccionar una imagen de la galería de fotos, que posteriormente será utilizada en el reconocimiento de caracteres.
US09	Muy alta	Como Usuario quiero seleccionar en qué parte de la imagen se encuentra el texto a reconocer.
US10	Alta	Como Usuario quiero que se traduzcan los caracteres reconocidos de una imagen, para poder entender el contenido del texto.
US11	Media	Como Usuario quiero seleccionar un idioma de reconocimiento.
US12	Media	Como Usuario quiero seleccionar un idioma al que traducir el texto reconocido.
US13	Baja	Como Usuario deseo disponer del manual de usuario.

Tabla C.3: Pila del producto inicial del Sprint 2

C.4.2. US07 - Como Usuario quiero realizar una foto con la cámara del dispositivo, que posteriormente será utilizada en el reconocimiento de caracteres

C.4.2.1. Descripción

Se considera una de las historias de usuario más importantes del proyecto, ya que su implementación permitirá al usuario hacer fotos durante sus viajes para poder traducir el texto contenido en ellas.

También se permitirá que el usuario pueda utilizar una aplicación de cámara de terceros para capturar la imagen, no es necesario que use la que viene por defecto.

C.4.2.2. Puntos

- **UX:** -
- **Design:** 0.5
- **Front:** 2.5
- **Back:** -
- **Documentation:** -

C.4.2.3. Tareas (Pila del sprint)

- **T20** - Diseñar y crear una interfaz básica que contenga un botón para lanzar la cámara. – 0.5 puntos
- **T21** - Lanzar un intent que abra la aplicación de la cámara del dispositivo u otras aplicaciones de terceros. – 1 punto
- **T22** - Recibir el resultado del intent de la cámara. – 0.5 puntos
- **T23** - Obtener la imagen del intent. – 1 punto

C.4.2.4. Comentarios

La captura de imágenes desde la cámara del dispositivo no se producía correctamente en algún caso concreto, principalmente cuando se giraba la cámara en varias ocasiones. Esto se solucionó creando una clase específica que tratara la captura de las imágenes, la rotación y otros aspectos.

C.4.3. US08 - Como usuario quiero seleccionar una imagen de la galería, que posteriormente será utilizada en el reconocimiento de caracteres

C.4.3.1. Descripción

A pesar de que la mayoría de captura de imágenes se hará probablemente desde la cámara, es interesante que el usuario pueda elegir una foto realizada anteriormente o una imagen de la galería.

C.4.3.2. Puntos

- **UX:** -
- **Design:** 0.5
- **Front:** 2.5
- **Back:** -
- **Documentation:** -

C.4.3.3. Tareas (Pila del sprint)

- **T24** - Diseñar y crear una interfaz básica que contenga un botón para abrir la galería de imágenes del dispositivo. – 0.5 puntos
- **T25** - Lanzar un intent que abra la aplicación de la galería del dispositivo u otras aplicaciones de terceros. – 1 punto
- **T26** - Recibir el resultado del intent de la galería de imágenes. – 0.5 puntos
- **T27** - Obtener la imagen del intent. – 1 punto

C.4.3.4. Comentarios

También se permite que el usuario pueda seleccionar una imagen desde cualquier aplicación de álbum o galería de fotos de terceros.

C.4.4. US09 - Como Usuario quiero seleccionar en qué parte de la imagen se encuentra el texto a reconocer

C.4.4.1. Descripción

Para realizar un reconocimiento óptico de caracteres lo más rápido y fiable posible, es fundamental que el texto a reconocer se encuentre centrado en la imagen y que la foto sea nítida y con unas condiciones lumínicas aceptables.

Esta historia de usuario se centrará en que el usuario pueda seleccionar de una forma sencilla la parte de la imagen donde se encuentra el texto a traducir.

C.4.4.2. Puntos

- **UX:** 0.5
- **Design:** 0.5
- **Front:** 3
- **Back:** -
- **Documentation:** -

C.4.4.3. Tareas (Pila del sprint)

- **T28** - Buscar biblioteca de código abierto que permita seleccionar parte de una imagen. – 1 punto
- **T29** - Implementar y/o adaptar la biblioteca del proyecto. – 1 punto
- **T30** - Diseñar y crear una interfaz básica que permita seleccionar la parte de una imagen ajustando un rectángulo. – 1 punto
- **T31** - Comprobar que seleccionando parte de una imagen se realiza correctamente el reconocimiento de caracteres. – 1 punto

C.4.4.4. Comentarios

La biblioteca utilizada para esta historia de usuario ha sido desarrollada por Arthur Hub y se llama *Android-Image-Cropper*. Es de código abierto y se encuentra subida en un repositorio Github [29].

