

tDec: Una aplicació generadora de sub-aplicacions de decisió

Nombre Estudiante: Ariadna Llovet Soto

Máster Universitario en Desarrollo de Aplicaciones para Dispositivos Móviles

Nombre Consultor/a: Eduard Martin Lineros

Profesor/a responsable de la asignatura: Carles Garrigues Olivella

3 de enero del 2018

© 2018 - Ariadna Llovet Soto

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

FICHA DEL TRABAJO FINAL

Título del trabajo:	tDec: Una aplicación generadora de sub-aplicaciones de decisión
Nombre del autor:	Ariadna Llovet Soto
Nombre del consultor/a:	Eduard Martin Lineros
Nombre del PRA:	Carles Garrigues Olivella
Fecha de entrega (mm/aaaa):	01/2018
Titulación:	Máster Universitario en Desarrollo de Aplicaciones para Dispositivos Móviles
Idioma del trabajo:	Castellano
Palabras clave	autogenerar aplicación móvil
Resumen del Trabajo:	
<p>Muchas profesiones o situaciones de la vida cotidiana se encuentran con tareas que se pueden resolver fácilmente contestando a una serie de preguntas, o tareas que deben o podrían homogeneizarse en un entorno empresarial o personal.</p> <p>En la actualidad, para resolver este tipo de tareas se acude a manuales o programas informáticos creados especialmente para resolver la tarea en concreto.</p> <p>Con este trabajo se pretende solucionar estos problemas, desarrollando una aplicación (tDec) sencilla capaz de crear sub-aplicaciones de toma de decisiones. Esta sub-aplicación será ejecutable dentro de la misma aplicación, y de este modo no será necesario contratar a ningún desarrollador para crear una aplicación a medida.</p> <p>La aplicación será híbrida con el fin de abarcar el mayor número de usuarios posible. Los cuales pueden encontrarse en cualquier parte del mundo por lo que se requiere que la aplicación sea portátil, fácilmente accesible, y funcional sin conexión a internet.</p> <p>Debido a los requisitos marcados de la aplicación, se realizará el desarrollo siguiendo el método “waterfall”, y utilizando el patrón de arquitectura de Modelo Vista Controlador (MVC).</p>	
Abstract:	
Different professions and everyday life situations have tasks that could be easily	

resolved answering questions, or tasks that could be homogenized for a personal use or even in a work environment.

Nowadays, in order to solve these kind of tasks, people use manuals or programs specifically created to solve a particular task.

The target of this work is to provide a solution to these problems by developing an easy to use mobile application (tDec), able to create decision making sub-applications. A sub-application will be executed inside of the very same application (tDec). This way, there will be no need to pay for a customized app development.

In order to be able to provide the service to the maximum amount of users, the application will be hybrid. tDec users could be found anywhere in the world, therefore the application needs to be portable, accessible, and not-network dependant.

Given the application requirements and the limited time to develop it, the methodology used will be “waterfall”, using the architecture pattern Model View Controller (MVC).

Índice

1. INTRODUCCIÓN	1
1.1 CONTEXTO Y JUSTIFICACIÓN DEL TRABAJO	1
<i>Situación del mercado</i>	2
<i>Stakeholders</i>	2
1.2 OBJETIVOS DEL TRABAJO	4
<i>El producto</i>	4
<i>Objetivos</i>	4
<i>Requisitos funcionales</i>	5
Crear sub-aplicación	5
Editar sub-aplicación	8
Eliminar sub-aplicación	9
Mostrar lista de sub-aplicaciones	9
Ejecutar sub-aplicación	10
Descargar sub-aplicación	10
Cargar sub-aplicación	10
<i>Requisitos no funcionales</i>	11
1.3 ENFOQUE Y MÉTODO SEGUIDO	12
<i>Tamaño de pantalla</i>	12
<i>Sistema operativo</i>	13
<i>Situación actual</i>	13
1.4 PLANIFICACIÓN DEL TRABAJO	14
<i>Estrategia de desarrollo</i>	14
<i>Diseño</i>	15
<i>Implementación</i>	18
<i>Implementación final</i>	22
1.5 BREVE SUMARIO DE PRODUCTOS OBTENIDOS	24
1.6 BREVE DESCRIPCIÓN DE LOS OTROS CAPÍTULO DE LA MEMORIA	24
2. USUARIOS Y CONTEXTO DE USO	25
2.1 FICHAS DE USUARIO	26
<i>Ficha de usuario I</i>	26
<i>Ficha de usuario II</i>	27
<i>Ficha de usuario III</i>	28
3. DISEÑO CONCEPTUAL	29
3.1 ESCENARIOS DE USO	29
<i>Crear sub-aplicación</i>	29
<i>Editar sub-aplicación</i>	29
<i>Editar información de la sub-aplicación</i>	29
<i>Ejecutar sub-aplicación</i>	30
<i>Eliminar sub-aplicación</i>	30
<i>Descargar sub-aplicación</i>	30
<i>Cargar sub-aplicación</i>	31
<i>Crear pantalla</i>	31
<i>Editar pantalla</i>	31
<i>Eliminar pantalla</i>	31
<i>Crear enlace</i>	31
<i>Editar enlace</i>	32
<i>Eliminar enlace</i>	32
3.2 FLUJOS DE INTERACCIÓN	33

3.2.1 Crear sub-aplicación	33
3.2.2 Editar sub-aplicación	33
3.2.3 Editar información de la sub-aplicación.....	34
3.2.4 Ejecutar sub-aplicación	34
3.2.5 Eliminar sub-aplicación	34
3.2.6 Descargar sub-aplicación	35
3.2.7 Cargar sub-aplicación.....	35
3.2.8 Crear pantalla.....	35
3.2.9 Editar pantalla.....	36
3.2.10 Eliminar pantalla	36
3.2.11 Crear enlace.....	36
3.2.12 Editar enlace.....	37
3.2.13 Eliminar enlace	37
3.2 DIAGRAMA DE FLUJO.....	38
4. PROTOTIPO.....	39
4.2 SITEMAP	39
4.3 PROTOTIPO	39
<i>Pantalla principal.....</i>	<i>40</i>
<i>Ejecución de sub-aplicación.....</i>	<i>40</i>
<i>Edición de sub-aplicación</i>	<i>43</i>
5. CASOS DE USO	49
CREAR SUB-APLICACIÓN	50
EDITAR SUB-APLICACIÓN.....	50
EDITAR INFORMACIÓN DE LA SUB-APLICACIÓN.....	50
EJECUTAR SUB-APLICACIÓN	50
ELIMINAR SUB-APLICACIÓN	51
DESCARGAR SUB-APLICACIÓN	51
CARGAR SUB-APLICACIÓN	51
CREAR PANTALLA.....	51
EDITAR PANTALLA.....	52
ELIMINAR PANTALLA	52
CREAR ENLACE	52
EDITAR ENLACE	52
ELIMINAR ENLACE.....	53
5. DISEÑO DE LA ARQUITECTURA	53
DATOS.....	53
ENTIDADES Y CLASES	56
ARQUITECTURA DEL SISTEMA.....	57
5. DESARROLLO	59
TECNOLOGÍAS DE DESARROLLO	59
ESTRUCTURA DEL PROYECTO	59
BASE DE DATOS.....	59
PROVEEDOR DE DATOS	60
IMPORTACIÓN Y EXPORTACIÓN DE SUB-APLICACIONES	60
IMPLEMENTACIONES ADICIONALES	61
COMPLICACIONES.....	62
6. PRUEBAS	63
7. CONCLUSIONES.....	75
FUTURAS MEJORAS.....	76

8. GLOSARIO.....	77
9. BIBLIOGRAFÍA	79
10. ANEXOS	81

Lista de figuras

Figura 1: Diagrama para la toma de decisiones.....	1
Figura 2: Situación del mercado en España.....	2
Figura 3: Stakeholders de la aplicación.....	4
Figura 4: Categorización de tipos de adaptaciones a los problemas de fragmentación.	13
Figura 5: Los números de Zingtree.....	14
Figura 6: Modelo de desarrollo waterfall.	15
Figura 7: Gantt. Fase de diseño.	17
Figura 8: Gantt. Fase de implementación.....	20
Figura 9: Gantt. Fase de implementación Parte 2.....	21
Figura 10: Gantt. Fase final.....	23
Figura 11: Usuario 1.....	26
Figura 12: Usuario 2.....	27
Figura 13: Usuario 3.....	28
Figura 14: Diagrama de flujo de la aplicación.....	38
Figura 15: Sitemap.....	39
Figura 16: Diagrama de casos de uso.....	49
Figura 17: UML BD.....	56
Figura 18: Diagrama de clases.....	57
Figura 19: Arquitectura MVC.....	58
Figura 20: Ionic MVC.....	58
Figura 21: Estructura del proyecto.....	59
Figura 22: Almacenamiento de datos.....	60

1. Introducción

1.1 Contexto y justificación del Trabajo

Actualmente, muchas profesiones o situaciones de la vida cotidiana se encuentran con tareas que se pueden resolver fácilmente contestando a una serie de preguntas.

Un ejemplo podría encontrarse en la medicina. Un médico, para saber el diagnóstico de un paciente con dolor torácico, realiza una serie de preguntas, y a partir de sus respuestas obtiene el diagnóstico para ese dolor.

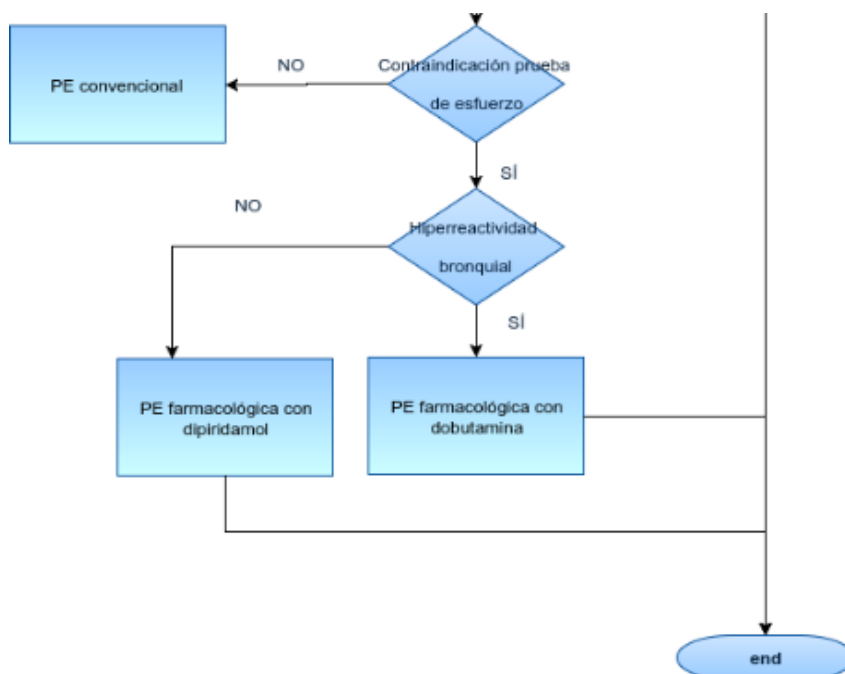


Figura 1: Diagrama para la toma de decisiones

En la actualidad, para resolver este tipo de tareas se acude a manuales o programas informáticos creados especialmente para resolver la tarea en concreto.

Al no poder permitirse todo el mundo el pagar el desarrollo de una aplicación informática para resolver un problema, en la mayoría de los casos se acude a manuales físicos. Esto es una tarea lenta, y en muchos casos puede provocar que la persona haya memorizado los pasos y que, por lo tanto, por descuido, se olvide de realizar algún paso.

Otro problema con el cual se puede encontrar es la falta de homogeneización de la tarea en un mismo puesto de trabajo, es decir, cada trabajador realiza la tarea siguiendo su propia metodología, que a veces puede o no ser la más adecuada.

Volviendo al caso de la medicina, los hospitales se encuentran con un elevado número de visitas diarias y aunque hay una plantilla fija de médicos y enfermeros, ésta es pequeña, con lo que el hospital ha de contratar personal a días sueltos para cubrir los turnos de trabajo. Este personal contratado tiene diferentes grados de experiencia, en general, son especialistas jóvenes que han terminado la especialidad en el propio hospital hace un año o dos, o especialistas formados en otros centros sanitarios con diferentes grados de implicación en el servicio, y por lo tanto con distintas metodologías y/o conocimientos.

Para tratar de homogeneizar la atención al paciente con dolor torácico se piensa que sería útil disponer de una aplicación actualizada de consulta rápida a través del Smartphone en el lugar de trabajo, pero el hospital actualmente no dispone de suficientes recursos como para pagar el desarrollo de la aplicación.

Con este trabajo se pretende solucionar estos problemas, desarrollando una aplicación sencilla capaz de crear sub-aplicaciones de toma de decisiones. Esta sub-aplicación será ejecutable dentro de la misma aplicación, y de este modo no será necesario contratar a ningún desarrollador para crear la aplicación que se necesitaba. La aplicación será nombrada tDec.

tDec será usada en un contexto donde el usuario puede o no disponer de conexión a internet de modo que es necesario que las sub-aplicaciones sean ejecutables sin la necesidad de disponer conexión a internet. Los usuarios pueden encontrarse en cualquier parte del mundo por lo que se requiere que la aplicación sea portátil y fácilmente accesible. Finalmente, la sub-aplicación debe ser accesible entre usuarios por lo que debe ser compartible.

Situación del mercado

A continuación, se muestra la situación del mercado actual para posibles usuarios de la aplicación:

En España existen aproximadamente 2.500.000 compañías registradas. Muchas de estas compañías podrían aprovechar los beneficios que tDec ofrece a la hora de crear ellos mismos sus aplicaciones de soporte.

Si se utilizase la aplicación únicamente para fines médicos, en España existen 787 hospitales, de donde se derivan más de 50 especialidades y más de 160.000 médicos.

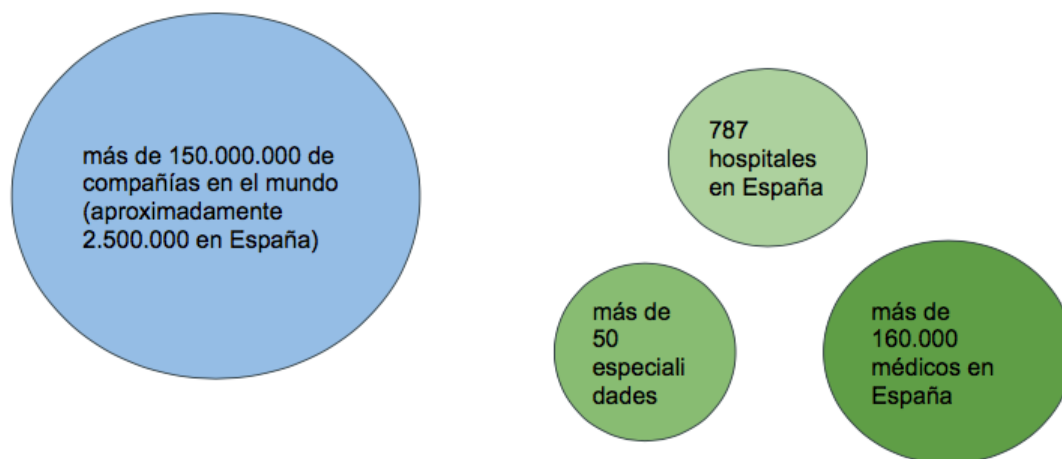


Figura 2: Situación del mercado en España

Stakeholders

Los stakeholders son personas u organizaciones involucradas de manera formal e informal con la solución de negocio ofrecida en la experiencia del cliente.

A continuación se indican los stakeholders de tDec:

Encargado

Persona o empleado que se encuentra en el puesto superior de una jerarquía de empresa y que tiene las facultades necesarias para mandar a sus subordinados.

En tDec será el usuario creador de la sub-aplicación y encargado de compartirlo con sus compañeros de trabajo.

Supervisado

Personal o empleado que se encuentra en un puesto inferior en la jerarquía de empresa y que es mandado por el encargado.

En tDec será el usuario consumidor de la sub-aplicación que se le habrá proporcionado para agilizar y homogenizar su trabajo.

Cliente de la empresa

Persona (cliente) que se beneficia indirectamente del uso de la aplicación.

Departamento

Departamento de la empresa que se beneficia del uso de la aplicación por la eficacia y por lo tanto menor coste que la aplicación le supone.

Empresa

Empresa en la que se utiliza la sub-aplicación.

Asociación

Asociación o empresa que colabora con la empresa que ha instalado la aplicación.

No siempre todos los stakeholders existirán en el uso de tDec, ya que puede darse el caso que un particular la utilice para fines propios. En este caso el único stakeholder sería el usuario, el cual tomaría los roles de encargado y supervisado a la vez.

A continuación se muestra el mapa genérico de stakeholders.

Empleados:	Los stakeholders tiene participación directa con el servicio
Beneficiarios del uso de la app:	Los stakeholders están involucrados en el servicio pero no interactúan directamente con el servicio.
Beneficiarios por extensión:	Los stakeholders que tiene poder e influencia sobre el servicio.

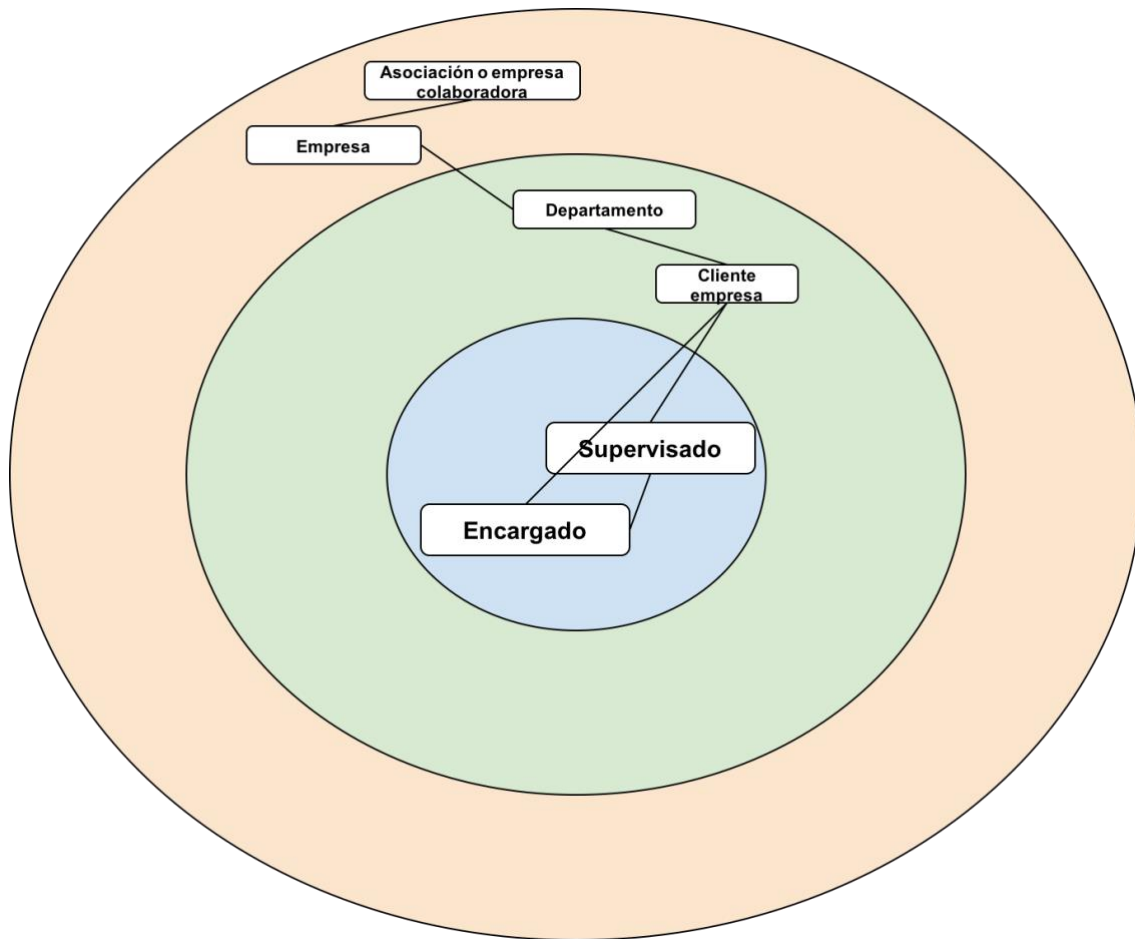


Figura 3: Stakeholders de la aplicación

1.2 Objetivos del Trabajo

El producto

Aplicación que permite la creación de sub-aplicaciones basadas en decisión (preguntas/respuestas) y ejecutarlas en ella como si de una aplicación se tratara.