C.4.5. US10 - Como usuario quiero que se traduzcan los caracteres reconocidos de una imagen para poder entender el contenido del texto

C.4.5.1. Descripción

Una vez que se han reconocido los caracteres contenidos en una imagen, se utilizarán los servicios de traducción correspondientes para traducir las palabras que forman esos caracteres.

En esta historia de usuario únicamente se realizará la traducción de palabras en inglés a otro idioma, ya que de momento solo se ha probado el reconocimiento OCR con el idioma inglés.

C.4.5.2. Puntos

- **UX:** 1
- **Design:** 1
- **Front:** 4
- **Back:** -
- **Documentation:** -

C.4.5.3. Tareas (Pila del sprint)

- **T32** - Suscribirse en el API de traducción de Microsoft. – 1 punto
- **T33** - Implementar una llamada al API que devuelva la traducción. – 2 puntos
- **T34** - Diseñar y crear una interfaz básica para mostrar el resultado de la traducción. – 2 puntos
- **T35** - Validar la respuesta traducida usando otro servicio de traducción. – 1 punto

C.4.5.4. Comentarios

La suscripción al API de traducción de Microsoft se hizo sin problemas, pero a la hora de realizar una llamada al mismo, se produjeron varios errores de validación del id del cliente, necesario para que el API devuelva el resultado. Tras varios intentos y reconfiguraciones, se consiguió que el API devolviera un texto traducido a partir de un texto en inglés contenido en una imagen.

C.4.6. Revisión de Sprint

C.4.6.1. Incremento

- Puntos de historias de usuario planificados: **16**
- Puntos de historias de usuario realizados: **16**
- Velocidad media: **14.67 puntos/sprint**

En este sprint se ha desarrollado la funcionalidad básica de la aplicación, que permite obtener una imagen con la cámara o galería, seleccionar parte de esa imagen y traducir el texto reconocido en la imagen. Por lo tanto, **se ha implementado parte del CU01, en concreto, lo correspondiente a los requisitos RF01, RF02, RF03 y RF04.**

Con esta funcionalidad básica, se ha generado la **versión 0.1 o Alpha** de la aplicación, que reconoce los caracteres de un texto en inglés contenido en una imagen obtenida desde la cámara o desde la galería.

Para comprobar el correcto funcionamiento de la versión Alpha de la aplicación, se han llevado a cabo los **casos de prueba CP01, CP02, CP03, CP07, CP11 y CP12**.

C.4.6.2. Riesgos

Se ha producido el riesgo **R07**, que consiste en pérdida de documentación y otros artefactos, al haber perdido parte de la memoria tras un fallo del editor de LaTeX.

Se ha aplicado la medida correctora de recuperar la última copia disponible guardada en Dropbox.

Por último, aunque no se ha incluido como riesgo al haber elegido el API de Microsoft voluntariamente, ha habido un problema con la identificación a la hora de realizar las llamadas contra el API. Esto se ha debido un problema de configuración que ha sido solucionado sin problema.

C.5. Sprint 3: 1 de abril – 24 de abril 2016

C.5.1. Pila del producto

ID	Prioridad	Descripción
US36	Muy alta	Como Product Owner quiero tener un modelo de dominio del sistema y una descripción de las clases que lo forman.
US39	Muy alta	Como Product Owner quiero tener un diseño de la arquitectura del sistema, así como una descripción de los patrones utilizados.
US11	Media	Como Usuario quiero seleccionar un idioma de reconocimiento.
US12	Media	Como Usuario quiero seleccionar un idioma al que traducir el texto reconocido.
US13	Baja	Como Usuario deseo disponer del manual de usuario.

Tabla C.4: Pila del producto inicial del Sprint 3

Una vez que se ha estudiado en profundidad el funcionamiento de las bibliotecas de OCR y traducción, se han creado dos nuevas historias de usuario, US36 y US39, y se han priorizado por encima del resto, ya que son necesarias para continuar con el desarrollo de la aplicación.

C.5.2. US36 - Como Product Owner quiero tener un modelo de dominio del sistema y una descripción de las clases que lo forman

C.5.2.1. Descripción

En este punto del proyecto, surge la necesidad de representar el dominio final del problema, con entidades, atributos y relaciones. En el sprint 1 se realizó un modelo de dominio inicial pero ha sido necesario avanzar algo más con el reconocimiento de caracteres y la traducción para realizar los cambios oportunos y obtener la versión final del dominio del problema.

C.5.2.2. Puntos

- UX: -
- Design: -
- Front: -
- Back: -
- Documentation: 4

C.5.2.3. Tareas (Pila del sprint)

- T37 - Realizar el modelo de dominio final. – 3 puntos
- T38 - Describir las clases del modelo de dominio. – 1 punto

C.5.2.4. Comentarios

Después de varios cambios en el modelo de dominio, la versión final consta de una clase *Traducción* y otra *Idioma*, generalización de otras dos, idioma de partida e idioma de llegada.

C.5.3. US39 - Como Product Owner quiero tener un diseño de la arquitectura del sistema, así como una descripción de los patrones de diseño utilizados

C.5.3.1. Descripción

Después de realizar el análisis del sistema, es muy importante definir la estructura, funcionamiento e interacción entre las partes del software. Esto es lo que se conoce como arquitectura del sistema.

Por otro lado, es importante hacer uso de soluciones que han resuelto problemas similares en ocasiones anteriores. Estas soluciones reciben el nombre de patrones de diseño.

C.5.3.2. Puntos

- **UX:** -
- **Design:** -
- **Front:** -
- **Back:** -
- **Documentation:** 12

C.5.3.3. Tareas (Pila del sprint)

- **T40** - Definir patrón arquitectónico. – 2 puntos
- **T41** - Definir patrones de diseño a utilizar - 2 puntos
- **T42** - Modelo general de la arquitectura. – 1 punto
- **T43** - Descomposición modular. – 1 punto
- **T44** - Diseño de los módulos o subsistemas. - 3 puntos
- **T45** - Relación entre módulos. - 3 puntos

C.5.3.4. Comentarios

Se ha producido una desviación en la planificación de puntos para esta historia de usuario. Se habían planificado 12, pero ha sido insuficiente y ha sido necesario dedicar más tiempo y por lo tanto, más puntos.

Esta desviación se ha producido, entre otras cosas, porque al desarrollar para Android, ha habido clases que no encajaban del todo en alguno de los módulos definidos, como pueden ser los adaptadores de listas, que se han incluido en el módulo Vista pero dependen del modelo. Esta dependencia va en contra del patrón MVP, que indica que no puede haber dependencia directa entre Vista y Modelo.

C.5.4. Revisión de Sprint

C.5.4.1. Incremento

- Puntos de historias de usuario planificados: **16**
- Puntos de historias de usuario realizados: **18**
- Velocidad media: **15.5 puntos/sprint**

En este sprint no ha habido incremento de funcionalidad, ya que era necesario definir el modelo de dominio, la arquitectura y los patrones a utilizar antes de continuar.

C.5.4.2. Riesgos

Se ha producido el riesgo **R08**, ya que ha habido una desviación de los puntos estimados para la historia de usuario US39. Los puntos desviados han sido 2, lo que ha supuesto un día más de trabajo.

C.6. Sprint 4: 25 de abril – 15 de mayo 2016

C.6.1. Pila del producto

ID	Prioridad	Descripción
US46	Muy alta	Como Product Owner quiero que el código de la aplicación siga el patrón arquitectónico seleccionado, en este caso MVP (Modelo-Vista-Presentador), para facilitar la modularidad y legibilidad del código.
US11	Muy alta	Como Usuario quiero seleccionar un idioma de reconocimiento.
US12	Muy alta	Como Usuario quiero seleccionar un idioma al que traducir el texto reconocido.
US13	Media	Como Usuario deseo disponer del manual de usuario.

Tabla C.5: Pila del producto inicial del Sprint 4

Se ha creado una nueva historia de usuario, US46, para aplicar los patrones definidos en el sprint anterior.

C.6.2. US46 - Como Product Owner quiero que el código de la aplicación siga el patrón arquitectónico seleccionado, en este caso MVP (Modelo-Vista-Presentador)

C.6.2.1. Descripción

Una vez seleccionado el patrón arquitectónico a utilizar, será aplicado en el código para facilitar la modularidad y legibilidad del mismo.