“La aplicación será para los usuarios como Blog es para usuarios con necesidad de una web”

La ejecución de una sub-aplicación de decisión será como un manual automatizado. Ya no será necesario buscar entre líneas el caso concreto; A base de preguntas y respuestas se llegará al resultado apropiado.

Objetivos

Es posible deducir los objetivos esenciales del trabajo utilizando los objetivos de los usuarios del sistema: el usuario “encargado” y el usuario “supervisado”.

1. El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión.

2. El sistema debe de permitir editar una sub-aplicación.
3. El sistema debe de ser capaz de ejecutar correctamente cualquier sub-aplicación generada.
4. El sistema debe permitir la compartición de una sub-aplicación.
5. El sistema siempre debe estar disponible para los usuarios.
6. El sistema debe estar disponible en el mercado de España.
7. El sistema debe disponer de un diseño visual atractivo que facilite el uso del mismo.
8. El sistema debe de permitir eliminar una sub-aplicación.

De este modo, el trabajo tiene como objetivo principal el proveer una aplicación capaz de crear y ejecutar sub-aplicaciones de tomas de decisión basadas en preguntas-respuestas. Como objetivo secundario se permitirá compartir una sub-aplicación.

Requisitos funcionales

Crear sub-aplicación

La aplicación permitirá al usuario crear su propia sub-aplicación de decisión. Para ello necesitará ser capaz de:

- Crear una pantalla indicando el texto a mostrar y posibles respuestas.
- Editar pantalla (textos, botones...)
- Eliminar pantalla.
- Enlazar la acción de un botón a una pantalla.
- Editar enlace de botón/pantalla.
- Eliminar enlace de botón/pantalla.

Nombre	<i>Crear sub-aplicación</i>		
Número	RF01		
Tipo	Esencial		
Descripción	La aplicación permitirá al usuario crear sus propias aplicaciones basadas en preguntas/respuestas.		
Justificación	Permite al usuario tener una aplicación a su medida.		
Dependencias	-	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita una aplicación a medida de toma de decisiones.		
Precondición	El usuario debe tener instalada la aplicación.		
Condición de satisfacción	El usuario ha creado una sub-aplicación y esta se muestra como disponible.		
Acorde al objetivo	1 - El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión		

Nombre	<i>Crear pantalla sub-aplicación</i>		
Número	RF02		
Tipo	Esencial		
Descripción	La aplicación permitirá al usuario crear pantallas para su sub-aplicación.		
Justificación	Permite al usuario mostrar la información que desea.		

Dependencias	RF01	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita añadir una pantalla en su sub-aplicación.		
Precondición	El usuario debe tener creada una sub-aplicación.		
Condición de satisfacción	El usuario ha creado una pantalla y esta se muestra en la sub-aplicación..		
Acorde al objetivo	1 - El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión 2 - El sistema debe de permitir editar una sub-aplicación.		

Nombre	<i>Mostrar lista de pantallas de la sub-aplicación</i>		
Número	RF00		
Tipo	Esencial		
Descripción	La aplicación mostrará la lista de pantallas de la sub-aplicaciones disponibles.		
Justificación	Para poder modificar una pantalla es necesario poder mostrar la lista de pantallas existentes.		
Dependencias	-	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita modificar una pantalla.		
Precondición	-		
Condición de satisfacción	La aplicación muestra todas las pantallas existentes de la sub-aplicación.		
Acorde al objetivo	1 - El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión. 2 - El sistema debe de permitir editar una sub-aplicación. 8 - El sistema debe de permitir eliminar una sub-aplicación.		

Nombre	<i>Editar pantalla sub-aplicación</i>		
Número	RF03		
Tipo	Esencial		
Descripción	La aplicación permitirá al usuario modificar el contenido de las pantallas de su sub-aplicación.		
Justificación	Permite al usuario modificar la información que desee ya que puede haber cometido algún error.		
Dependencias	RF02,RF00	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita modificar una pantalla de su sub-aplicación.		
Precondición	El usuario debe tener creada la pantalla en una sub-aplicación.		
Condición de satisfacción	El usuario ha modificado la pantalla y esta se muestra en la sub-aplicación..		
Acorde al objetivo	1 - El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión		

	2 - El sistema debe de permitir editar una sub-aplicación.
--	--

Nombre	<i>Eliminar pantalla sub-aplicación</i>		
Número	RF04		
Tipo	Esencial		
Descripción	La aplicación permitirá al usuario eliminar pantallas de su sub-aplicación.		
Justificación	Debe poder eliminar una pantalla ya que puede haberla creado por error o puede que ya no le sea necesaria.		
Dependencias	RF02,RF00	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita eliminar una pantalla en su sub-aplicación.		
Precondición	El usuario debe tener creada la pantalla en una sub-aplicación.		
Condición de satisfacción	El usuario ha eliminado una pantalla y ésta ya no se muestra en la sub-aplicación..		
Acorde al objetivo	1 - El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión 2 - El sistema debe de permitir editar una sub-aplicación.		

Nombre	<i>Enlazar pantallas</i>		
Número	RF05		
Tipo	Esencial		
Descripción	La aplicación permitirá al usuario realizar enlace de pantallas mediante botones.		
Justificación	El usuario quiere permitir el acceso a una pantalla mediante un botón.		
Dependencias	RF02,RF00	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita enlazar pantallas.		
Precondición	El usuario debe tener creada la pantalla origen en una sub-aplicación.		
Condición de satisfacción	El usuario ha creado el enlace entre pantallas, permitiendo la creación en blanco de la pantalla destino.		
Acorde al objetivo	1 - El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión 2 - El sistema debe de permitir editar una sub-aplicación.		

Nombre	<i>Editar enlace de pantallas</i>		
Número	RF06		
Tipo	Esencial		
Descripción	La aplicación permitirá al usuario editar el enlace de pantallas mediante botones.		
Justificación	El usuario puede haber cometido un error a la hora de enlazar pantallas.		
Dependencias	RF05,RF00	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita modificar un enlace entre pantallas.		

Precondición	El usuario debe tener creada la pantalla con un botón con enlace.
Condición de satisfacción	El usuario ha modificado el enlace entre pantallas.
Acorde al objetivo	1 - El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión 2 - El sistema debe de permitir editar una sub-aplicación.

Nombre	<i>Eliminar enlace de pantallas</i>		
Número	RF07		
Tipo	Esencial		
Descripción	La aplicación permitirá al usuario editar el enlace de pantallas mediante botones.		
Justificación	El usuario puede haber cometido un error a la hora de enlazar pantallas.		
Dependencias	RF05,RF00	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita modificar un enlace entre pantallas.		
Precondición	El usuario debe tener creada la pantalla con un botón con enlace.		
Condición de satisfacción	El usuario ha modificado el enlace entre pantallas.		
Acorde al objetivo	1 - El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión 2 - El sistema debe de permitir editar una sub-aplicación.		

Editar sub-aplicación

Una vez creada una sub-aplicación, ésta siempre se podrá modificar.

Nombre	<i>Editar sub-aplicación</i>		
Número	RF08		
Tipo	Esencial		
Descripción	La aplicación permitirá al usuario editar sus propias sub-aplicaciones basadas en preguntas/respuestas.		
Justificación	En caso de error o nuevos datos, el usuario debe ser capaz de modificar la sub-aplicación.		
Dependencias	RF01, RF02, RF03, RF04, RF05, RF06, RF07	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita modificar la aplicación a medida de toma de decisiones.		
Precondición	El usuario debe tener instalada creara la sub-aplicación a modificar.		
Condición de satisfacción	El usuario ha modificado sub-aplicación y esta se muestra como disponible.		
Acorde al objetivo	2 - El sistema debe de permitir editar una sub-aplicación.		

Eliminar sub-aplicación

Se eliminará del sistema y no podrá recuperarse.

Nombre	<i>Eliminar sub-aplicación</i>		
Número	RF08		
Tipo	Esencial		
Descripción	La aplicación permitirá al usuario editar eliminar sub-aplicaciones basadas en preguntas/respuestas.		
Justificación	En caso de no querer usar más la sub-aplicación el usuario debe poder borrarla.		
Dependencias	RF01	Conflictos	-
Actor	Encargado		
Evento disparador	El usuario necesita quiere borrar la sub-aplicación.		
Precondición	El usuario debe tener creada la sub-aplicación.		
Condición de satisfacción	El usuario ha eliminado la sub-aplicación y ésta ya no se muestra.		
Acorde al objetivo	8 - El sistema debe de permitir eliminar una sub-aplicación.		

Mostrar lista de sub-aplicaciones

La aplicación será capaz de mostrar la lista de sub-aplicaciones del usuario.

Nombre	<i>Mostrar lista de sub-aplicaciones</i>		
Número	RF09		
Tipo	Esencial		
Descripción	La aplicación mostrará la lista de sub-aplicaciones disponible.		
Justificación	Para poder ejecutar una sub-aplicación o editarla es necesario mostrar su existencia.		
Dependencias	-	Conflictos	-
Actor	Encargado, Supervisado		
Evento disparador	El usuario necesita acceder a una sub-aplicación.		
Precondición	Sub-aplicación previamente creada.		
Condición de satisfacción	La aplicación muestra todas las sub-aplicaciones existentes.		
Acorde al objetivo	1 - El sistema debe de ser capaz de permitir crear sub-aplicaciones de decisión. 2 - El sistema debe de permitir editar una sub-aplicación. 8 - El sistema debe de permitir eliminar una sub-aplicación.		

Ejecutar sub-aplicación

Una vez se seleccione la sub-aplicación, ésta se podrá ejecutar como si de una aplicación se tratara.

Nombre	<i>Ejecutar sub-aplicaciones</i>		
Número	RF10		
Tipo	Esencial		
Descripción	Se ejecutará la sub-aplicación como de una aplicación independiente se tratara.		
Justificación	El usuario quiere poder ejecutar la sub-aplicación para tomar una decisión contestando a una serie de preguntas.		
Dependencias	-	Conflictos	-
Actor	Supervisado		
Evento disparador	El usuario necesita ejecutar la sub-aplicación.		
Precondición	La sub-aplicación debe estar previamente creada.		
Condición de satisfacción	La sub-aplicación se ha ejecutado y tras contestar todas las preguntas el usuario sabe que debe hacer.		
Acorde al objetivo	3 - El sistema debe de ser capaz de ejecutar correctamente cualquier sub-aplicación generada.		

Descargar sub-aplicación

El usuario podrá descargar una sub-aplicación generada con el fin de poder compartirla con usuarios de la misma aplicación.

Nombre	<i>Descargar sub-aplicación</i>		
Número	RF11		
Tipo	Secundaria		
Descripción	Se permitirá la descarga de una sub-aplicación.		
Justificación	El usuario quiere poder compartir una sub-aplicación.		
Dependencias	-	Conflictos	-
Actor	Supervisado		
Evento disparador	El usuario quiere descargar la sub-aplicación.		
Precondición	La sub-aplicación debe estar previamente creada.		
Condición de satisfacción	La sub-aplicación se ha descargado en un fichero que el usuario tiene ahora disponible.		
Acorde al objetivo	4 - El sistema debe permitir la compartición de una sub-aplicación.		

Cargar sub-aplicación

A partir de un fichero, el usuario podrá cargar una sub-aplicación generada por la aplicación y usarla.

Nombre	<i>Cargar sub-aplicación</i>		
Número	RF12		
Tipo	Secundaria		
Descripción	Se permitirá la carga de una sub-aplicación.		
Justificación	El usuario quiere poder usar una sub-aplicación creada por otro usuario.		
Dependencias	-	Conflictos	-
Actor	Supervisado		
Evento disparador	El usuario quiere cargar la sub-aplicación para su uso.		
Precondición	-		
Condición de satisfacción	La sub-aplicación se ha cargado y ahora se encuentra disponible para su uso.		
Acorde al objetivo	4 - El sistema debe permitir la compartición de una sub-aplicación.		

Requisitos no funcionales

La aplicación pretende ser estable, portable, usable, fácil de mantener, y sobretodo que sea escalable y extensible.

Nombre	<i>Localización en España</i>		
Número	RNF01		
Descripción	La aplicación debe ser como mínimo en español.		
Justificación	Para asegurar el uso de la aplicación en España, la lengua de la aplicación debe ser al menos entendible por todos los habitantes del país.		
Dependencias	-	Conflictos	-
Condición de satisfacción	Si el sistema usa correctamente el lenguaje, los usuarios no deben tener problemas en entenderlo.		
Acorde al objetivo	6 - El sistema debe estar disponible en el mercado de España.		
Expectaciones necesarias	La mayoría de la población residente en España entiende el español.		

Nombre	<i>Disponibilidad de la aplicación</i>		
Número	RNF02		
Descripción	La aplicación debe ser usable en cualquier momento y lugar.		
Justificación	Para asegurar el uso de la aplicación en cualquier localización, la aplicación deberá ser portable.		
Dependencias	-	Conflictos	-
Condición de satisfacción	Si la aplicación es portable, los usuarios podrán acceder desde cualquier sitio.		
Acorde al objetivo	5 - El sistema siempre debe estar disponible para los usuarios.		

Expectaciones necesarias	La mayoría de la población de España dispone de Smartphone.
---------------------------------	---

Nombre	<i>Soportado en los dispositivos móviles principales</i>		
Número	RNF03		
Descripción	La aplicación debe ser soportada por todos los dispositivos móviles principales: iOS, Android.		
Justificación	Para asegurar el uso de la aplicación por la mayoría de usuarios potenciales, es necesario que la aplicación esté disponible para sus dispositivos.		
Dependencias	-	Conflictos	-
Condición de satisfacción	Como mayor sea el abanico, mayor aceptación tendrá.		
Acorde al objetivo	5 - El sistema siempre debe estar disponible para los usuarios.		
Expectaciones necesarias	La mayoría de la población de España utiliza dispositivos iOS o Android.		

Nombre	<i>Interfaz simple y usable</i>		
Número	RNF04		
Descripción	La aplicación debe ser simple, atractiva y fácil de utilizar por los usuarios.		
Justificación	Los usuarios son atraídos por aplicaciones con claras y estructuradas, dónde la funcionalidad es fácilmente comprensible.		
Dependencias	-	Conflictos	-
Condición de satisfacción	La aplicación utiliza una estructura consistente a lo largo de las distintas pantallas.		
Acorde al objetivo	7 - El sistema debe disponer de un diseño visual atractivo que facilite el uso del mismo.		
Expectaciones necesarias	Los usuarios son atraídos por aplicaciones con claras y estructuradas, dónde la funcionalidad es fácilmente comprensible.		

1.3 Enfoque y método seguido

El trabajo se enfocará en el desarrollo de una aplicación para dispositivos móviles, por lo cual se conocen al menos dos problemas de fragmentación: fragmentación por tamaño de pantalla, y fragmentación por el sistema operativo del dispositivo.

Tamaño de pantalla

Para solucionar el problema de fragmentación de tamaño de pantalla, se realizará el desarrollo de adaptación única, es decir, se producirá una única aplicación que funcione en todos los tamaños de dispositivo sin necesidad de realizar más cambios. Para ello se utilizará la estrategia “todos en uno”, la cual es capaz de conocer la información necesaria para poder adaptarse a todos los dispositivos.

Gracias a esta estrategia, se conseguirá realizar el desarrollo en menos tiempo y que sea más genérico.

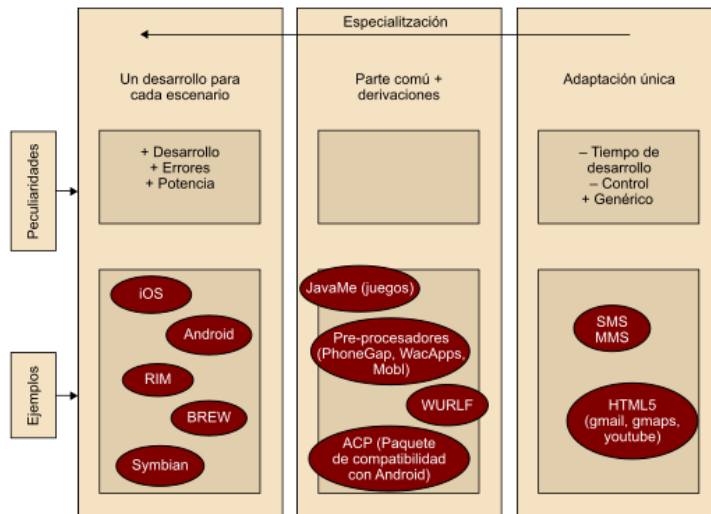


Figura 4: Categorización de tipos de adaptaciones a los problemas de fragmentación.
Fuente: Métodos para el desarrollo de aplicaciones móviles

Sistema operativo

Para solucionar el problema de fragmentación debido al sistema operativo del dispositivo, se seguirá una estrategia de desarrollo híbrido multiplataforma, en la que a partir de un único código se generarán aplicaciones nativas para los principales sistemas operativos. Para ello se utilizará el framework de Ionic.

Ionic es un SDK HTML5, construido sobre Angular, que ayuda a desarrollar aplicaciones móviles nativas utilizando tecnologías web como HTML, CSS y Javascript.

Ionic está principalmente centrada en la capa de presentación de la aplicación por lo que es necesaria la utilización de PhoneGap, Cordova o Trigger.io para el despliegue nativo. También puede ser ejecutado en un navegador como “Progressive Web App”.

Para crear, construir, testear y desplegar una aplicación para cualquier plataforma solo es necesaria la ejecución de un comando.

Ionic presenta una serie de componentes CSS y Javascript con el fin de crear fácilmente aplicaciones. Lo único que se debe hacer para crear una app es programar HTML, CSS y Javascript.

Utilizando esta estrategia se generarán y mantendrán ambas aplicaciones a través de un código único, reduciendo así el tiempo de desarrollo y de mantenimiento.