C.6.2.2. Puntos

- **UX:** -
- **Design:** -
- **Front:** 8
- **Back:** -
- **Documentation:** -

C.6.2.3. Tareas (Pila del sprint)

- **T47** - Crear diferentes paquetes dentro del proyecto: views, models, presenters y helpers. – 0.5 puntos
- **T48** - Trasladar la actividad principal (MainActivity.java) al paquete views, crear su presentador en el paquete presenters y establecer referencias para conectarlos entre sí. – 0.5 puntos
- **T49** - Implementar una llamada desde la vista principal a un método de su presentador, de modo que cuando se pulse el botón de obtener imagen, mande a la vista abrir un menú para que el usuario realice una foto con la cámara o la seleccione de la galería. – 1 punto
- **T50** - Implementar una llamada desde la vista principal a un método de su presentador, que obtenga la URI de la imagen seleccionada y ordene a la vista cambiarse a otra una nueva vista (OCRActivity.java) para realizar el reconocimiento de caracteres. – 1 punto

- **T51** - Una vez creada la nueva vista de reconocimiento, crear su presentador en el paquete presenters y establecer referencias para conectarlos entre sí. – 0.5 puntos
- **T52** - Implementar una llamada desde la vista de reconocimiento a su presentador una vez que ésta esté cargada, para que muestre la imagen seleccionada por el usuario con un rectángulo que le permita seleccionar una parte de la misma. – 0.5 puntos
- **T53** - Implementar una llamada desde la vista a un método de su presentador, de modo que cuando se pulse el botón del tick, obtenga y guarde la parte de la imagen seleccionada y le ordene a la vista iniciar una nueva vista (TranslationActivity.java) en la que se realizará la traducción. – 1.5 puntos
- **T54** - Una vez creada la vista de traducción, crear su presentador en el paquete presenters y establecer referencias para conectarlos entre sí. – 0.5 puntos
- **T55** - Implementar una llamada desde la vista de traducción a su presentador una vez que ésta esté cargada, de modo que se muestre el texto reconocido y su idioma – 0.5 puntos
- **T56** - Implementar una llamada desde la vista de traducción a un método de su presentador, de modo que cuando se pulse el botón de traducción, realice la traducción y devuelva el resultado a la vista para mostrárselo al usuario. – 1.5 puntos

C.6.2.4. Comentarios

Han existido algunos problemas a la hora de pasar imágenes en alta resolución desde la vista de seleccionar parte de una imagen a la vista resultado. Esto se debe a que la imagen debe estar lo más nítida posible para realizar el reconocimiento OCR pero los buffers utilizados para pasar imágenes tienen sus límites de tamaño. La solución que se ha adoptado es guardar la imagen temporalmente desde una actividad y recuperarla desde la siguiente.

C.6.3. US11 - Como Usuario quiero seleccionar un idioma de reconocimiento

C.6.3.1. Descripción

Uno de los objetivos de esta aplicación es que pueda ser utilizada en diferentes países y por personas de diferentes nacionalidades, por lo que es fundamental no limitar el reconocimiento de caracteres y la traducción a un único idioma.

En esta historia de usuario se habilitará la opción para que el usuario seleccione un idioma de reconocimiento de una lista.

Para reconocer cualquier idioma es necesario descargarse un fichero ".traineddata" e incluirlo en el directorio de la aplicación en el dispositivo. Por lo tanto, y para evitar problemas de almacenamiento, habrá que mostrar una opción para descargar y/o borrar idiomas de reconocimiento.

C.6.3.2. Puntos

- **UX:** - 0.5
- **Design:** 0.5
- **Front:** 4
- **Back:** -
- **Documentation:** -

C.6.3.3. Tareas (Pila del sprint)

- **T57** - Diseñar y crear una interfaz básica con una lista desplegable que permita seleccionar, descargar y/o borrar un idioma de reconocimiento. – 1 punto
- **T58** - Implementar una llamada al repositorio Github de Tesseract [30] para descargar los idiomas de reconocimiento seleccionados por el usuario. – 2 puntos
- **T59** - Implementar el borrado de idiomas de reconocimiento seleccionados por el usuario. – 1 punto
- **T60** - Obtener el idioma de reconocimiento seleccionado y pasárselo al motor OCR para que realice el reconocimiento usando ese idioma. – 1 punto

C.6.3.4. Comentarios

Para la descarga/borrado de los idiomas de reconocimiento se han utilizado unos botones de progreso, que muestran el porcentaje de descarga de un idioma, los idiomas descargados y a su vez sirven para descargar y/o borrar un idioma si se pulsa sobre ellos.