Situación actual

Actualmente no existe ninguna aplicación para dispositivos móviles capaz de crear aplicaciones de toma de decisión.

La aplicación más parecida es una app web: Zingtree.

Zingtree es una aplicación capaz de generar programas online para la toma de decisiones y se basa en un modelo de negocio de suscripción mensual.



Figura 5: Los números de Zingtree

Factores diferenciales

- Permite crear sub-aplicaciones y utilizarlas como si de una aplicación para dispositivo móvil se tratara.
- Permite compartir el árbol de decisión sin necesidad de estar vinculado a pagos recurrentes.
- No hay necesidad de disponer de conexión a internet para la creación y ejecución de sub-aplicaciones.
- Es gratuito.

Al no existir actualmente ninguna herramienta como la propuesta, se desarrollará un producto totalmente nuevo.

1.4 Planificación del Trabajo

Con el fin de realizar el trabajo, será necesario disponer de un ordenador con el framework de Ionic v2 instalado, un editor de texto, un navegador web con emulador de dispositivo móvil (por ejemplo, Chrome), y de un Smartphone.

El trabajo se realizará con una dedicación de 2 horas diarias entre semana y 6 diarias en fines de semana.

Estrategia de desarrollo

Para el desarrollo de la aplicación de este trabajo, se dispone de unos requisitos muy marcados que deben cumplirse y no van a cambiarse durante el ciclo de su desarrollo. Debido a este motivo, la aplicación se desarrollará siguiendo el modelo “waterfall”, en

la que el proyecto se divide en fases muy marcadas. En este modelo, el desarrollo se interpreta como el agua que va cayendo de un estanque al siguiente. Se le da mucho énfasis a la planificación, a los tiempos, a las fechas límite y al presupuesto.

Ejemplo de fases de un proyecto de *software* en un modelo *waterfall*

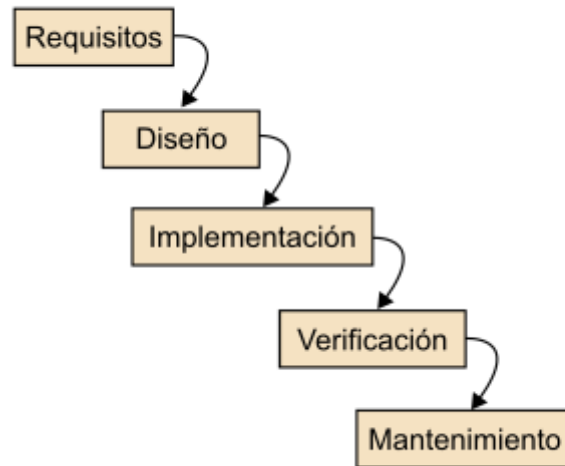


Figura 6: Modelo de desarrollo waterfall.

Fuente: Métodos para el desarrollo de aplicaciones móviles

En el contexto del desarrollo de aplicaciones móviles, el modelo waterfall puede ser aplicable a proyectos realmente controlados y previsibles, en los que no hay mucha incertidumbre por lo que se desea hacer.

En el caso del desarrollo de la aplicación se dispondrá de las siguientes etapas:

- Requisitos
- Diseño
- Implementación
- Verificación

A continuación, se muestra el diagrama de Gantt de la planificación, la cual ha sido dividida según los hitos marcados para el TFM.

Diseño

Durante el diseño de la aplicación, se realizarán las siguientes tareas:

- Investigar los usuarios de la aplicación y recoger requisitos, tanto cuantitativos como cualitativos, que ayudarán a conocer los usuarios y definir perfiles.
- Examinar y analizar las condiciones en que se utilizará el sistema para definir su contexto de uso.
- Elaborar escenarios de uso.
- Definición de los flujos de interacción en el sistema.

- Diseñar y construir un prototipo de alto nivel de la aplicación: se realizará una primera iteración del prototipo, después de ser evaluada, se realizará una segunda iteración, y tras la siguiente evaluación, una última iteración del prototipo.
- Definición de los casos de uso.
- Diseño de la arquitectura de la aplicación.

Tarea	Horas
Usuarios y requisitos	4
Condiciones de uso	4
Escenarios de uso	4
Flujos de uso	4
Prototipo it1	16
Evaluación	2
Prototipo it2	8
Evaluación	2
Prototipo it3	4
Casos de uso	14
Arquitectura	2

*Mientras se realizan las tareas de la etapa, éstas se irán documentando.

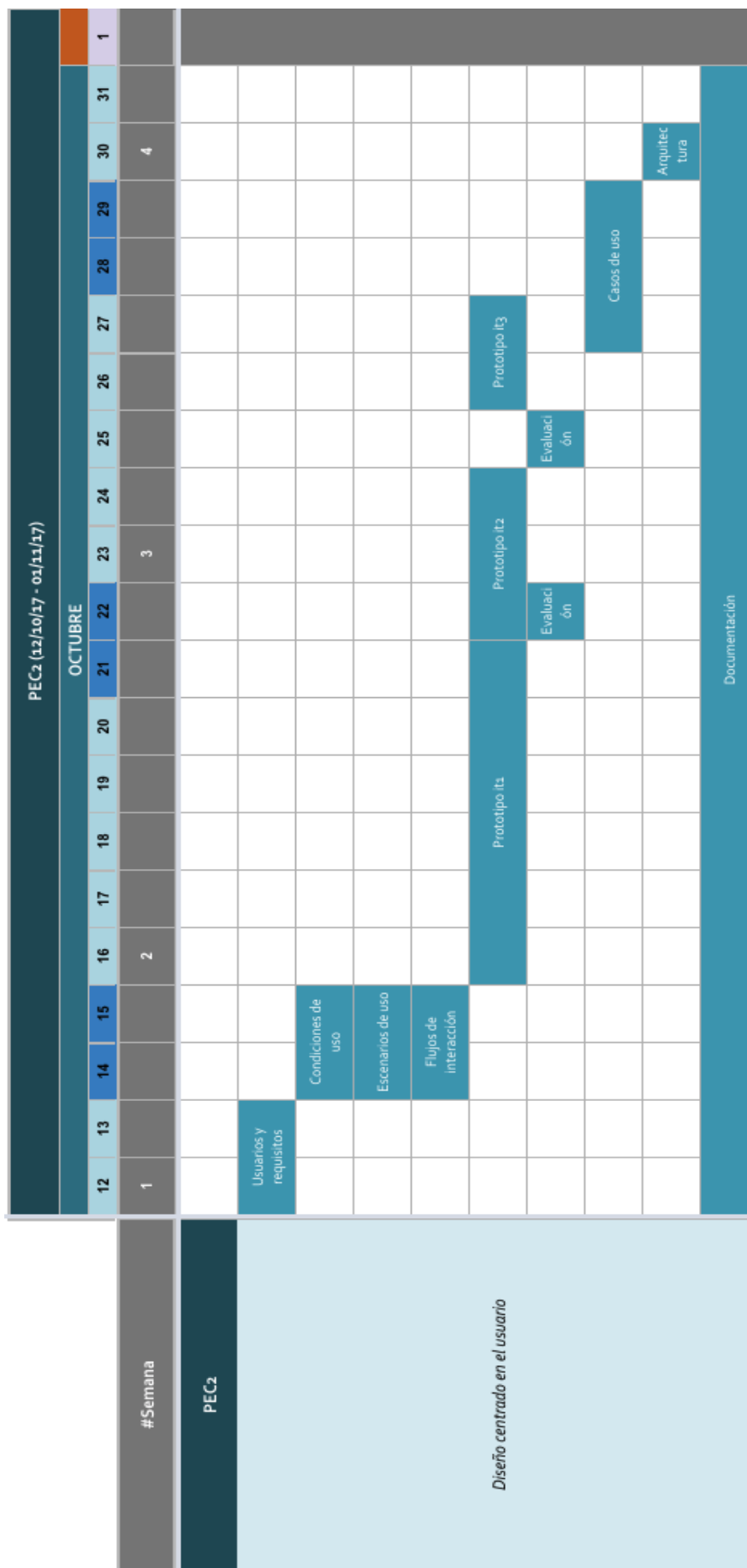


Figura 7: Gantt. Fase de diseño.

Implementación

Durante la fase principal de implementación, se realizará el desarrollo de los objetivos principales de la aplicación, así como de las pruebas necesarias para su correcto funcionamiento.

- Decisión de la estructura de datos que la aplicación generará para almacenar las sub-aplicaciones.
- Implementación del servicio de datos que se encargará de leer, generar, editar, y eliminar la estructura de datos de las sub-aplicaciones.
- Implementación de la pantalla principal.¹
- Implementación de la ejecución de una sub-aplicación dada su estructura de datos². Esta tarea se dividirá en:
 - o Visualización de una pantalla dada su estructura de datos.
 - o Navegación entre pantallas.
- Implementación de carga y descarga de sub-aplicaciones³.
- Implementación de la creación y edición de una sub-aplicación. Esta tarea se dividirá en:
 - o Crear nueva pantalla.
 - o Listar pantallas de la sub-aplicación.
 - o Enlazar pantallas.
 - o Editar pantalla.
 - o Eliminar pantalla y/o componentes de ésta.
 - o Indicar pantalla inicial de la sub-aplicación.⁴
 - o Indicar pantalla enlazada con el botón.⁵
- Implementación del listado de sub-aplicaciones del usuario.

¹ Implementación que no se había tenido en cuenta durante la planificación inicial.

² Implementación después de la pantalla de home y reducida en tiempo.

³ Planificación movida respecto a entregas anteriores. En vez de implementarse durante la PEC3 pasa a implementarse durante la PEC2.

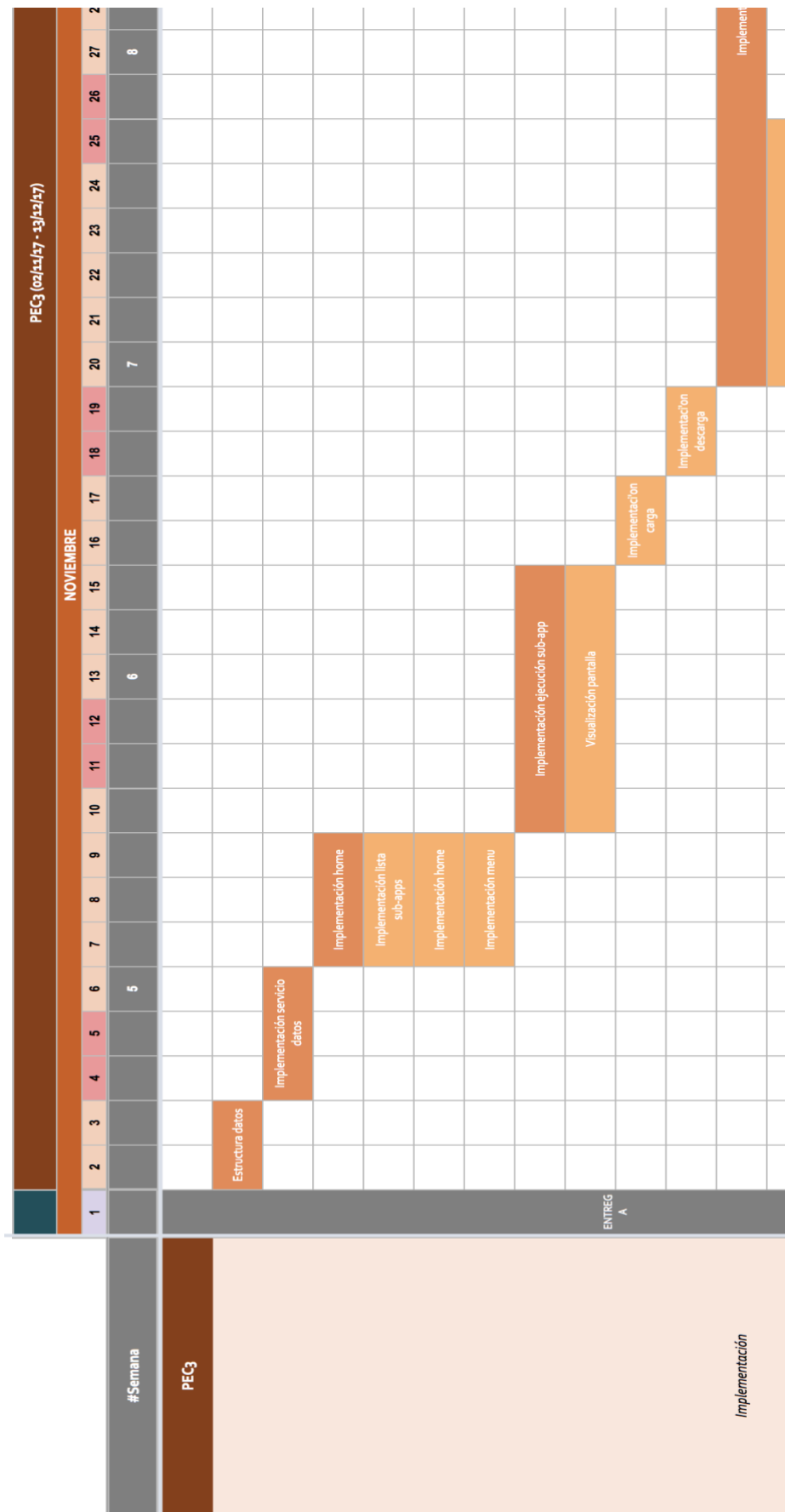
⁴ Nueva funcionalidad no planificada con anterioridad.

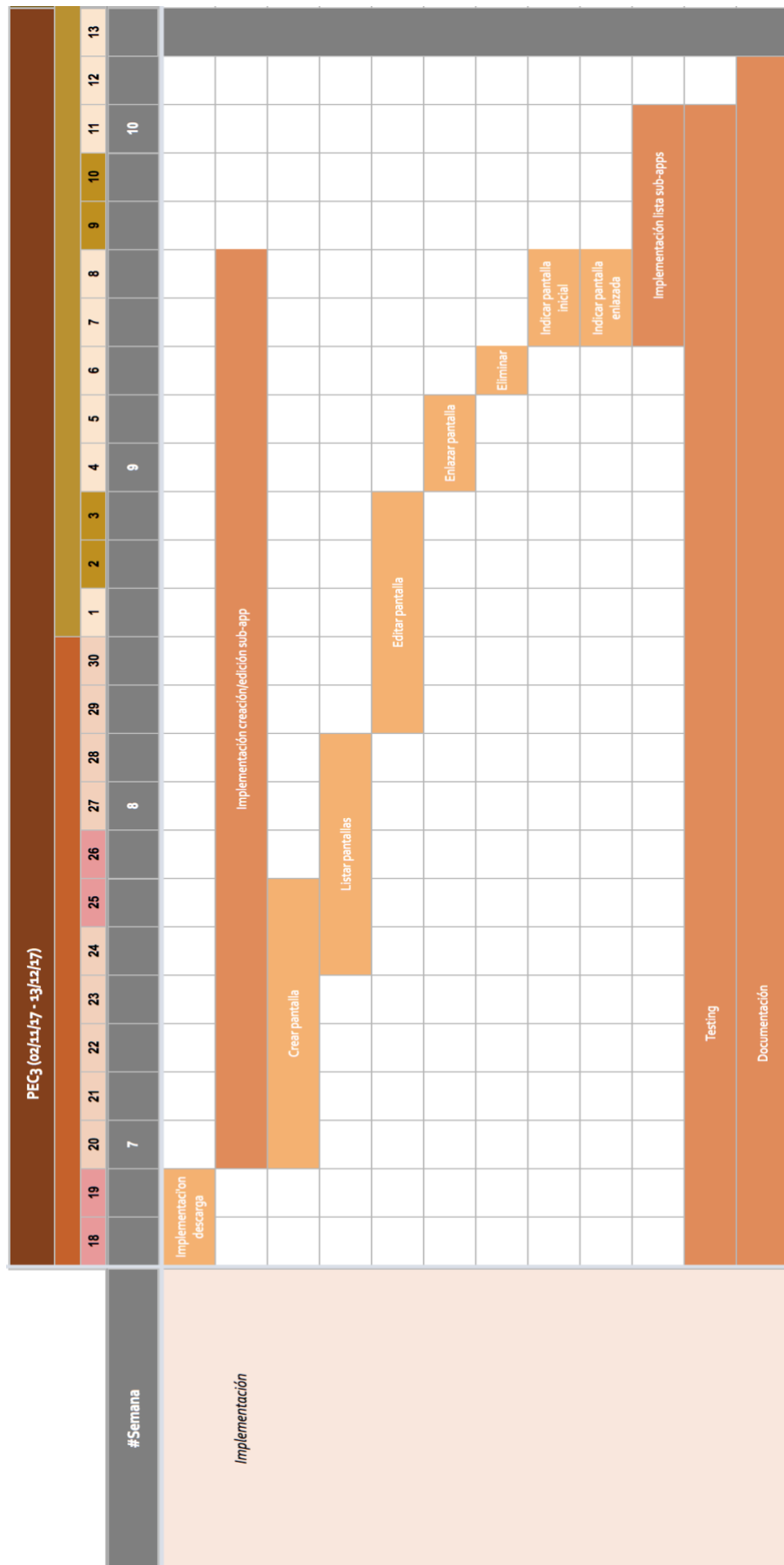
⁵ Nueva funcionalidad no planificada con anterioridad.

Tarea	Horas
Estructura de datos	4
Implementación servicio de datos	14
Implementación home	6
Implementación ejecución sub-app	20
Implementación carga/descarga	16
Implementación crear/editar sub-app	54
Implementación lista sub-apps	15

Sub-tarea	Horas
Visualización pantalla y navegación	20
Crear pantalla	12
Listar pantallas	16
Enlazar pantalla	4
Editar pantalla	18
Eliminar	4
Indicar pantalla inicial	2
Indicar pantalla enlazada	2

*Mientras se realizan las tareas de la etapa, éstas se irán documentando y probando.





Implementación final

En esta última etapa se realizará el desarrollo de los objetivos secundarios (implementación), se realizarán pruebas exhaustivas de la aplicación (verificación), se solucionarán todos los errores que se encuentren, y finalmente, se preparará la presentación del trabajo.

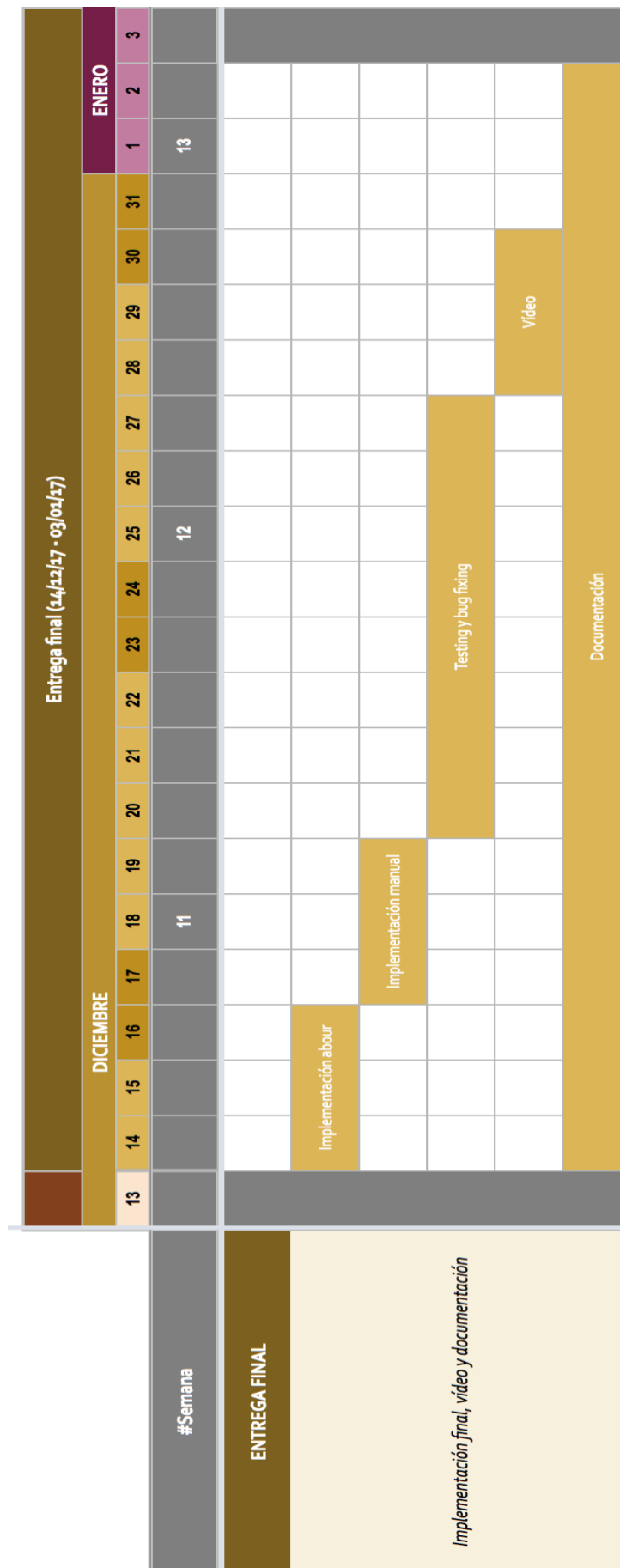
- Implementación de “sobre tDec”.⁶
- Implementación del manual de tDec.⁷
- Pruebas y solución de errores.
- Preparación del vídeo de la presentación.

Tarea	Horas
Implementación de “sobre tDec”	10
Implementación del manual de tDec	10
Test y bug fixing	24
Vídeo	10

*Mientras se realizan las tareas de la etapa, éstas se irán documentando.

⁶ Cambio respecto a la planificación inicial. Descarga pasa realizarse durante a la fase de implementación y se añade en esta fase la implementación de la pantalla “sobre tDec”.

⁷ Cambio respecto a la planificación inicial. Carga de sub-aplicación pasa a realizarse durante la fase de implementación y se añade en esta fase la implementación de la pantalla “manual”.



1.5 Breve resumen de productos obtenidos

Una vez finalizado el trabajo, se dispondrá de una aplicación Ionic lista para ser empaquetada para iOS y Android.

Esta aplicación será capaz de generar sub-aplicaciones y ejecutarlas.

1.6 Breve descripción de los otros capítulos de la memoria

El documento está estructurado con los siguientes capítulos.

- Usuarios y contextos de uso: se estudian las características de los usuarios, sus necesidades y objetivos, así como el contexto de uso.
- Diseño conceptual: se detallan los escenarios de uso de la aplicación y los flujos de interacción entre el usuario y el sistema.
- Prototipo: se muestra el prototipo de las pantallas principales de la aplicación.
- Casos de uso: se describen los casos de uso del usuario con la aplicación, así como su diagrama UML.
- Diseño de arquitectura: se describe cómo se almacenarán los datos, la estructura de las clases a utilizar para la implementación de la aplicación y la arquitectura del sistema.
- Implementación: se explican las decisiones de implementación que se han ido tomando durante el desarrollo, así como los imprevistos surgidos.
- Pruebas: se definen las pruebas a realizar con tal de comprobar el correcto funcionamiento de la aplicación.
- Conclusiones: conclusiones extraídas de la realización del trabajo, así como explicación de futuras mejoras que podría tener la aplicación.
- Glosario: definición de los términos y acrónimos más relevantes.
- Bibliografía consultada.
- Anexos: listado de apartados demasiado extensos o con formato no compatible para haberse incluido en la memoria.

2. Usuarios y contexto de uso

Tal y como se ha comentado anteriormente, los usuarios de la aplicación serán aquellas personas que se encuentren con la necesidad de saber qué acción realizar a partir de una serie de preguntas y sus respuestas. Por ejemplo, médicos queriendo saber si un paciente padece o no una enfermedad.

Los usuarios potenciales de la aplicación son:

- Encargados (Jefes)
- Supervisados (Trabajadores)
- Independientes para uso personal

Estos usuarios necesitarán disponer de una serie de pautas y pasos a realizar según los datos que tengan.

La aplicación será utilizada por el propio individuo, gente que se encuentre con el mismo problema, o en el lugar de trabajo a cualquier hora del día (24/7).

Un usuario podrá crear una sub-aplicación para auto-ayudarse o distribuirla. Utilizándose dicha aplicación cuando el usuario se encuentre con un problema y éste deba ser clasificado.

El uso de la aplicación será de pocos minutos por problema/decisión.

Hay que tener en cuenta que los usuarios pueden encontrarse en constante movimiento (un médico puede ir moviéndose de sala, un electricista puede cambiar de casa...). Por este motivo, la aplicación debe ser siempre accesible desde cualquier lugar (ya sea en el interior o en el exterior).

El usuario dispondrá de un smartphone o tablet personal sin necesidad de conexión a internet. La visualización de estos dispositivos será vertical para su mayor comodidad.

Es probable que el profesional sea interrumpido por personas o eventos de su alrededor, por lo que se diseñará de tal forma que sea fácil de retomar la acción. La luz que recibirá el dispositivo permitirá que la pantalla se vea claramente.

Los usuarios pretenden lograr una correcta clasificación del problema con el fin de obtener una mayor eficacia a la hora de tomar una decisión y por lo tanto resolver un problema.

2.1 Fichas de usuario

A continuación, se muestran tres fichas de posibles usuarios de la aplicación.

Ficha de usuario I

Nombre: M^a Salud Redondo

Edad: 46 años

Profesión: Médico contratado por turnos



Figura 11: Usuario 1

M^a Salud Redondo tiene 46 años es médico especialista que trabaja por turnos en urgencias de un gran hospital de Barcelona. Tiene que realizar tres guardias a la semana de 24h, con un equipo constituido por residentes en diferentes años de formación de distintas especialidades. Sobre todo trabaja en el área de enfermo grave y de pacientes que presentan dolor torácico. Utiliza habitualmente el ordenador para su trabajo diario, es usuaria habitual de aplicaciones móviles. M^a Salud se ha dado cuenta de que la actuaciones sobre una misma patología difieren entres distintos compañeros.

Quiero disponer de información rápida y actualizada para el manejo de la clasificación de los pacientes con dolor torácico.

Objetivos

- Ser capaz de clasificar un paciente con dolor torácico de forma rápida y eficaz.
- Mejorar la atención del paciente con dolor torácico.

Comportamientos

- Dispone de un smartphone y tablet Android.
- Exigente con la calidad y actualidad de la información.

Necesidades

- Acceso inmediato al proceso de clasificación del dolor torácico.

Ficha de usuario II

Nombre: Cristina Díaz

Edad: 28 años

Profesión: Médico residente de cuarto año



Figura 12: Usuario 2

Cristina Díaz es médico residente de cuarto año de medicina interna. Realiza guardias de 24h entre 3 y 4 veces al mes en el servicio de Urgencias del hospital, donde se forma con otros compañeros de menor rango y un adjunto asignado a la planta de urgencias donde esté asignado. Los médicos de guardia tienen formación distinta (tanto por especialidad como por años de experiencia) y piensa que sería útil de cara a poder ofrecer mejor calidad asistencial algún recurso que facilite la actuación en casos de mayor gravedad como en el dolor torácico.

Quiero disponer de una aplicación que facilite la actuación en casos de dolor torácico.

Objetivos

- Ser capaz de clasificar y atender un paciente con dolor torácico de forma rápida y eficaz.
- Mejorar la atención del paciente con dolor torácico.

Comportamientos

- Dispone de un smartphone LG y de una tablet Samsung Galaxy Tab A (ambos Android).
- Exigente a la hora de diagnosticar pacientes.

Necesidades

- Acceso inmediato al proceso de clasificación y tratamiento del dolor torácico.
- El proceso debe ser sencillo de entender y utilizar.

Ficha de usuario III

Nombre: Julian Hernando

Edad: 35 años

Profesión: Ingeniero multidisciplinar

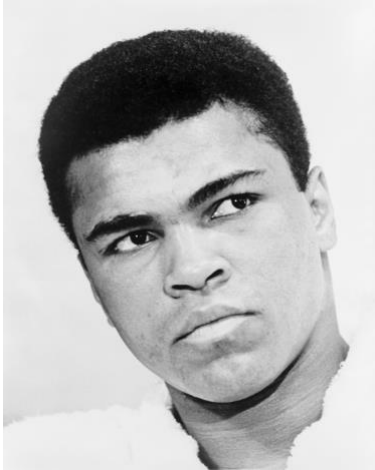


Figura 13: Usuario 3

Julian Hernando es un ingeniero emprendedor que acaba de crear un prototipo de un producto que necesita instalarse. Acostumbra a trabajar de 8 a 17, pero si le surgen imprevistos puede modificar su horario de trabajo sin problemas. Las instalaciones del producto no son perfectas, y a veces surgen problemas que deben solucionarse. Para solucionarlos, piensa que sería útil poder diagnosticar el problema rápidamente con algún recurso que le facilite información de cómo actuar.

Quiero disponer de mi propia aplicación para solucionar problemas de instalación del producto.

Objetivos

- Poder crear tu propio asistente de problemas de instalación..
- Ser capaz de diagnosticar el problema y solucionarlo.
- Mejorar la rapidez de detección y solución del problema.

Comportamientos

- Dispone de iPhone 6.
- Exigente a la hora de verificar la funcionalidad del producto.

Necesidades

- Acceso inmediato al proceso de diagnóstico.
- El proceso debe ser sencillo de entender y utilizar.
- Debe poder crear el asistente de forma fácil.

3. Diseño conceptual

En este apartado se describen los escenarios de uso de la aplicación. Es decir, se describe desde el punto de vista del usuario cómo utilizará el producto en un contexto concreto. Posteriormente, se describen los flujos de interacción entre el usuario y la aplicación.

3.1 Escenarios de uso

A continuación, se describen escenarios dónde el usuario interacciona con la aplicación.

Crear sub-aplicación

M^a Salud ha estado hoy en una reunión de servicio dónde se ha decidido que se creará una sub-aplicación para el diagnóstico de dolor torácico. En dicha reunión se le ha encargado a M^a Salud crear dicha aplicación. Por lo que, después de realizar sus rondas, abre la aplicación y selecciona la opción de crear una sub-aplicación. Durante la tarde genera todas las pantallas necesarias y las enlaza entre sí. Una vez ha terminado, se dirige al restaurante donde tenía prevista una cita.

Julian ha estado toda la mañana instalando el producto. La instalación ha sido tres horas más largas de lo habitual ya que se ha encontrado con un error de instalación que ha sido difícil de detectar. Después de la tediosa instalación ha decidido que ya ha tenido suficiente, por lo que decide crear su propio asistente de solución de problemas de instalación. Después de pensar y mapear todos los posibles casos de error que conoce, accede a la app para crear una sub-aplicación de asistencia de errores. Una vez ha creado todas las pantallas, ya puede utilizar el asistente como si de una aplicación se tratara desde cualquier parte del mundo. Ahora será capaz de encontrar los problemas de instalación más fácilmente.

Editar sub-aplicación

Después de estar toda la mañana intentando instalar el producto y no obteniendo ningún diagnóstico concreto en la aplicación, Julian encuentra la causa del error. Es la primera vez que se encuentra con este tipo de error por lo que decide incorporarlo en el asistente que creó. Para ello, accede a la app y edita el asistente añadiendo las nuevas pantallas. Ahora, si el problema surge de nuevo, el asistente le indicará que puede ser por este motivo. Contento, cierra la aplicación y prosigue con su trabajo.

Editar información de la sub-aplicación

Julian, al utilizar el asistente de instalación que creó, se ha percatado de que tiene una falta ortográfica en el título de la sub-aplicación. Debido a ello, después de finalizar la instalación, accede a la app y modifica el título del asistente, corrigiendo así la falta ortográfica. Ahora, aparecerá siempre el título bien escrito.

Ejecutar sub-aplicación

Cristina lleva de guardia desde las 8am del día previo en el nivel 2 de su servicio de urgencias atendiendo pacientes graves, principalmente aquellos que consultan por dolor torácico con la supervisión/ayuda del adjunto responsable. Son las 5 de la mañana, se ha partido la guardia, lo que significa que en urgencias sólo se encuentra ella y un residente de primer año de dermatología, y consulta un paciente por dolor torácico de características típicas de isquemia. Recuerda el algoritmo de actuación en estos casos y además hace poco el servicio del hospital ha puesto en marcha una sub-aplicación disponible para tablets y móviles que permite seguir el algoritmo de actuación en estos casos. Abre la sub-aplicación y sigue las indicaciones, lo que hace que esté más segura en la toma de decisiones y no olvide ningún paso a la hora del tratamiento.

Julian está realizando una nueva instalación del producto, pero le ha surgido un error. Después de probar un par de posibles razones y no solucionar el problema, ha decidido utilizar el asistente que creó con el fin de no perder más el tiempo. Julian empieza a interactuar con la aplicación contestando a las preguntas que le aparecen. Finalmente, la aplicación le hace una pregunta y ve que ahí radica el problema. La aplicación le indica los pasos de cómo solucionarlo y los lleva a término. De este modo, Julian no ha perdido el tiempo pensando en todas las posibles razones de error de instalación y lo ha podido solucionar de forma rápida.

Eliminar sub-aplicación

Julian lleva los últimos 3 meses mejorando el producto que creó. Esto ha provocado que el proceso de instalación de éste ha cambiado tanto que el asistente que generó ya no le es útil. Esta mañana ha decidido generar el nuevo asistente por lo que ahora piensa que estaría bien eliminar el anterior para evitar confusiones. Julian coge su dispositivo móvil, abre tDec, selecciona el asistente antiguo y tras comprobar que es el correcto, lo elimina. Ahora Julian no se confundirá de asistente y utilizará siempre el correcto. Así pues, va a la cocina, coge un vaso de agua, y se dirige de nuevo a su escritorio para continuar trabajando.

Descargar sub-aplicación

M^a Salud ha obtenido hoy el visto bueno del asistente de diagnóstico que generó, por lo que se decide que la aplicación empezará a usarse el siguiente lunes. Para ello, redacta un e-mail a sus compañeros de servicio indicando que deben descargarse la aplicación tDec y cargar en ella la sub-aplicación adjuntada. Así pues, M^a Salud, accede a la aplicación y se descarga el asistente de diagnóstico adjuntándolo posteriormente al e-mail. Una vez enviado el mensaje, M^a Salud recoge sus pertenencias de la taquilla y se va contenta a casa. Ahora, todos los compañeros realizarán las mismas acciones al tratar una misma patología.

Cargar sub-aplicación

Cristina ha entrado a trabajar a las ocho de la mañana y le indican que hay una mini-reunión del servicio. En él, indican que todos los médicos deben utilizar una aplicación para el diagnóstico del dolor torácico y que encontrarán la información y los datos en un e-mail que se les ha sido enviado. Cristina abre el e-mail, lee la información que se le indica y finalmente se descarga la aplicación tDec. A través de ella, carga la sub-aplicación que estaba adjuntada en el e-mail, y Cristina ve con satisfacción como ahora puede usar la aplicación de diagnóstico.

Crear pantalla

M^a Salud ha estado probando la sub-aplicación de diagnóstico con sus compañeros de servicio durante un mes, y se han dado cuenta que estaría bien disponer de una lista de pasos a seguir después de un diagnóstico en concreto. Después de hablarlo con la junta, se ha acordado que se añadirá esta nueva pantalla y se redistribuirá la aplicación. Así pues, M^a Salud, justo antes de ir a casa, coge la aplicación, entra en modo edición de tDec y selecciona la opción de crear pantalla. Allí rellena todos los pasos que se deberá seguir y la guarda, a espera de ser enlazada.

Editar pantalla

Mientras Julian está intentando encontrar la razón del error de instalación a través de su asistente, se da cuenta de que en una de las pantallas tiene tal error ortográfico que le duelen los ojos, por lo que se apunta qué pantalla es para después modificarla. Una vez Julian ha finalizado la instalación, accede a través de tDec al modo edición de su asistente e indica que quiere modificar la pantalla que había detectado con el error. Julian corrige el error y guarda la pantalla. Ahora, feliz, sabe que ya no volverá a ver ese desgarrador error ortográfico.

Eliminar pantalla

M^a Salud ha sido notificada de que el cambio que se hizo en la aplicación para añadir los pasos a seguir tras un diagnóstico no está funcionando como tenían previsto, de tal modo que se ha decidido que la pantalla debe eliminarse. Tras realizar la ronda de urgencias, M^a Salud tiene unos minutos libres por lo que decide que realizará la actualización. Tras coger su dispositivo móvil, accede a tDec y desde el apartado de edición accede a la pantalla en cuestión y la elimina. Más tarde redistribuirá la aplicación ya que ahora se dirige a realizar otra ronda.

Crear enlace

M^a Salud ha estado probando la sub-aplicación de diagnóstico con sus compañeros de servicio durante un mes, y se han dado cuenta que estaría bien disponer de una lista de

pasos a seguir después de un diagnóstico en concreto. Después de hablarlo con la junta, se ha acordado que se añadirá esta nueva pantalla y se redistribuirá la aplicación. Así pues, M^a Salud, justo después de crear la nueva pantalla, se dirige a la pantalla desde la que se accederá, crea un botón y selecciona la opción de crear un enlace. Al hacerlo, M^a Salud indica que la pantalla enlazada será la que había creado anteriormente y guarda la pantalla. Una vez se comparte la aplicación, todos los médicos verán los pasos que deben seguir después del diagnóstico.

Editar enlace

Mientras Julian está intentando encontrar la razón de un error de instalación a través de su asistente, se da cuenta de que la respuesta en una pantalla le está enviando a una pantalla que no tiene sentido, por lo que se apunta qué pantalla es para después modificarla. Una vez Julian ha finalizado la instalación, accede a través de tDec al modo edición de su asistente e indica que quiere modificar la pantalla que había detectado con el error. Julian selecciona la opción de editar enlace del botón, selecciona la pantalla correcta, y guarda los cambios realizados. Ahora, cuando acceda de nuevo a la aplicación y se encuentre con el mismo problema de instalación, el asistente le redireccionará a la pantalla correcta.

Eliminar enlace

Julian, al utilizar la aplicación para detectar el problema que tiene de la instalación, ha obtenido la razón por la que falla y ha apretado el botón de finalizar sin leer el texto. Este botón le ha redireccionado a la primera pantalla, y él sin darse cuenta, contesta a la pregunta que se le indica. Después de contestar a un par de preguntas, se da cuenta de que está dando vueltas a la aplicación, por lo que decide que no quiere que el último botón le redirija a la primera pantalla. Tras solucionar el problema de instalación, entra a tDec y a través del modo edición entra a la pantalla en cuestión y tras seleccionar el botón, le indica que desea eliminar el enlace. Ahora, cuando Julian utilice la aplicación ya no volverá a dar vueltas a la aplicación sin darse cuenta. Le queda por decidir si dejar el botón o directamente eliminarlo.