Para la implementación de estos botones de progreso, se ha hecho uso de una biblioteca de un desarrollador canadiense llamado Prateek Srivastava. [31]

C.6.4. US12 - Como Usuario quiero seleccionar un idioma al que traducir el texto reconocido

C.6.4.1. Descripción

Para que la aplicación pueda ser utilizada por usuarios de diferentes nacionalidades, es fundamental que el texto reconocido se pueda traducir a una gran variedad de idiomas.

En esta historia de usuario se habilitará la opción para que el usuario seleccione un idioma al que traducir el texto reconocido previamente.

El idioma seleccionado será utilizado para realizar la llamada al API de traducción correspondiente.

C.6.4.2. Puntos

- **UX:** - 0.5
- **Design:** 0.5
- **Front:** 1
- **Back:** -
- **Documentation:** -

C.6.4.3. Tareas (Pila del sprint)

- **T61** - Diseñar y crear una interfaz básica con una lista desplegable que permita seleccionar un idioma de llegada en la traducción. – 1 punto
- **T62** - Obtener el idioma de traducción seleccionado y usarlo en la llamada al API de traducción para obtener el texto traducido al idioma correspondiente. – 1 punto

C.6.4.4. Comentarios

No existe ningún comentario adicional.

C.6.5. Revisión de Sprint

C.6.5.1. Incremento

- Puntos de historias de usuario planificados: **15**
- Puntos de historias de usuario realizados: **15**
- Velocidad media: **15.4 puntos/sprint**

En este sprint se ha modularizado el código desarrollado hasta este punto, aplicando el patrón MVP.

Por otro lado, se ha completado la funcionalidad básica de la aplicación con la selección del idioma de reconocimiento y el idioma de traducción. Por lo tanto, **se ha continuado con la implementación del CU01, en concreto, lo correspondiente a los requisitos RF05 y RF06.**

Con esta funcionalidad, se ha generado la **versión 0.2 o Beta** de la aplicación, que además de reconocer los caracteres y realizar la traducción, permite seleccionar, descargar y/o borrar idiomas de reconocimiento y seleccionar un idioma de traducción entre 39 diferentes.

Para comprobar el correcto funcionamiento de la versión Beta de la aplicación, se han llevado a cabo los **casos de prueba CP04, CP05, CP06, CP08, CP09 y CP10.**

C.6.5.2. Riesgos

En este sprint se ha producido el riesgo **R05**, que consiste en la inestabilidad en el IDE Android Studio o posibilidad de cambio de versión.

En este caso, cuando se ha comenzado a separar el código en diferentes capas, Gradle tardaba más de lo normal en construir el proyecto e incluso ha dado algún error al intentar ejecutar el comando ndk-build. Este comando se usa para construir el proyecto OCR Tesseract pero, al haber incluido en el proyecto las bibliotecas ya compiladas, no es necesario que se ejecute este comando más veces. La solución a este problema ha sido deshabilitar la ejecución de ese comando cuando Gradle construya el proyecto.

Por otro lado, Google lanzó la nueva versión r11 de Android NDK en Marzo y producía errores a la hora de construir tesseract, por lo que hubo que aplicar la medida de corrección de volver a la versión anterior r10e, la cual funciona correctamente.

C.7. Sprint 5: 16 de mayo – 5 de junio 2016

C.7.1. Pila del producto

ID	Prioridad	Descripción
US63	Muy alta	Como Product Owner quiero tener un modelo de datos del sistema con su correspondiente esquema relacional.
US66	Muy alta	Como Usuario quiero guardar una traducción realizada.
US70	Muy alta	Como Usuario quiero visualizar las traducciones guardadas.
US13	Media	Como Usuario deseo disponer del manual de usuario.

Tabla C.6: Pila del producto inicial del Sprint 5

Se han creado tres nuevas historias de usuario. US63 se ha repriorizado por encima de US66 ya que es necesaria para implementar la base de datos donde se almacenarán las traducciones de US66.

C.7.2. US63 - Como Product Owner quiero tener un modelo de datos del sistema con su correspondiente esquema relacional

C.7.2.1. Descripción

Es muy importante obtener un esquema relacional completo de la base de datos antes de comenzar su implementación. De esta forma, se evitan relaciones innecesarias entre tablas y se previenen problemas de integridad y/o consistencia de los datos.