3.2 Flujos de interacción

A continuación, se muestran los flujos de interacción entre el usuario y la aplicación a la hora de satisfacer las necesidades del usuario.

3.2.1 Crear sub-aplicación

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales con la opción de crear una nueva.
3. El usuario indica que quiere crear una sub-aplicación nueva.
4. El sistema le muestra una pantalla indicando que debe rellenar los datos generales de la sub-aplicación.
5. El usuario rellena el formulario e indica que quiere proceder con la creación de la sub-aplicación.
6. El sistema le muestra la pantalla de información con las posibles opciones a realizar. Entre ellas editar.
7. El usuario selecciona la opción de editar sub-aplicación.
8. El sistema le muestra la lista de todas las pantallas de la sub-aplicación con la opción de crear una nueva.
9. El usuario indica que quiere crear una pantalla nueva.
10. El sistema le muestra una pantalla donde puede rellenarla con los datos y botones que desea.
11. El usuario rellena todos los datos que quiere mostrar (preguntas, checklists, botones)
12. Para cada botón creado, el usuario presiona sobre él para realizar un enlace.
13. El sistema muestra todas las pantallas disponibles y la opción de pantalla nueva.
 - a. El usuario escoge la pantalla a la que enlazar,
 - i. El sistema le indica que ha sido enlazada y vuelve a la pantalla de edición a la que se encontraba.
 - b. El usuario decide que enlazará a una pantalla nueva.
 - i. El sistema le indica que introduzca el nombre de la pantalla.
 - ii. El usuario introduce el nombre.
 - iii. El sistema crea la pantalla y le redirecciona a la pantalla de edición a la que se encontraba.
14. Una vez el usuario ha introducido todos los datos, guarda la pantalla.
15. El sistema le redirecciona a 6, hasta que el usuario finaliza la creación de la sub-aplicación.

3.2.2 Editar sub-aplicación

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales con la opción de crear una nueva.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que quiere editar la sub-aplicación.

6. El sistema le muestra la lista de todas las pantallas de la sub-aplicación con la opción de crear una nueva.
7. El usuario selecciona la pantalla a editar o crear pantalla.
8. El sistema le muestra una pantalla con la información de la pantalla.
9. El usuario modifica los datos que desea (preguntas, checklists, botones)
10. Para editar un enlace, el usuario presiona sobre el botón pertinente.
11. El sistema muestra todas las pantallas disponibles y la opción de pantalla nueva.
 - a. El usuario escoge la pantalla a la que enlazar,
 - i. El sistema le indica que ha sido enlazada y vuelve a la pantalla de edición a la que se encontraba.
 - b. El usuario decide que enlazará a una pantalla nueva.
 - i. El sistema le indica que introduzca el nombre de la pantalla.
 - ii. El usuario introduce el nombre.
 - iii. El sistema crea la pantalla y le redirecciona a la pantalla de edición a la que se encontraba.
12. Una vez el usuario ha introducido todos los datos, guarda la pantalla.
13. El sistema le redirecciona a 6, hasta que el usuario finaliza la edición de la sub-aplicación.

3.2.3 Editar información de la sub-aplicación

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales con la opción de crear una nueva.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que quiere editar la información de la sub-aplicación.
6. El sistema muestra una pantalla con el formulario de creación/edición.
7. El usuario modifica los datos y oprime el botón de guardar.
8. El sistema almacena los cambios y redirecciona al usuario a la pantalla de información.

3.2.4 Ejecutar sub-aplicación

1. El usuario se dirige a la sección de sub-aplicaciones.
2. El sistema muestra todas las sub-aplicaciones disponibles.
3. El usuario selecciona la sub-aplicación que desea ejecutar.
4. El sistema muestra la información general de la sub-aplicación.
5. El usuario indica que quiere ejecutarla.
6. El sistema muestra una pantalla con información y las acciones/respuestas posibles.
7. El usuario indica la acción/respuesta pertinente.
8. El sistema le redirecciona a 6 hasta que la sub-aplicación finaliza.

3.2.5 Eliminar sub-aplicación

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales con la opción de crear una nueva.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que desea eliminarla.
6. El sistema pregunta si está seguro, y en caso afirmativo elimina la sub-aplicación.

3.2.6 Descargar sub-aplicación

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales con la opción de crear una nueva.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que desea descargarla.
6. El sistema prepara el fichero y este se descarga.

3.2.7 Cargar sub-aplicación

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales con la opción de crear una nueva, o cargarla.
3. El usuario selecciona la opción de cargar fichero.
4. El sistema le pide que indique el fichero.
5. El usuario indica el fichero.
6. El sistema realiza la carga y le muestra la información general de la aplicación.

3.2.8 Crear pantalla

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que quiere editar la sub-aplicación.
6. El sistema le muestra la lista de todas las pantallas de la sub-aplicación con la opción de crear una nueva.
7. El usuario selecciona crear una pantalla nueva.
8. El sistema le muestra una pantalla donde puede rellenarla con los datos y botones que desea.

9. El usuario rellena todos los datos que quiere mostrar (preguntas, checklists, botones) y guarda.
10. El sistema guarda la nueva pantalla en la sub-aplicación.

3.2.9 Editar pantalla

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que quiere editar la sub-aplicación.
6. El sistema le muestra la lista de todas las pantallas de la sub-aplicación con la opción de crear una nueva.
7. El usuario selecciona la pantalla que quiere editar.
8. El sistema le muestra la pantalla con los datos y botones que tiene.
9. El usuario realiza las modificaciones que deseaba hacer y guarda los cambios.
10. El sistema guarda la modificación de la pantalla en la sub-aplicación.

3.2.10 Eliminar pantalla

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que quiere editar la sub-aplicación.
6. El sistema le muestra la lista de todas las pantallas de la sub-aplicación con la opción de crear una nueva.
7. El usuario selecciona la pantalla que quiere eliminar.
8. El sistema le muestra la pantalla con los datos y botones que tiene, así como la opción de eliminarla.
9. El usuario selecciona que quiere eliminar la pantalla.
10. El sistema tras preguntar al usuario si está seguro, borra la pantalla de la sub-aplicación.

3.2.11 Crear enlace

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que quiere editar la sub-aplicación.
6. El sistema le muestra la lista de todas las pantallas de la sub-aplicación con la opción de crear una nueva.
7. El usuario selecciona la pantalla que quiere editar.
8. El sistema le muestra la pantalla con los datos y botones que tiene.

9. El usuario crea un nuevo botón y selecciona que desea enlazarlo.
10. El sistema le muestra la lista de pantallas de la sub-aplicación y la opción de crear una nueva en blanco.
11. El usuario selecciona la pantalla con la que desea realizar el enlace.
12. El sistema realiza el enlace entre el botón y la pantalla seleccionada.

3.2.12 Editar enlace

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que quiere editar la sub-aplicación.
6. El sistema le muestra la lista de todas las pantallas de la sub-aplicación con la opción de crear una nueva.
7. El usuario selecciona la pantalla que quiere editar.
8. El sistema le muestra la pantalla con los datos y botones que tiene.
9. El usuario selecciona un botón e indica que quiere modificar el enlace.
10. El sistema le muestra la lista de pantallas de la sub-aplicación y la opción de crear una nueva en blanco.
11. El usuario selecciona la pantalla con la que desea realizar el enlace.
12. El sistema realiza el enlace entre el botón y la pantalla seleccionada.

3.2.13 Eliminar enlace

1. El usuario se dirige a la sección de edición de sub-aplicaciones.
2. El sistema le muestra una lista con todas las sub-aplicaciones actuales.
3. El usuario selecciona la sub-aplicación a editar.
4. El sistema le muestra una pantalla con los datos generales de la sub-aplicación.
5. El usuario indica que quiere editar la sub-aplicación.
6. El sistema le muestra la lista de todas las pantallas de la sub-aplicación con la opción de crear una nueva.
7. El usuario selecciona la pantalla que quiere editar.
8. El sistema le muestra la pantalla con los datos y botones que tiene.
9. El usuario selecciona un botón e indica que desea eliminar el enlace.
10. El sistema tras preguntar al usuario si está seguro, borra el enlace entre el botón y la pantalla.

3.2 Diagrama de flujo

A partir de los flujos de interacción detallados anteriormente, se han generado el diagrama de flujo de la aplicación.

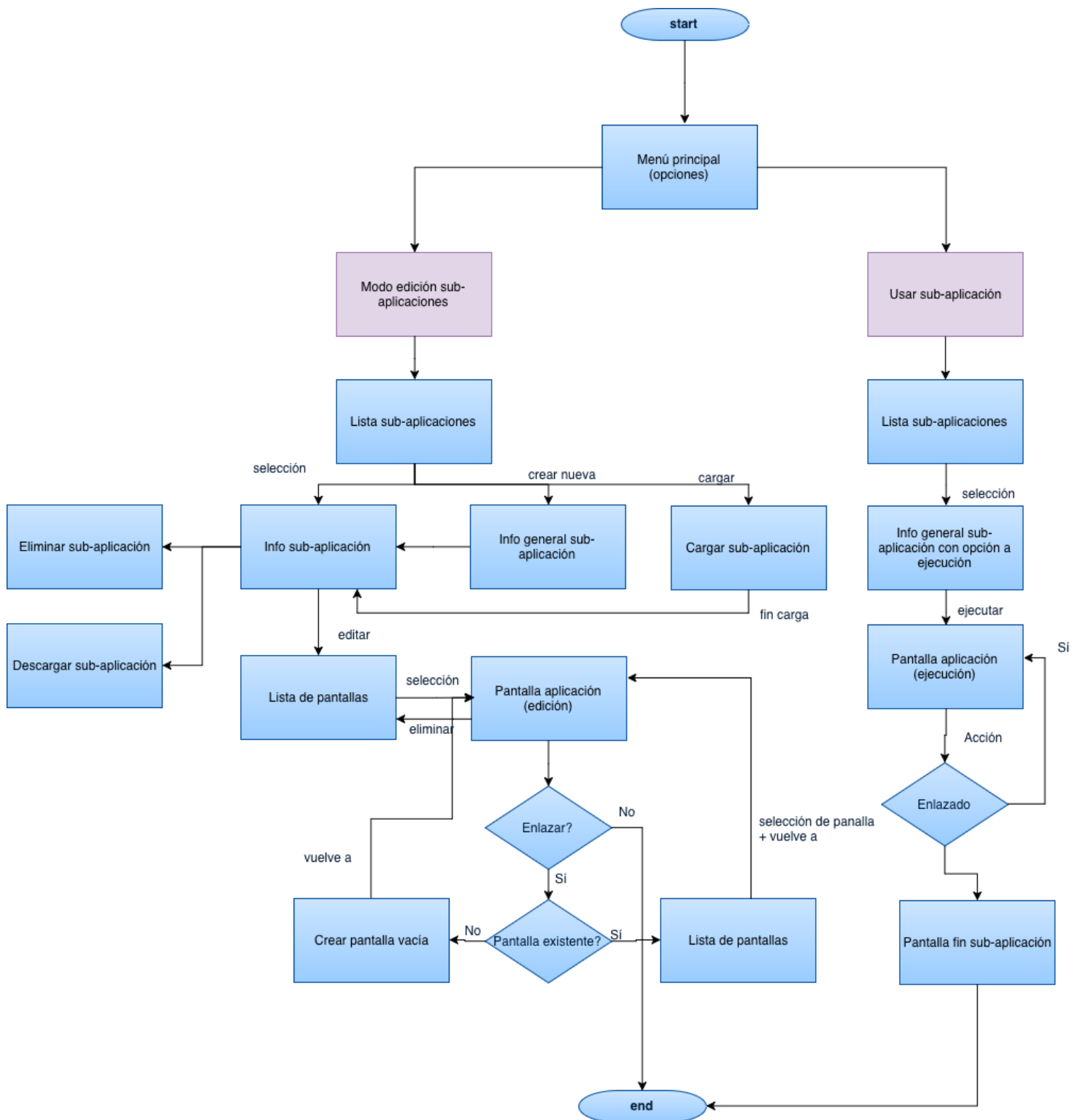


Figura 14: Diagrama de flujo de la aplicación

4. Prototipo

A partir de los flujos de interacción detallados anteriormente, se ha generado el sitemap de la aplicación y su prototipo.

4.2 Sitemap

Gracias al siguiente sitemap se puede saber el número de pantallas que tendrá la aplicación y cómo se relacionan entre ellas.

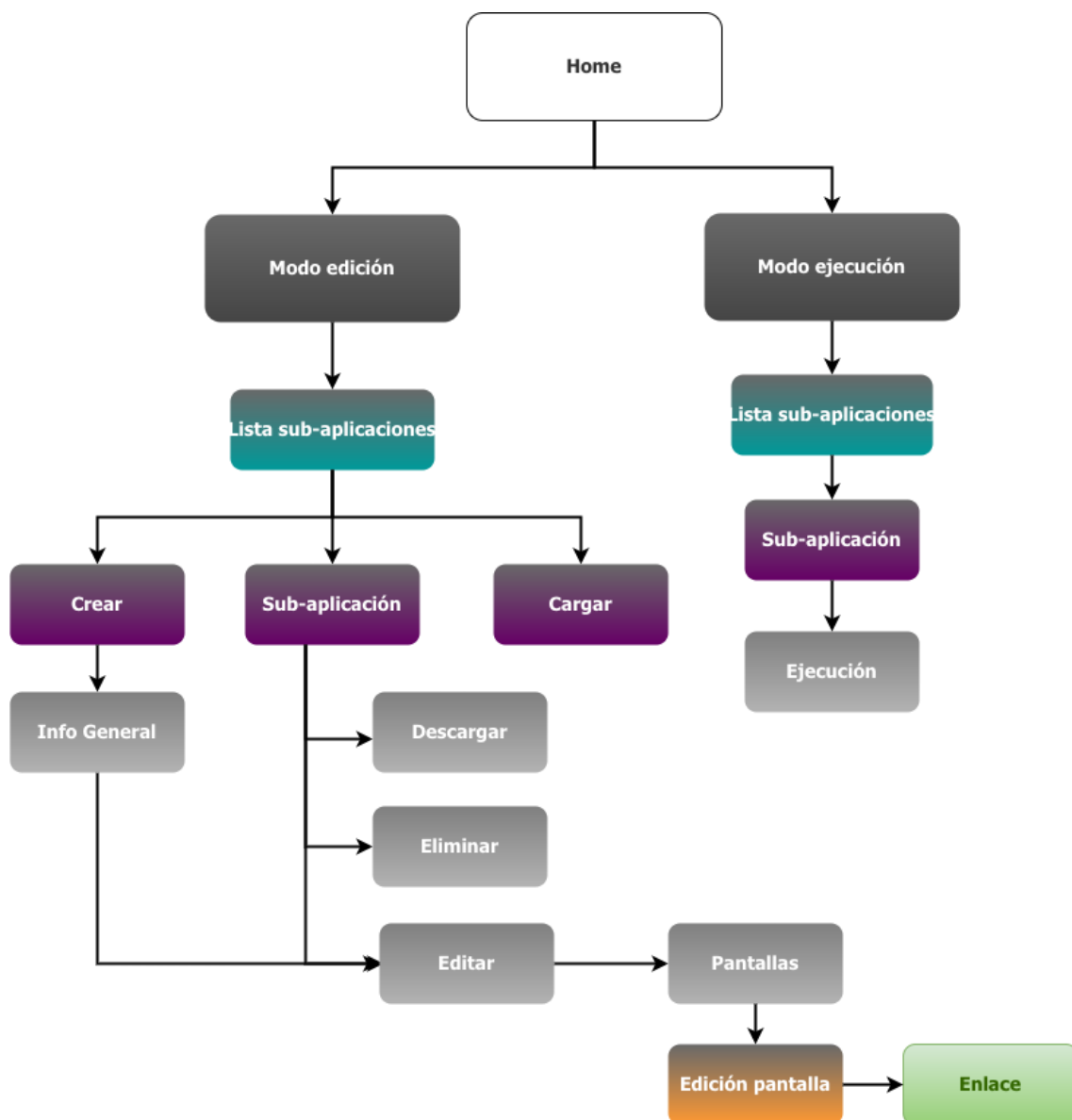


Figura 15: Sitemap

4.3 Prototipo

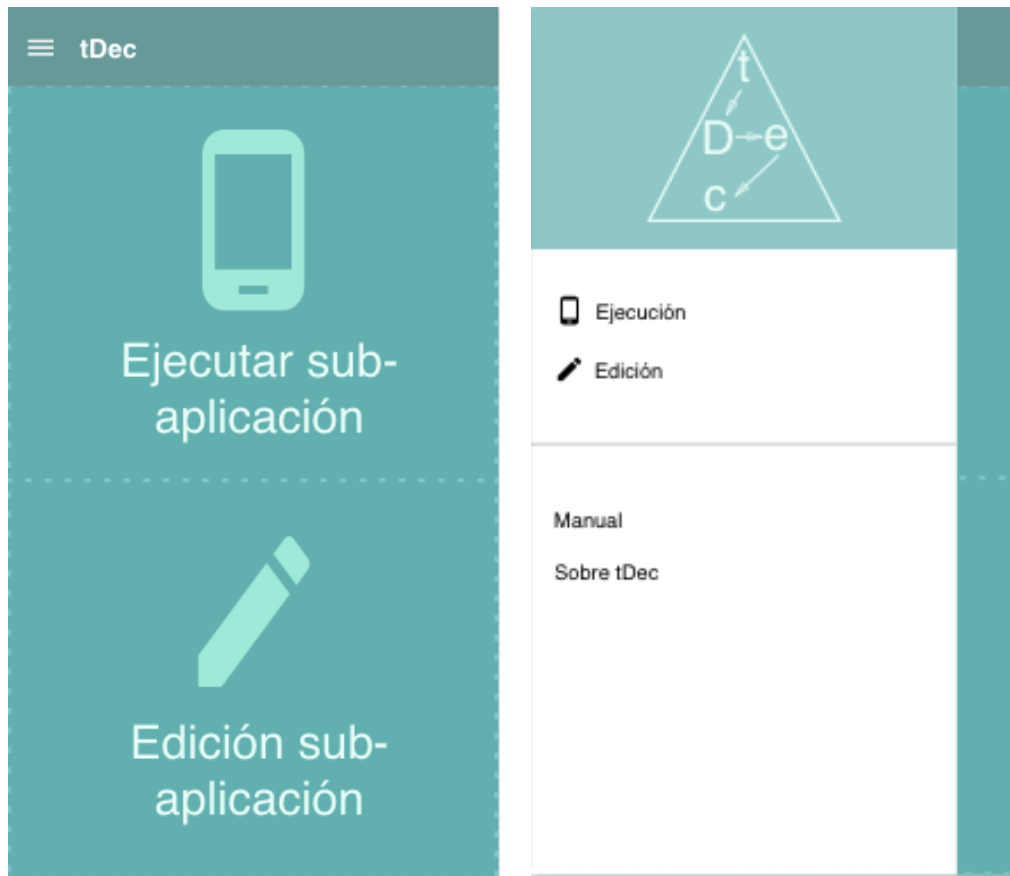
A continuación, se muestra el prototipo de las pantallas principales de la aplicación.