C.7.2.2. Puntos

- **UX:** -
- **Design:** -
- **Front:**
- **Back:** -
- **Documentation:** 3

C.7.2.3. Tareas (Pila del sprint)

- **T64** - Realizar el esquema relacional de la base de datos – 2 puntos
- **T65** - Describir las tablas utilizadas en el esquema relacional. – 1 puntos

C.7.2.4. Comentarios

No existe ningún comentario adicional.

C.7.3. US66 - Como Usuario quiero guardar una traducción realizada

C.7.3.1. Descripción

Cuando se planteó este proyecto, se pensó en buscar un punto diferenciador de otros traductores convencionales. El punto diferenciador en este caso es el almacenamiento de traducciones, ya que otros traductores no tienen persistencia, simplemente permiten realizar la traducción.

Además, para cada traducción se guardará su fecha de creación y localización (opcional), lo cual permitirá al usuario realizar un filtrado en función de los lugares visitados y en qué fechas. El filtrado será implementando siempre y cuando los plazos lo permitan.

C.7.3.2. Puntos

- **UX:** -
- **Design:** 1
- **Front:** 2
- **Back:** 2
- **Documentation:** -

C.7.3.3. Tareas (Pila del sprint)

- **T67** - Añadir un botón de guardado y otro de localización en la vista de la traducción. – 1 punto
- **T68** - Implementar la obtención de la localización del dispositivo. – 2 puntos
- **T69** - Implementar una llamada a la base de datos para guardar una traducción. – 2 puntos

C.7.3.4. Comentarios

Para obtener la localización del dispositivo se ha utilizado Google Play Services. [32]

C.7.4. US70 - Como Usuario quiero visualizar las traducciones guardadas

C.7.4.1. Descripción

Otra de las funcionalidades interesantes de la aplicación es poder consultar las traducciones guardadas.

En esta historia de usuario se implementará una lista de tarjetas con las traducciones almacenadas en la base de datos.

C.7.4.2. Puntos

- **UX:** 1
- **Design:** 1
- **Front:** 3
- **Back:** - 1
- **Documentation:** -

C.7.4.3. Tareas (Pila del sprint)

- **T71** - Diseñar una plantilla de tarjeta que represente cada una de las traducciones a mostrar. – 2 puntos
- **T72** - Implementar una llamada a la base de datos para recuperar las traducciones guardadas por el usuario. – 1 punto
- **T73** - Implementar una vista reciclable y un adaptador que gestione la lista de traducciones representada en esta vista. – 3 puntos

C.7.4.4. Comentarios

Ha sido necesario invertir la lista de traducciones obtenida de la base de datos para que en la lista se muestren primero las traducciones más recientes.

C.7.5. Revisión de Sprint

C.7.5.1. Incremento

- Puntos de historias de usuario planificados: **14**
- Puntos de historias de usuario realizados: **14**
- Velocidad media: **15.17 puntos/sprint**

En este sprint se ha realizado el **modelo de datos**, que representa la persistencia de la aplicación.

En este sprint se ha finalizado la implementación del caso de uso **CU01. Añadir Traducción**, con la historia de usuario US66, que se corresponde a los requisitos **RF07 y RF13**.

También se ha implementado el caso de uso **CU02. Visualizar Traducciones**, correspondiente al requisito **RF08**.

Para comprobar el correcto funcionamiento de la funcionalidad añadida en este sprint se han llevado a cabo los **casos de prueba CP13, CP14 y CP15**.

C.7.5.2. Retroalimentación del Product Owner

En la reunión con el Product Owner al final de este sprint, éste sugirió que el modelo de datos podría contener también a los idiomas, guardándoles en la base de datos e indicando con un atributo si están descargados o no.

En la US11 y US12 realizadas en el sprint anterior, los idiomas disponibles se almacenaban en dos ficheros de strings en los recursos del proyecto y se mostraban como descargados o no comprobando si existían los ficheros de cada idioma en el dispositivo.

Se ha decidido realizar estos cambios, tanto en el modelo de datos como en el código, en el siguiente sprint.

C.7.5.3. Riesgos

En este sprint no se ha disparado ningún riesgo.

C.8. Sprint 6: 7 de junio – 30 de junio 2016

Este sprint es cuatro días más largo que el resto de sprints. Esto se debe a que se ha decidido añadir más funcionalidades a la aplicación y con un sprint de tres semanas no daría tiempo a implementar todos los puntos planificados.