Pantalla principal

La pantalla principal consta de dos secciones:

- Ejecutar sub-aplicación
- Editar sub-aplicación

Cuando el usuario presione sobre una de las áreas, será re direccionado a la pantalla correspondiente.



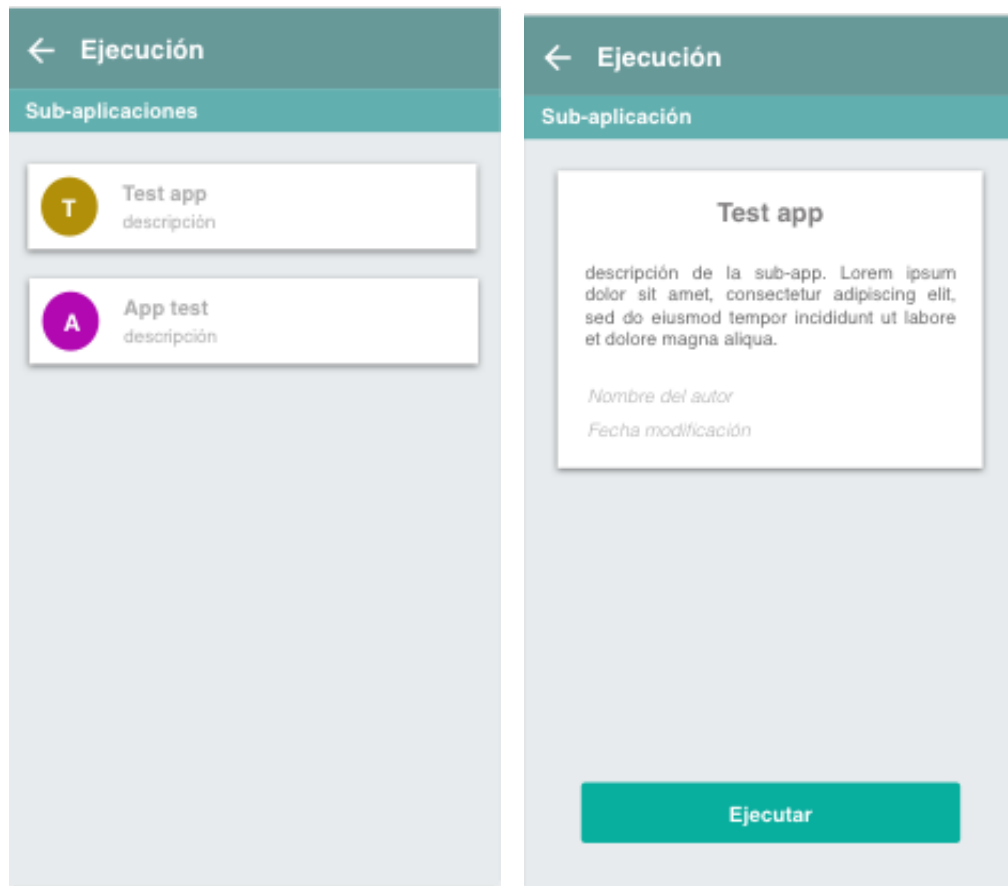
La pantalla consta de un menú principal, el cual dispone de acceso a las pantallas de la aplicación:

- Ejecución de sub-aplicación
- Edición de sub-aplicación
- Manual de uso de tDec
- Información de tDec

Ejecución de sub-aplicación

Al acceder a la pantalla de ejecución de sub-aplicación, al usuario le aparecerán todas las sub-aplicaciones de las que dispone. Al seleccionar una de ellas, el usuario es re direccionado a una pantalla que contiene la información principal de la sub-aplicación

(nombre, descripción, autor y fecha de modificación), y un botón que le permitirá ejecutarla.



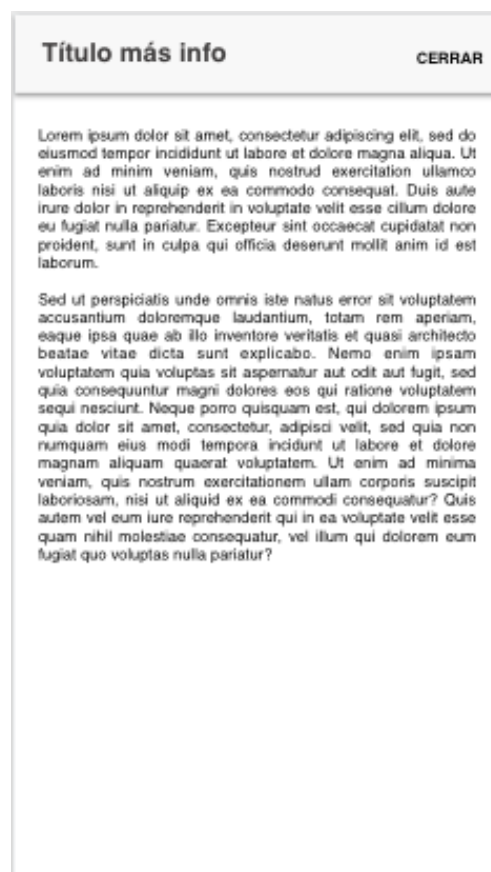
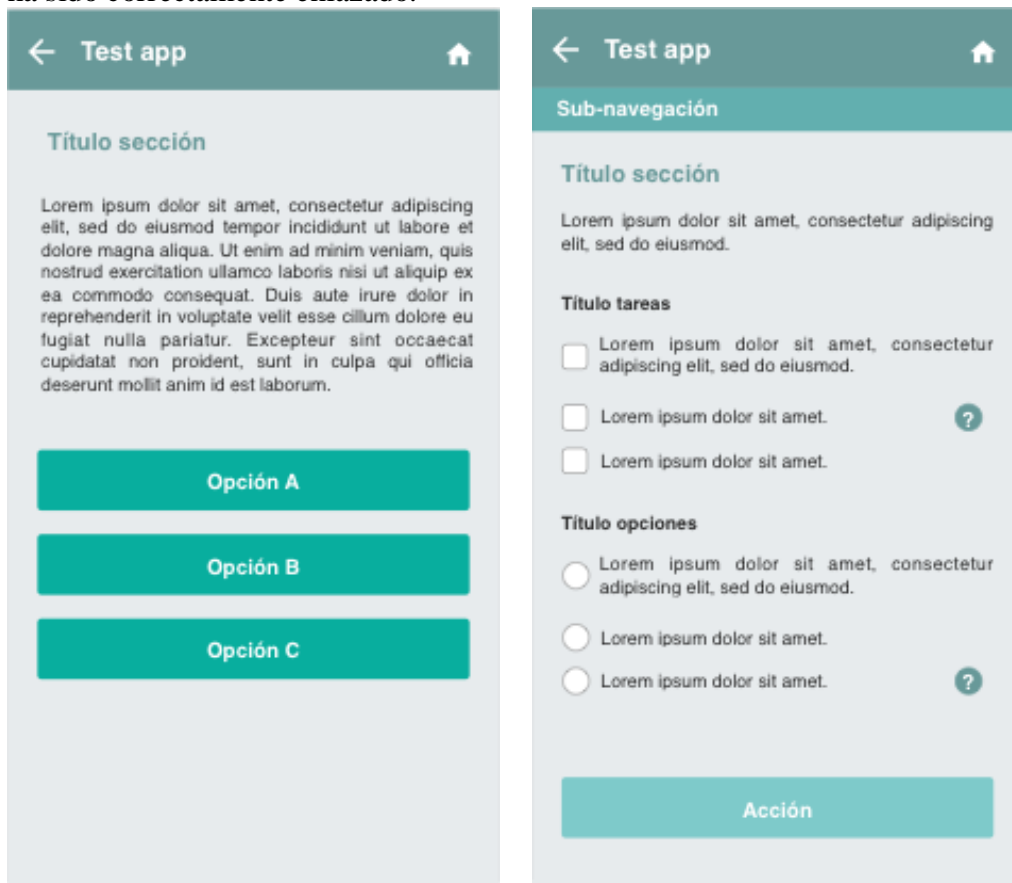
Cuando el usuario presiona el botón ejecutar, éste es re direccionado a la primera pantalla de la sub-aplicación, la cual es autogenerada a partir de la información que se proveyó al crearla.

Esta pantalla puede tomar varios formatos, pero se basa únicamente en el uso de haber provisto o no de la siguiente información:

- Título de sub-navegación
- Título de la sección
- Texto (descripción, pregunta, explicación...)
- Sección de tareas
 - Título
 - Tareas (mostradas en forma de checkboxes)
- Sección de opciones
 - Título
 - Opciones posibles (mostrado en forma de radio buttons)
- Respuestas o acciones posibles (mostrado en forma de botones)

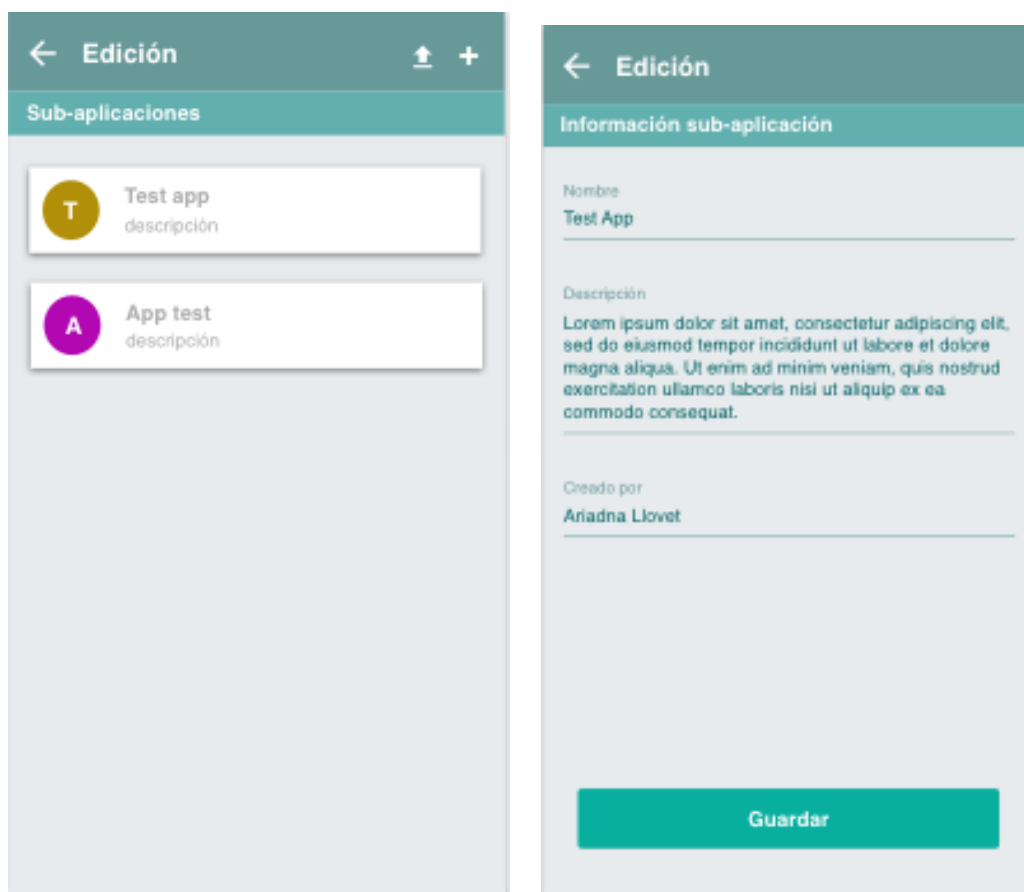
*Cada uno de los textos puede estar acompañado de un icono en forma de interrogante, el cual permitirá acceder a más información.

Al presionar sobre un botón, el usuario será re direccionado a la siguiente pantalla si éste ha sido correctamente enlazado.



Edición de sub-aplicación

Al acceder a la pantalla de edición de sub-aplicaciones, se le mostrarán al usuario todas las sub-aplicaciones de las que dispone.



En esta pantalla, el usuario tiene las opciones de:

- Crear una nueva sub-aplicación (símbolo +)
- Cargar sub-aplicación (símbolo de carga)
- Editar sub-aplicación (al presionar sobre una sub-aplicación)

Si el usuario decide crear una nueva sub-aplicación, aparecerá una nueva pantalla con un formulario a rellenar. Una vez el usuario ha indicado el nombre, descripción, y autor, si se presiona el botón “Guardar”, la aplicación será creada y el usuario re direccionado a la pantalla de información de la sub-aplicación.

En la pantalla de información, al usuario se le presentan los datos principales de la sub-aplicación: nombre, descripción, autor, y fecha de modificación. Así como se le presentan también las diferentes acciones que puede realizar:

- Editar información
- Editar sub-aplicación
- Descargar sub-aplicación

- Eliminar sub-aplicación

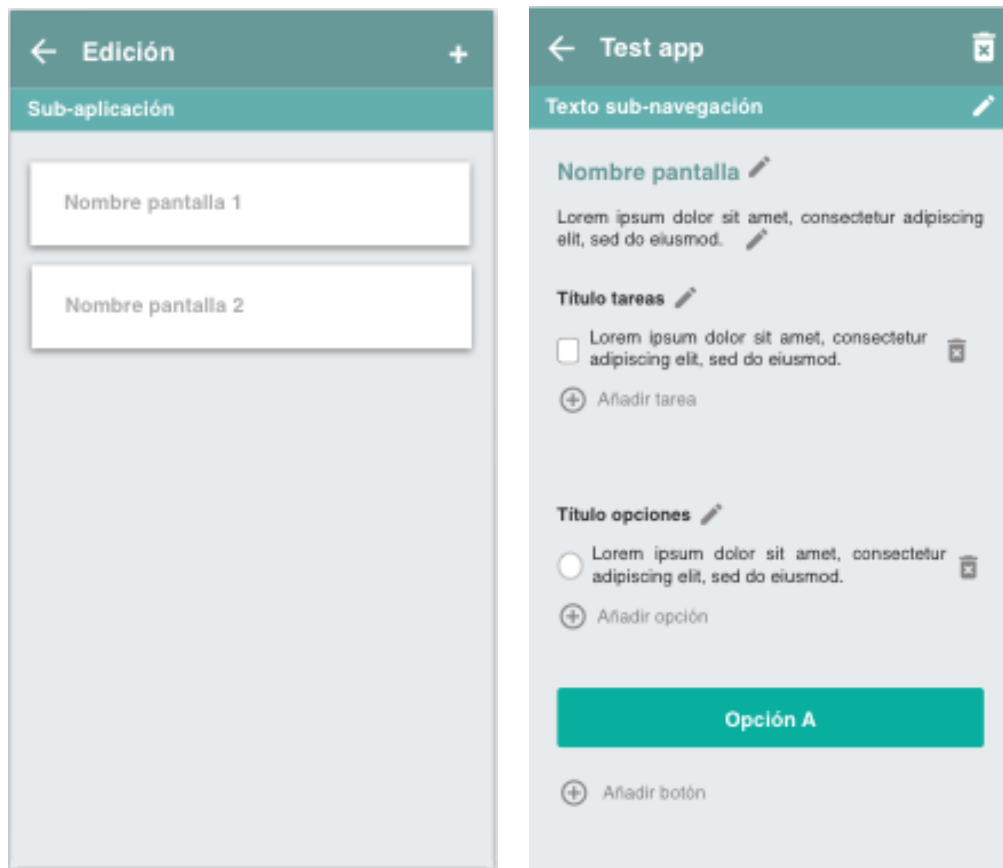


Para editar la información, el usuario deberá oprimir el icono del lápiz, y será re direccionado a la misma pantalla de creación de sub-aplicación.

Si el usuario decide que quiere descargar la sub-aplicación, tDec le generará un archivo que podrá ser utilizado para cargarlo en cualquier dispositivo con la aplicación tDec.

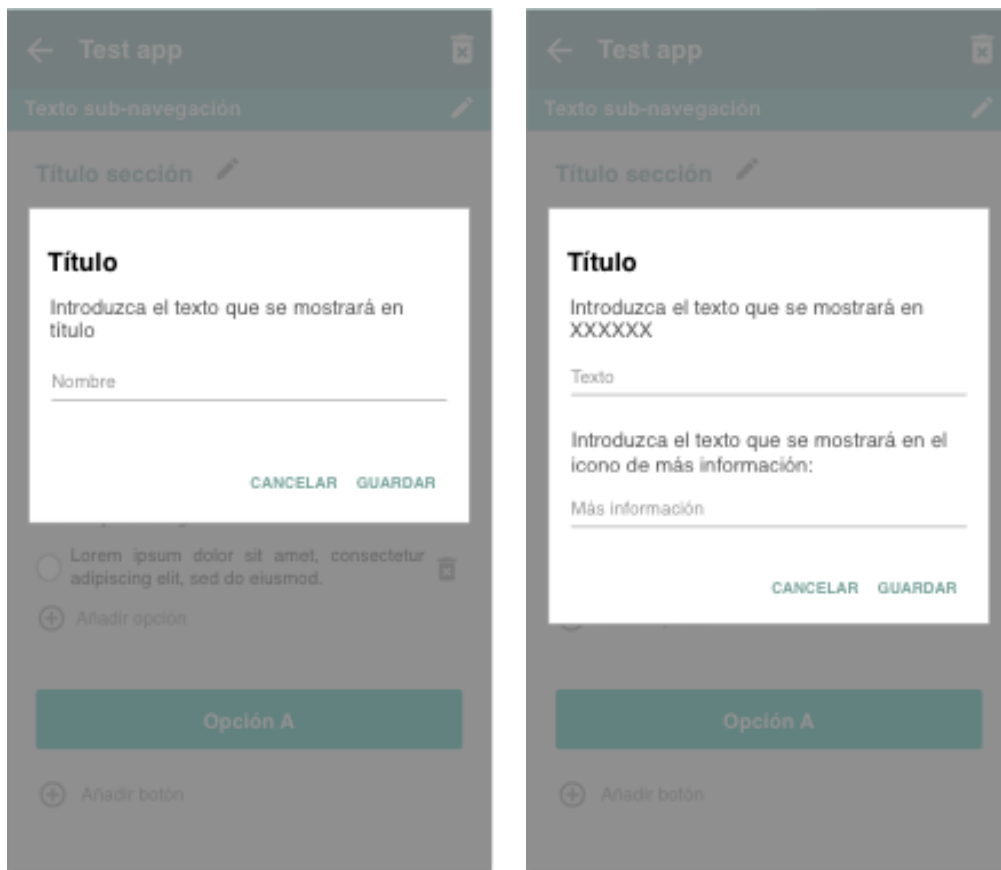
Si el usuario indica que desea eliminar la sub-aplicación, le aparecerá un mensaje informándole de que será borrado permanentemente del sistema. El usuario podrá decidir si desea proseguir con la acción o cancelarla.

Finalmente, si el usuario indica que quiere editar la sub-aplicación, será re direccionado a una nueva pantalla que contendrá una lista de los nombres de todas las pantallas existentes en la sub-aplicación. En esta pantalla el usuario podrá decidir entre crear una nueva pantalla (símbolo +) o editar una existente (seleccionando la pantalla en cuestión). En ambos casos, se re direccionará al usuario a la misma pantalla (edición pantalla). La única diferencia será que si el usuario ha decidido editar una pantalla, se presentarán en los campos la información disponible de la pantalla, mientras que si ha decidido crearla estarán con información descriptiva del campo.