C.8.1. Pila del producto

ID	Prioridad	Descripción
US74	Muy alta	Como Product Owner quiero que los idiomas aparezcan como una tabla más en el modelo de datos y que se actualice la base de datos SQLite con estos cambios.
US78	Muy alta	Como Product Owner quiero tener la documentación actualizada en caso de añadir nueva funcionalidad a la aplicación.
US82	Muy alta	Como Usuario quiero consultar el detalle de una traducción.
US85	Muy alta	Como Usuario quiero borrar una traducción.
US89	Alta	Como Usuario quiero buscar una traducción.
US94	Alta	Como Usuario quiero filtrar las traducciones.
US13	Alta	Como Usuario deseo disponer del manual de usuario.

Tabla C.7: Pila del producto inicial del Sprint 6

En este sprint se ha decidido añadir más funcionalidad a la aplicación. Esta funcionalidad aporta un valor adicional a la aplicación y a la experiencia de usuario.

Ha sido necesario actualizar parte de la documentación para añadir los nuevos casos de uso derivados de esta funcionalidad añadida. Estos cambios se realizarán en la historia de usuario US74.

C.8.2. US74 - Como Product Owner quiero que los idiomas aparezcan como una tabla más en el modelo de datos y que se actualice la base de datos SQLite con estos cambios

C.8.2.1. Descripción

En esta historia de usuario se realizan los cambios propuestos por el Product Owner al final del sprint anterior.

C.8.2.2. Puntos

- **UX:** -
- **Design:** -
- **Front:**
- **Back:** 3
- **Documentation:** 1

C.8.2.3. Tareas (Pila del sprint)

- **T75** - Añadir una nueva tabla al esquema relacional de la base de datos. – 1 punto

- **T76** - Añadir al script de creación de la base de datos la tabla Language. – 1 punto
- **T77** - Añadir nuevos métodos en la base de datos para recuperar los idiomas de reconocimiento y/o traducción y para comprobar si un idioma está descargado o no. – 2 puntos

C.8.2.4. Comentarios

No existe ningún comentario adicional.

C.8.3. US78 - Como Product Owner quiero tener la documentación actualizada en caso de añadir nueva funcionalidad a la aplicación

C.8.3.1. Descripción

En esta historia de usuario se actualizan todos aquellos documentos que se vean afectados por la nueva funcionalidad añadida.

C.8.3.2. Puntos

- **UX:** -
- **Design:** -
- **Front:**
- **Back:** -
- **Documentation:** 6

C.8.3.3. Tareas (Pila del sprint)

- **T79** - Actualizar la lista de requisitos, tanto funcionales como no funcionales. – 1 punto
- **T80** - Análisis: añadir los nuevos casos de uso, su especificación y diagramas de actividad. – 2 puntos
- **T81** - Diseño/Arquitectura: añadir las nuevas clases utilizadas a los diferentes módulos y las relaciones entre ellas u otras ya existentes. Añadir nuevos diagramas de secuencia. – 3 puntos

C.8.3.4. Comentarios

No existe ningún comentario adicional.

C.8.4. US82 - Como Usuario quiero consultar el detalle de una traducción

C.8.4.1. Descripción

Esta funcionalidad pretende mostrar toda aquella información que no se pueda representar en la lista principal de traducciones.

C.8.4.2. Puntos

- **UX:** 0.5
- **Design:** 0.5
- **Front:** 1
- **Back:** -
- **Documentation:** -

C.8.4.3. Tareas (Pila del sprint)

- **T83** - Diseñar y crear una interfaz que muestre el detalle de una traducción, incluyendo una miniatura de la imagen, fecha, localización, idioma de partida, texto reconocido, idioma de llegada y texto traducido. – 1 punto
- **T84** - Obtener los detalles de la traducción y mostrarlos en la vista de detalle de traducción. – 1 punto

C.8.4.4. Comentarios

Para la tarea T84, inicialmente se llamaba a la base de datos para traer los datos de la traducción pero posteriormente se cambió y los datos se obtienen desde la propia lista de traducciones. De esta forma, se evita una consulta a la base de datos.

C.8.5. US85 - Como Usuario quiero borrar una traducción**C.8.5.1. Descripción**

Una vez añadida la funcionalidad de guardar una traducción, lo lógico es permitir al usuario que pueda borrarla si ya no le interesa.