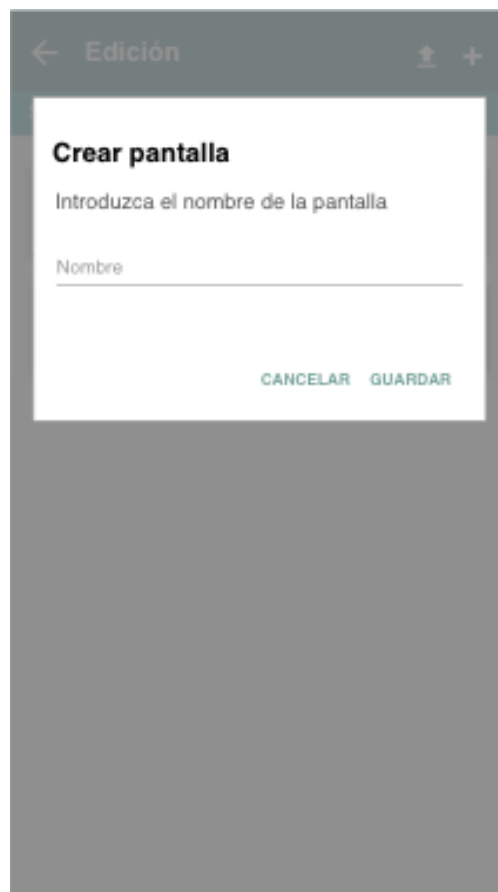
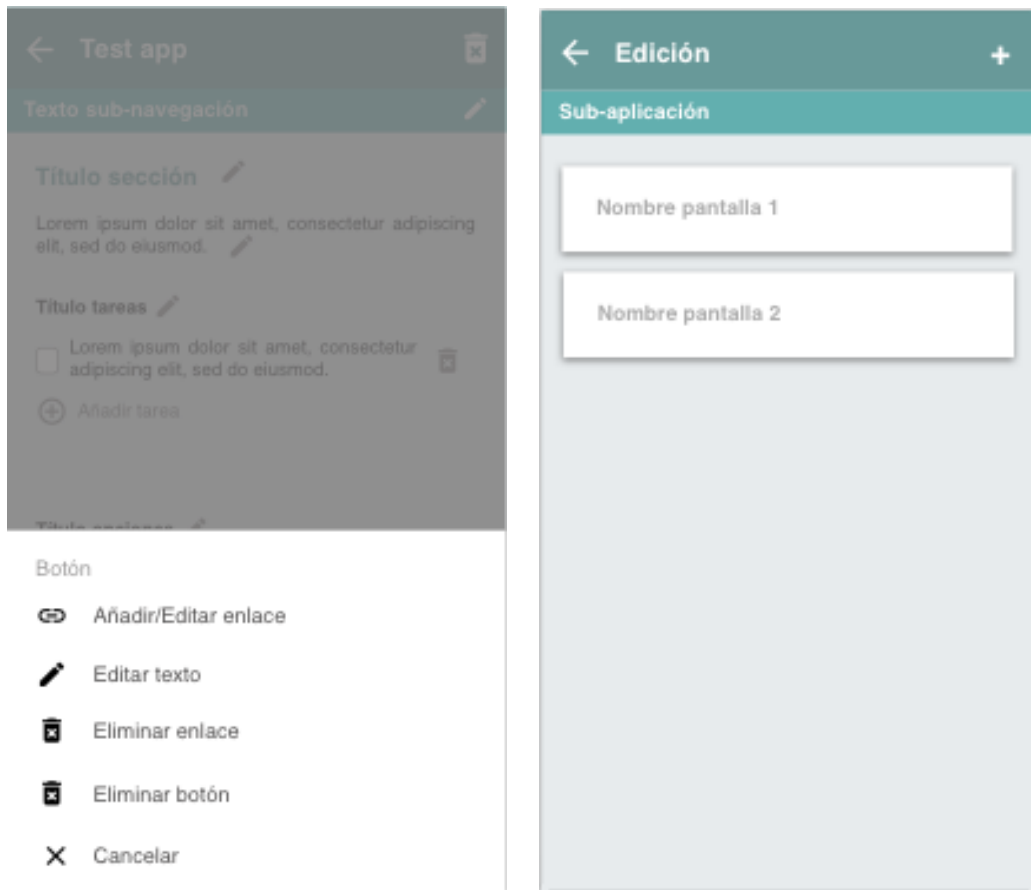


La pantalla de edición de pantalla, es la que permitirá al usuario decidir qué mostrar.

- Sub-navegación: mediante la imagen del lápiz, el usuario podrá editar el texto a mostrar o borrar el texto existente. En este último caso, la barra de sub-navegación no se mostrará al ejecutarse. Para ello, aparecerá un dialog con un input para introducir o borrar el texto.
- Título de la sección o página: Este título será el utilizado a la hora de mostrar la lista de pantallas de la sub-aplicación. Al presionar el lápiz, aparecerá un dialog para indicar el texto a mostrar.
- Texto: Texto utilizado como descripción de la pantalla, pregunta a realizar, explicación... Al presionar el lápiz, aparecerá un dialog más complejo, el cual permitirá escribir el texto a mostrar y opcionalmente, texto adicional que sería mostrado a través de un icono con el signo de interrogación.
- Sección de tareas
 - Título: título para presentar las tareas. Al presionar el lápiz, aparecerá el dialog complejo para indicar el texto a mostrar y, opcionalmente, la información adicional.
 - Tareas: checkboxes de tareas. Éstas pueden:
 - Eliminarsse: presionando el icono de la papelera que se encuentra en su lateral derecho.
 - Añadir nueva: presionando “Añadir tarea” aparecerá el dialog complejo para indicar el texto a mostrar y, opcionalmente, la información adicional.



- Sección de opciones
 - Título: título para presentar las opciones. Al presionar el lápiz, aparecerá el dialog complejo para indicar el texto a mostrar y, opcionalmente, la información adicional.
 - Tareas: radio buttons de opciones. Éstos pueden:
 - Eliminars: presionando el icono de la papelera que se encuentra en su lateral derecho.
 - Añadir nueva: presionando “Añadir opción” aparecerá el dialog complejo para indicar el texto a mostrar y, opcionalmente, la información adicional.
- Respuestas o acciones posibles: cada una de éstas, será presentado en forma de botón:
 - Añadir: presionando “Añadir botón” aparecerá un dialog para indicar el texto a mostrar.
 - Al presionar el botón, aparecerá un menú en el cual se permiten las opciones de:
 - Añadir/Editar enlace: se le presentará al usuario la lista de pantallas.
 - Editar texto: permitirá al usuario cambiar el texto del botón.
 - Eliminar enlace: permitirá eliminar el enlace del botón a la pantalla previamente seleccionada.
 - Eliminar botón: borrará el botón de la lista de botones disponibles.
 - Cancelar: cerrará el menú de opciones.



Para realizar el enlace, al usuario se le presentará la lista de pantallas existentes y la opción de crear una nueva pantalla. Si selecciona una pantalla, el enlace se creará directamente, mientras que si selecciona crear pantalla, el usuario deberá introducir el nombre de la pantalla, ésta se creará en blanco y el botón quedará enlazado.

5. Casos de uso

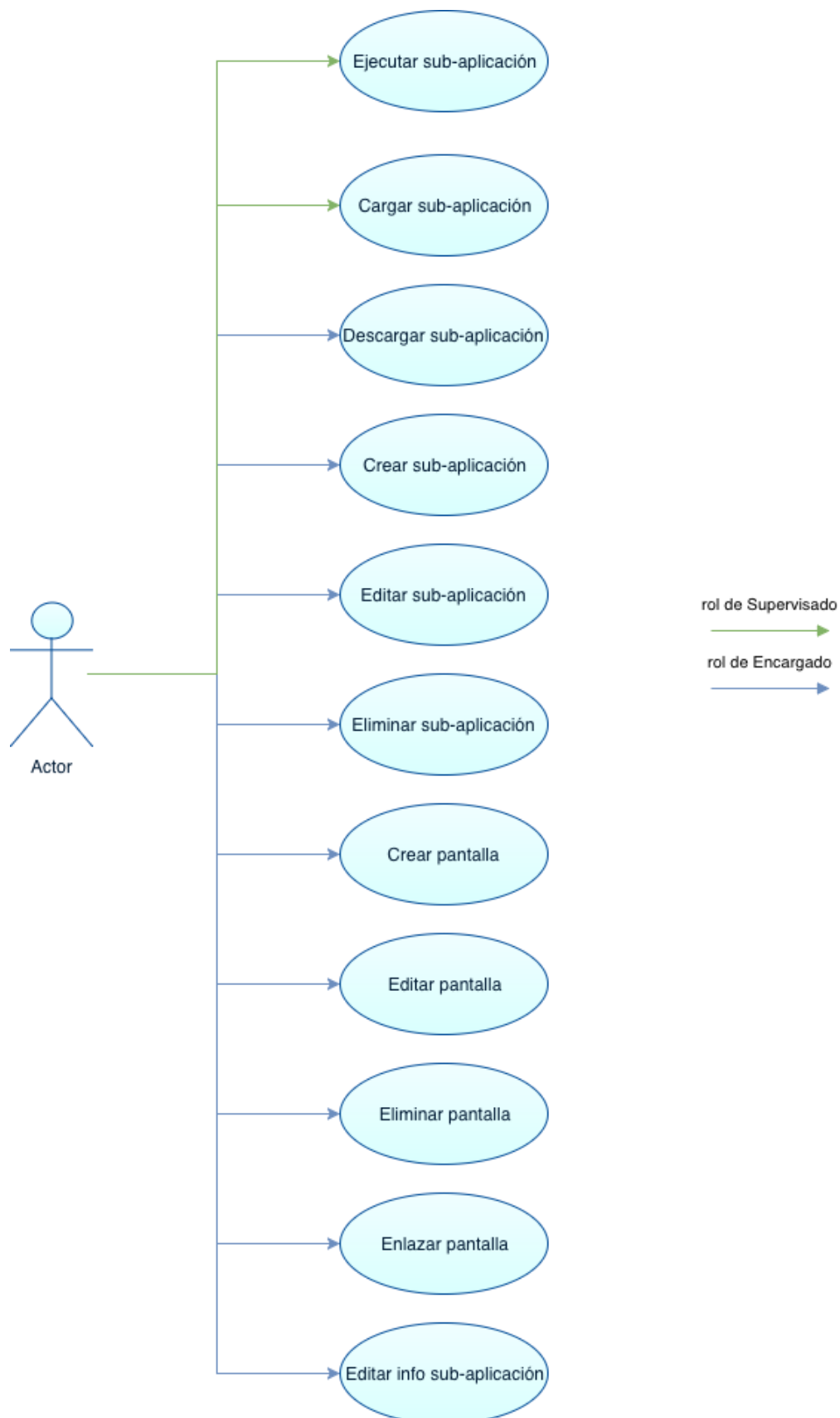


Figura 16: Diagrama de casos de uso

Tal y como se ha comentado anteriormente, el actor de la aplicación puede ser un único usuario con dos roles: el rol de supervisado (ejecución de sub-aplicaciones), y el rol de encargado (creación y edición de sub-aplicaciones).

A continuación se detallan los casos de uso de la aplicación.

Crear sub-aplicación

Caso de uso: Crear sub-aplicación
<p>Descripción: El usuario crea una nueva sub-aplicación de toma de decisiones.</p> <p>Actor principal: Encargado.</p> <p>Precondición: -</p> <p>Flujo principal de ejecución: Flujo de interacción de crear sub-aplicación</p> <p>Postcondición: El usuario dispone de una sub-aplicación que podrá ser ejecutada desde el apartado de ejecución o editada desde del apartado de edición.</p>

Editar sub-aplicación

Caso de uso: Editar sub-aplicación
<p>Descripción: El usuario edita el contenido de la sub-aplicación.</p> <p>Actor principal: Encargado.</p> <p>Precondición: Sub-aplicación previamente creada.</p> <p>Flujo principal de ejecución: Flujo de interacción de editar sub-aplicación</p> <p>Postcondición: El usuario dispone de la sub-aplicación editada.</p>

Editar información de la sub-aplicación

Caso de uso: Editar información de la sub-aplicación
<p>Descripción: El usuario edita la información básica de la sub-aplicación.</p> <p>Actor principal: Encargado.</p> <p>Precondición: Sub-aplicación previamente creada.</p> <p>Flujo principal de ejecución: Flujo de interacción de editar información de la sub-aplicación</p> <p>Postcondición: El usuario dispone de la sub-aplicación con la información editada.</p>

Ejecutar sub-aplicación

Caso de uso: Ejecutar sub-aplicación
<p>Descripción: El usuario desea ejecutar una sub-aplicación en concreto.</p> <p>Actor principal: Supervisado.</p> <p>Precondición: Sub-aplicación previamente creada.</p> <p>Flujo principal de ejecución: Flujo de interacción de ejecutar sub-aplicación</p> <p>Postcondición: El usuario ha ejecutado con éxito la sub-aplicación indicada. Las pantallas se muestran</p>

según las respuestas que indica el usuario.

Eliminar sub-aplicación

Caso de uso: Eliminar sub-aplicación

Descripción: El usuario desea eliminar una sub-aplicación.

Actor principal: Encargado.

Precondición: Sub-aplicación previamente creada.

Flujo principal de ejecución:

[Flujo de interacción de eliminar sub-aplicación](#)

Postcondición:

La sub-aplicación ha sido borrada completamente del sistema.

Descargar sub-aplicación

Caso de uso: Descargar sub-aplicación

Descripción: El usuario descarga una sub-aplicación en concreto para poder ser cargada en cualquier dispositivo con la aplicación tDec.

Actor principal: Encargado.

Precondición: Sub-aplicación previamente creada.

Flujo principal de ejecución:

[Flujo de interacción de descargar sub-aplicación](#)

Postcondición:

Al usuario se le genera un fichero con la información de la sub-aplicación para ser posteriormente importada.

Cargar sub-aplicación

Caso de uso: Cargar sub-aplicación

Descripción: El usuario quiere importar un fichero con una sub-aplicación en él.

Actor principal: Supervisado.

Precondición: Sub-aplicación previamente creada y descargada.

Flujo principal de ejecución:

[Flujo de interacción de cargar sub-aplicación](#)

Postcondición:

Se guarda en el sistema una nueva sub-aplicación basada en el contenido del fichero importado.

Crear pantalla

Caso de uso: Crear pantalla

Descripción: El usuario desea crear una nueva pantalla para una sub-aplicación.

Actor principal: Encargado.

Precondición: Sub-aplicación previamente creada.

Flujo principal de ejecución:

[Flujo de interacción de crear pantalla](#)

Postcondición:

Se guarda en la sub-aplicación la nueva pantalla.

Editar pantalla

Caso de uso: Editar pantalla

Descripción: El usuario desea editar una pantalla de una sub-aplicación.

Actor principal: Encargado.

Precondición: Sub-aplicación creada con pantallas previamente creadas.

Flujo principal de ejecución:

[Flujo de interacción de editar pantalla](#)

Postcondición:

Se guarda en la sub-aplicación la pantalla modificada.

Eliminar pantalla

Caso de uso: Eliminar pantalla

Descripción: El usuario desea eliminar una pantalla de una sub-aplicación.

Actor principal: Encargado.

Precondición: Sub-aplicación creada con pantallas previamente creadas.

Flujo principal de ejecución:

[Flujo de interacción de eliminar pantalla](#)

Postcondición:

El sistema ha eliminado completamente de la sub-aplicación la pantalla indicada.

Crear enlace

Caso de uso: Crear enlace

Descripción: El usuario desea crear el enlace de un botón de una pantalla a otra pantalla.

Actor principal: Encargado.

Precondición: Sub-aplicación creada con pantallas previamente creadas.

Flujo principal de ejecución:

[Flujo de interacción de crear enlace](#)

Postcondición:

El sistema ha generado un enlace en la sub-aplicación entre el botón en cuestión y la pantalla indicada.

Editar enlace

Caso de uso: Editar enlace

Descripción: El usuario desea editar el enlace de un botón de una pantalla a otra pantalla.

Actor principal: Encargado.

Precondición: Sub-aplicación creada con al menos una pantallas creada con el botón enlazado.

Flujo principal de ejecución:

[Flujo de interacción de editar enlace](#)

Postcondición:

El sistema ha modificado el enlace en la sub-aplicación entre el botón en cuestión y la pantalla indicada.

Eliminar enlace

Caso de uso: Eliminar enlace

Descripción: El usuario desea eliminar el enlace de un botón de una pantalla a otra pantalla.

Actor principal: Encargado.

Precondición: Sub-aplicación creada con al menos una pantallas creada con el botón enlazado.

Flujo principal de ejecución:

[Flujo de interacción de eliminar enlace](#)

Postcondición:

El sistema elimina el enlace del botón indicado de la pantalla de la sub-aplicación.

5. Diseño de la arquitectura

En este apartado se define la arquitectura del sistema, identificando cómo se almacenarán los datos, y las clases y objetos que se utilizarán para gestionar los diferentes procesos.

Datos

En este trabajo, para almacenar los datos se utilizarán ficheros en formato JSON. Cada uno de estos ficheros representará una sub-aplicación de tDec.

El hecho de almacenar los datos en ficheros JSON es debido al tiempo limitado que se dispone para la implementación de la aplicación. Una futura actualización incluiría el almacenamiento de los datos en una base de datos sqlite.

Formato del fichero JSON:

```
{
  "info": {
    "name": "Nombre app",
    "descr": "descripcion de la app",
    "author": "nombre del autor",
    "modDate": "DATE"
  },
  "screens": {
    "1": {
      "nav_title": "Tratamiento",
      "title": "ECG Inicial",
      "description": {
        "text": "Texto principal de la pantalla",
        "moreinfo": "Info adicional"
      }
    },
  },
}
```

```

    "description2": {
      "text": "Texto previo checkboxes",
      "moreinfo": "Info adicional"
    },
    "checkboxes": [{
      "text": "Texto checkbox 1",
      "moreinfo": "Info adicional"
    }],
    "description3": {
      "text": "Texto previo radiobuttons",
      "moreinfo": "Info adicional"
    },
    "radiobuttons": [{
      "text": "Texto radiobutton 1",
      "moreinfo": "Info adicional"
    }],
    "buttons": [{
      "text": "Texto boton",
      "linkTo": "2"
    }]
  },
  "2": {
    "title": "Pantalla 2",
    "description": {
      "text": "Esta es la pantalla 2"
    }
  }
}

```

Una sub-aplicación está formada de dos estructuras: “info” y “screens”.

El campo “info” contiene la información básica de la aplicación:

- name: Nombre
- descr: Descripción
- author: Autor
- modDate: Fecha de la última modificación

El campo “screen” contiene un lista con todas las pantallas de la sub-aplicación. Cada una de estas pantallas es identificada por un número y puede estar compuesta por:

- nav_title: texto que se mostrará en la sub-barra de navegación.