C.8.5.2. Puntos

- **UX:** 1
- **Design:** -
- **Front:** 0.5
- **Back:** 2
- **Documentation:** -

C.8.5.3. Tareas (Pila del sprint)

- **T86** - Implementar el deslizamiento de una tarjeta (traducción). – 0.5 puntos
- **T87** - Implementar una opción para deshacer el borrado de la traducción. - 1 punto
- **T88** - Implementar una llamada a la base de datos para borrar una traducción. – 2 puntos

C.8.5.4. Comentarios

No existe ningún comentario adicional.

C.8.6. US89 - Como Usuario quiero buscar una traducción

C.8.6.1. Descripción

Para completar la experiencia de usuario con el historial de traducciones, se ha decidido implementar una búsqueda de traducciones por texto.

C.8.6.2. Puntos

- **UX:** 1
- **Design:** 0.5
- **Front:** 2.5
- **Back:** -
- **Documentation:** -

C.8.6.3. Tareas (Pila del sprint)

- **T90** - Diseñar una barra de búsqueda. – 1 punto
- **T91** - Implementar un filtrado de traducciones en función del texto introducido. – 2 puntos
- **T92** - Actualizar la lista de traducciones y mostrar una animación en las tarjetas. – 1 punto

C.8.6.4. Comentarios

Para realizar esta historia de usuario se ha aprovechado parte del código de la respuesta de esta consulta en el foro Stack Overflow [33].

C.8.7. US93 - Como Usuario quiero filtrar las traducciones

C.8.7.1. Descripción

Por último, se ha decidido añadir la funcionalidad de filtrar traducciones.

C.8.7.2. Puntos

- **UX:** 0.5
- **Design:** 0.5
- **Front:** 1
- **Back:** -

- **Documentation:** -

C.8.7.3. Tareas (Pila del sprint)

- **T94** - Diseñar un diálogo que contenga filtros por fecha y/o localización. – 1 punto
- **T95** - Implementar un filtrado de traducciones en función de los filtros seleccionados. – 0.5 puntos
- **T96** - Actualizar la lista de traducciones y mostrar una animación en las tarjetas. – 0.5 puntos

C.8.7.4. Comentarios

En las tareas T95 y T96 se aprovecha gran parte del código realizado en las tareas T91 y T92.

C.8.8. US13 - Como Usuario quiero disponer del manual de usuario

C.8.8.1. Descripción

Esta historia de usuario ha sido incluida en la pila del producto desde el inicio del proyecto pero no ha sido posible realizarla hasta que se ha implementado toda la funcionalidad.

C.8.8.2. Puntos

- **UX:** -
- **Design:** -
- **Front:** -
- **Back:** -
- **Documentation:** 4

C.8.8.3. Tareas (Pila del sprint)

- **T97** - Realizar capturas de la aplicación para cada caso de uso. – 1 punto
- **T98** - Detallar como utilizar la aplicación en cada caso de uso. – 3 puntos

C.8.8.4. Comentarios

Las capturas de la aplicación se han realizado en un dispositivo Nexus 5X.

C.8.9. Revisión de Sprint

C.8.9.1. Incremento

- Puntos de historias de usuario planificados: **24.5**
- Puntos de historias de usuario realizados: **24.5**

- Velocidad media: **16.5 puntos/sprint**

En este sprint se ha **actualizado el modelo de datos** y parte del código de la base de datos.

Se ha añadido a la documentación los cambios correspondientes a las nuevas funcionalidades añadidas.

Se ha llevado a cabo la implementación de los casos de uso **CU03, CU04, CU05 y CU06** que se corresponden con los requisitos **RF09, RF10, RF11 y RF12**.

Además, se ha generado la **versión 1.0 o Release** de la aplicación, que contiene toda la funcionalidad planificada.

Para comprobar el correcto funcionamiento de la funcionalidad añadida en este sprint, se han llevado a cabo los casos de prueba **CP16, CP17, CP18, CP19, CP20, CP21, CP22 y CP23**.

C.8.9.2. Riesgos

En este sprint no se ha disparado ningún riesgo.

Apéndice D

Contenido del CD

El contenido del CD adjunto es el siguiente:

- **aplicación:** contiene el ejecutable de la aplicación.
 - **AsistevApp.apk**
- **código:** contiene el software desarrollado en versión fuente.
- **diagramas:** contiene el fichero con los diagramas UML realizados.
 - **diagramas.asta**
- **memoria.pdf:** documento de la memoria del TFG.