- title: título de la pantalla.
- description: texto principal de la pantalla (descripción, pregunta, pasos a seguir, etc). Campo compuesto por:
 - text: texto a mostrar
 - moreinfo: información adicional que será mostrado al clicar un icono.
- description2: Texto normalmente utilizado para acompañar las tareas.
 - text: texto a mostrar.
 - moreinfo: información adicional que será mostrado al clicar un icono.
- checkboxes: lista de tareas. Una tarea está compuesta de:
 - text: texto a mostrar.
 - moreinfo: información adicional que será mostrado al clicar un icono.
- Description3: Texto normalmente utilizado para acompañar las opciones.
 - text: texto a mostrar.
 - moreinfo: información adicional que será mostrado al clicar un icono.
- radiobuttons: lista de opciones. Una opción está compuesta de:
 - text: texto a mostrar.
 - moreinfo: información adicional que será mostrado al clicar un icono.
- buttons: Lista de respuestas o acciones. Compuesto de:
 - text: texto a mostrar.
 - linkTo: identificador de la pantalla con la que se enlaza.

El diseño UML de la base de datos sería de la siguiente manera:

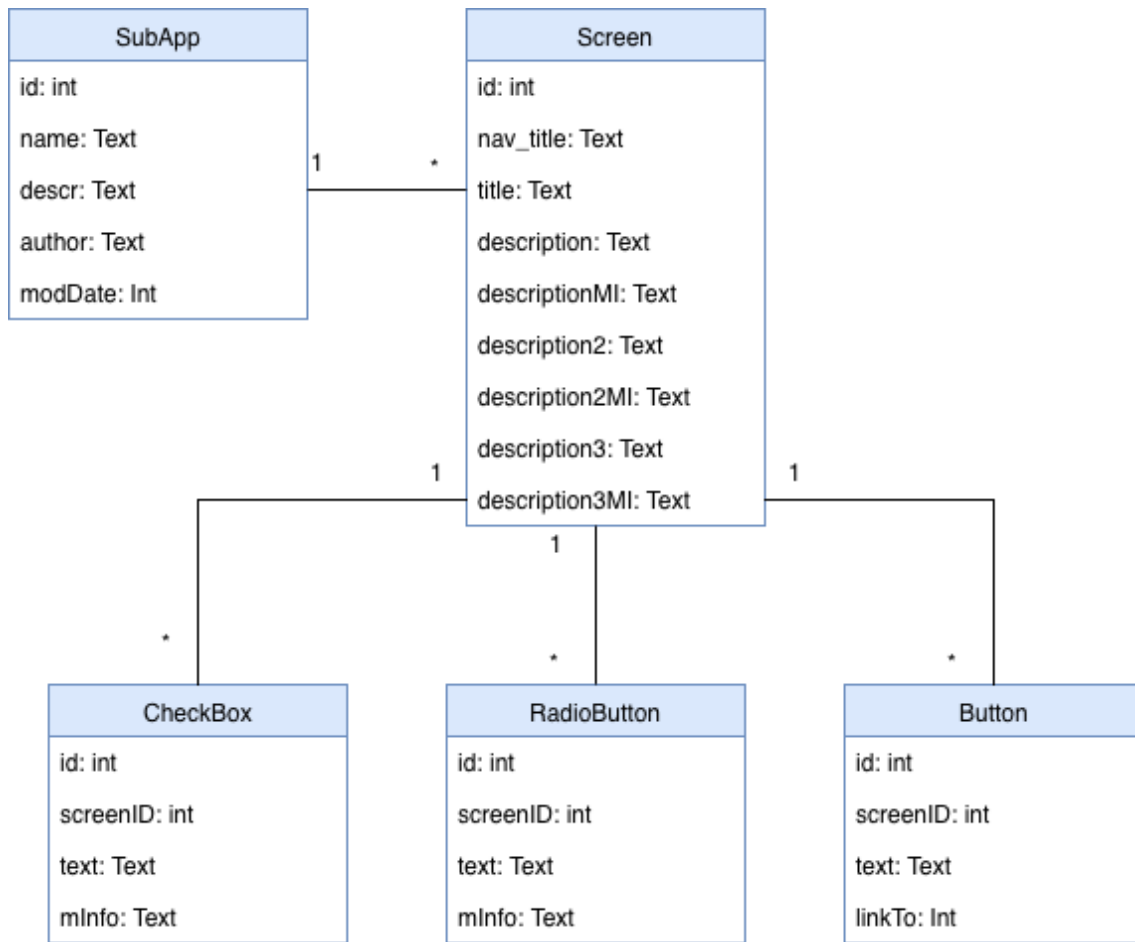


Figura 17: UML BD

Entidades y clases

A continuación, se muestra el diagrama UML de las clases principales de la aplicación. En él se muestran las clases del modelo (verde), los controladores principales (amarillo) que incluyen el controlador del modelo, el servicio proveedor de los datos, y el controlador de importación y exportación, y finalmente una caja (naranja) que emula las clases referentes a los controladores de las vistas.

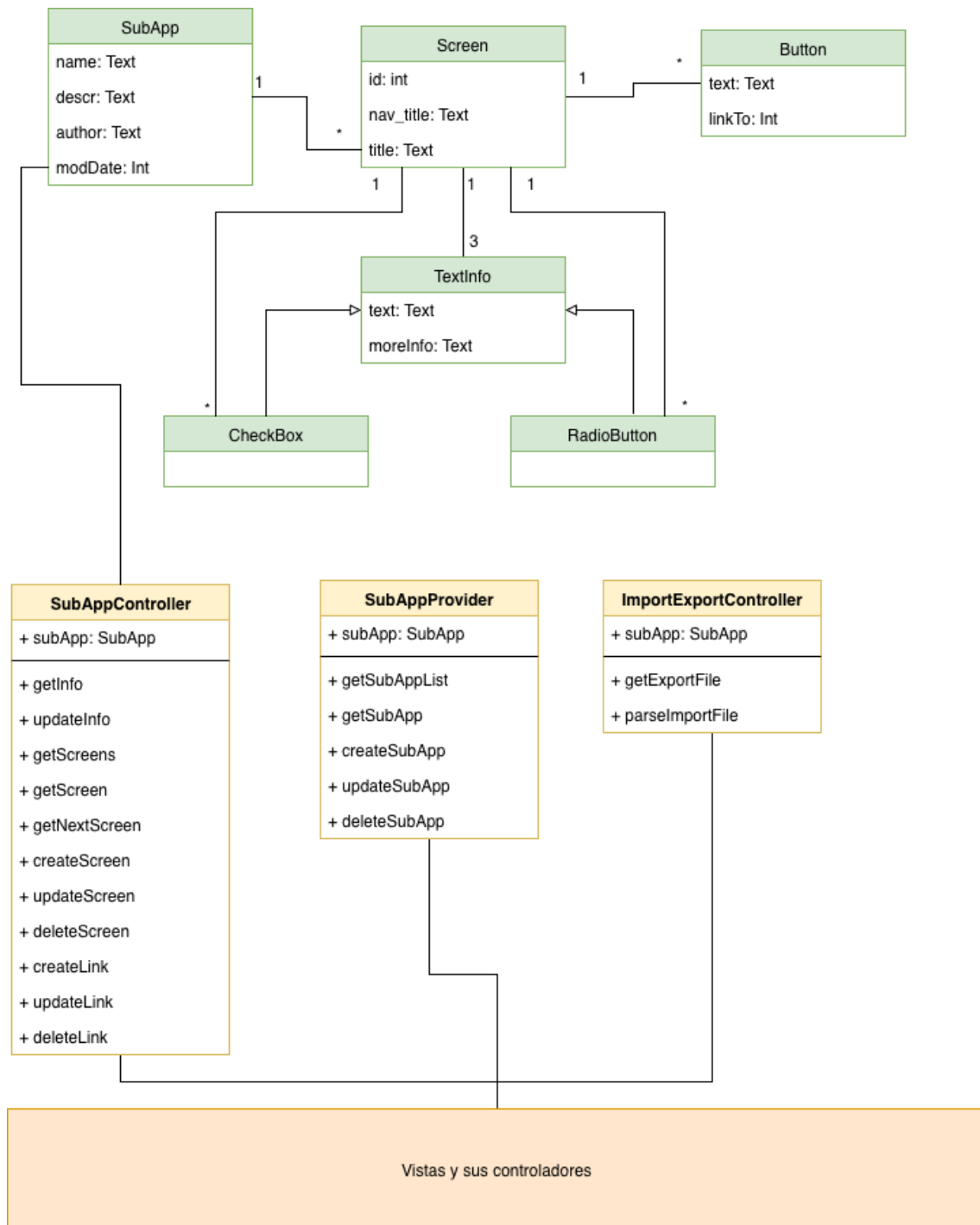


Figura 18: Diagrama de clases

Arquitectura del sistema

La aplicación seguirá el patrón de arquitectura de Modelo Vista Controlador (MVC), el cual separa los datos y la lógica de la interfaz de usuario.

- **Modelo:** Es la representación de la información sobre la que la aplicación opera.

- Controlador: Responde a los eventos (acciones de usuario) y este realiza las peticiones al modelo. También solicita cambios a la vista con tal de que se muestren.
- Vista: Presenta el modelo en forma de interfaz de usuario.

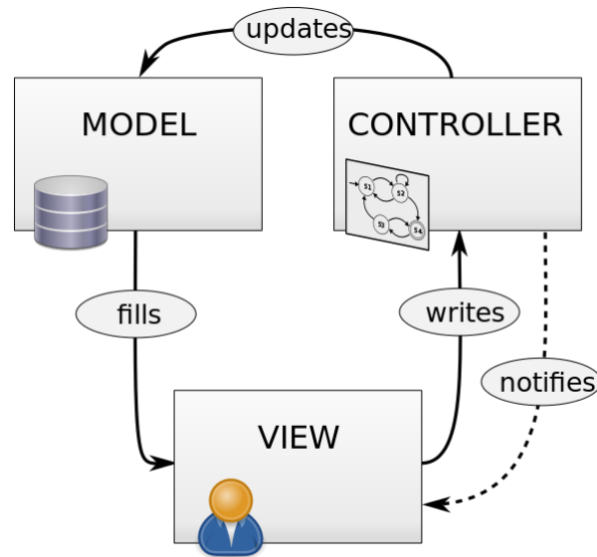


Figura 19: Arquitectura MVC

Ionic, tal y como se ha mencionado anteriormente, corre sobre Angular, el cual está construido siguiendo el patrón de arquitectura de MVC.

Así pues, Ionic dispone de Vistas (Templates), sus controladores, y los servicios o proveedores de datos. Adicionalmente incorpora un configurador de la aplicación y sus directivas.

Views (Templates) <pre> <div id="forecast" class="weather-box"> <h4 class="title">Forecast</h4> <ion-scroll direction="x" id="forecast-scroll"> <div id="hourly-forecast"> <div class="hourly-hour" ng-repeat="hour in current <div class="time" ng-bind="hour.time * 1000 date <weather-icon icon="hour.icon"></weather-icon> <div class="temp">{{hour.temperature templint}} </div> </div> </ion-scroll> <div class="hour" ng-repeat="hour in currentHourlyData </div> </pre>	Controllers <pre> .controller('SettingsCtrl', function(\$scope, Settings) { \$scope.settings = Settings.getSettings(); // Watch deeply for settings changes, and save them // if necessary \$scope.\$watch('settings', function(v) { Settings.save(v); }, true); \$scope.closeSettings = function() { \$scope.model.hide(); } } </pre>	Data (Services/Factories) <pre> .factory('Flickr', function(\$q, \$resource, FLICKR_API_KEY) { var baseUrl = 'http://api.flickr.com/services/rest/'; var flickrSearch = \$resource(baseUrl, { method: 'flickr.groups.pools.getPhotos', group_id: '1463451@N25', safe_search: 1, jsoncallback: 'JSON_CALLBACK', api_key: FLICKR_API_KEY, format: 'json' }); } </pre>
	App Configuration <pre> .config(function(\$stateProvider, \$urlRouterProvider) { \$stateProvider .state('app', { url: "/app", abstract: true, templateUrl: "templates/menu.html", controller: 'AppCtrl' }) .state('app.search', { url: "/search" }) } </pre>	Directives <pre> .directive('weathericon', function(WEATHER_ICONS) { return { restrict: 'E', replace: true, scope: { icon: "=" }, template: '<i class="icon" ng-class="weathericon"></i>', link: function(\$scope) { \$scope.\$watch('icon', function(v) { if(v) return; }); } }; } </pre>

Figura 20: Ionic MVC

5. Desarrollo

En este apartado se explican las decisiones de desarrollo que se han tomado a la hora de programar la aplicación, así como las complicaciones encontradas.

Tecnologías de desarrollo

Tal y como se ha comentado en apartados anteriores, el desarrollo se ha realizado con el framework de Ionic, concretamente se ha utilizado la versión 3.19.0, con las plataformas de android y iOS añadidas mediante cordova.

El IDE utilizado para programar ha sido WebStorm, concretamente la versión 2017.1.

Durante el desarrollo de la aplicación el código de la aplicación se ha gestionado mediante GIT, sistema de control de versiones.

Estructura del proyecto

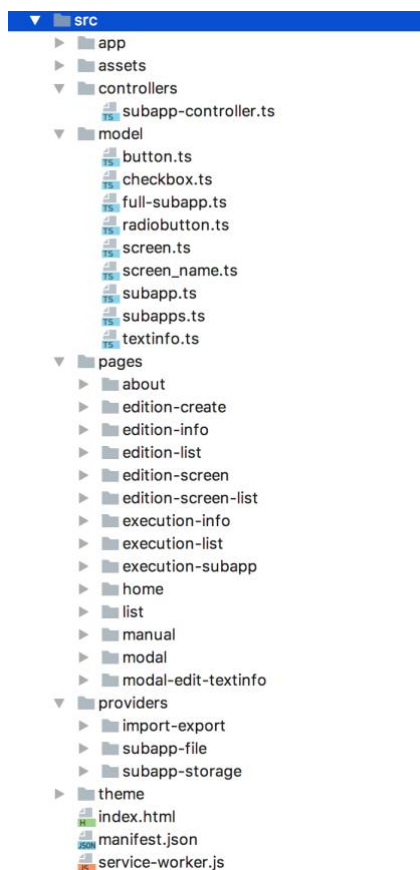


Figura 21: Estructura del proyecto

La estructura del proyecto sigue la línea de Ionic/AngularJS donde cada pantalla es un componente formado por su vista y controlador. Al disponer de una clara separación entre ellos, se puede cambiar en cualquier momento la parte visual de la pantalla sin necesidad de modificar el código que la controla. Estos componentes se han creado dentro de la carpeta pages.

El modelo de representación de los datos ha sido creado dentro de la carpeta models, donde se han definido todas las clases necesarias de la aplicación. El controlador de éstos ha sido creado en la carpeta controllers.

Los proveedores de datos y archivos se han creado dentro de la carpeta providers.

Finalmente, las imágenes de la aplicación han sido almacenadas en la carpeta assets.

Base de datos

Finalmente, tras verificar el mejor modo de almacenar los datos, se ha decidido hacer mediante **ionic-storage**, el cual es un plugin que permite almacenar key/value u objetos

JSON. Este plugin utiliza la mejor forma de guardar los datos según la plataforma en que se ejecuta la aplicación ya que utiliza varios sistemas de almacenamiento: sqlite, webSQL, local storage.

La razón por la que se ha optado a usar el plugin, es debido a que durante el desarrollo las pruebas mayoritariamente se realizaban mediante el navegador Chrome y por lo tanto no se podían almacenar archivos en memoria, aparte de ser la opción más adecuada para el almacenamiento de datos.

Con tal de acceder eficientemente a los datos, se han creado identificadores y variables intermedias con tal de no tener que acceder a toda la información de las sub-aplicaciones para obtener datos concretos.

- Variable con los identificadores de sub-aplicación (*subappsIndexes*)
- Cada sub-aplicación es identificada como *subapp_[INDICE]* y contiene la información básica de la sub-aplicación.
- Cada sub-aplicación tiene una variable identificada como *subapp_screens_[INDICE]* que contiene todos los índices de las pantallas que contiene.
- Cada pantalla es identificada por la variable *subapp_screen_[INDICE]_[INDICE_PANTALLA]*, la cual contiene la información de la pantalla.

#	Key	Value
3	"subapp_16"	► Object
4	"subapp_17"	► Object
5	"subapp_3"	▼ Object author: "author" descr: "description223-a-a-a-a-a" index: 3 modDate: Sat Nov 18 2017 17:43:36 GMT-0800 (PST) {} name: "name"
6	"subapp_screen_10_1"	► Object
7	"subapp_screen_10_2"	► Object
8	"subapp_screen_10_3"	► Object
9	"subapp_screen_15_1"	► Object
10	"subapp_screen_15_2"	► Object
11	"subapp_screen_15_4"	► Object
12	"subapp_screen_15_5"	► Object
13	"subapp_screen_16_1"	▼ Object ► buttons: [{...}] ► checkboxes: (4) [{...}, {...}, {...}, {...}] ► description: {text: "Aqui un texto descriptivo con algo mas", moreInfo: "mas info" ► description2: {text: "tarea-a", moreInfo: "algo mas"}}

Figura 22: Almacenamiento de datos

Proveedor de datos

Se ha creado un proveedor de datos que es el encargado de almacenar y leer los datos de ionic-storage. Las lecturas devuelven una Promise del dato en cuestión, la cual es una abstracción de programación asíncrona y provee el valor o la excepción de la función. De este modo, cada uno de los accesos que se realizan al storage es realizado asíncronamente.

Importación y exportación de sub-aplicaciones

La importación de sub-aplicaciones mediante el tag de HTML input con el type "file", el cual abre automáticamente el selector de archivos del dispositivo que se está usando. Tras la selección del archivo, se realiza una comprobación de que el tipo de fichero sea el correcto.

La exportación de sub-aplicaciones ha sido más compleja ya que el navegador, Android y iOS funcionan de forma distinta:

- Los navegadores pueden descargar archivos mediante HTML.
- Android dispone únicamente de un sistema de almacenamiento de archivos público si el dispositivo dispone de memoria externa.
- iOS no dispone de ningún tipo de almacenamiento público.

Con tal de abarcar los tres dispositivos tratados, se ha realizado lo siguiente:

Para todos ellos el primer paso, es generar un fichero para la sub-aplicación en formato BLOB.

- En el caso del navegador se realiza la descarga forzando una llamada a un elemento “a” autocreado que contiene el fichero.
- En el caso de Android se han tomado dos aproximaciones: en caso de disponer de memoria externa, se descarga en la carpeta de Descargas de ésta utilizando el plugin **File** de Ionic. En caso contrario, se muestra al usuario la opción de compartir con una aplicación (de este modo puede enviar por e-mail la sub-app)
- En el caso de iOS, se realiza la compartición del fichero, el cual entre las opciones posibles dispone de descarga.

Para realizar comparticiones se ha utilizado el plugin **social sharing** de Ionic.

Implementaciones adicionales

Con tal de disponer de un mayor feedback a la hora de realizar acciones, se han utilizado Toasts para mostrar mensajes. Por ejemplo, para indicar que una pantalla ha sido eliminada.

Durante el desarrollo de edición de pantalla, se ha detectado que no había ningún tipo de visibilidad sobre si un botón se encontraba enlazado o no, por lo que se ha añadido un icono de enlace si éstos se encuentran enlazados.

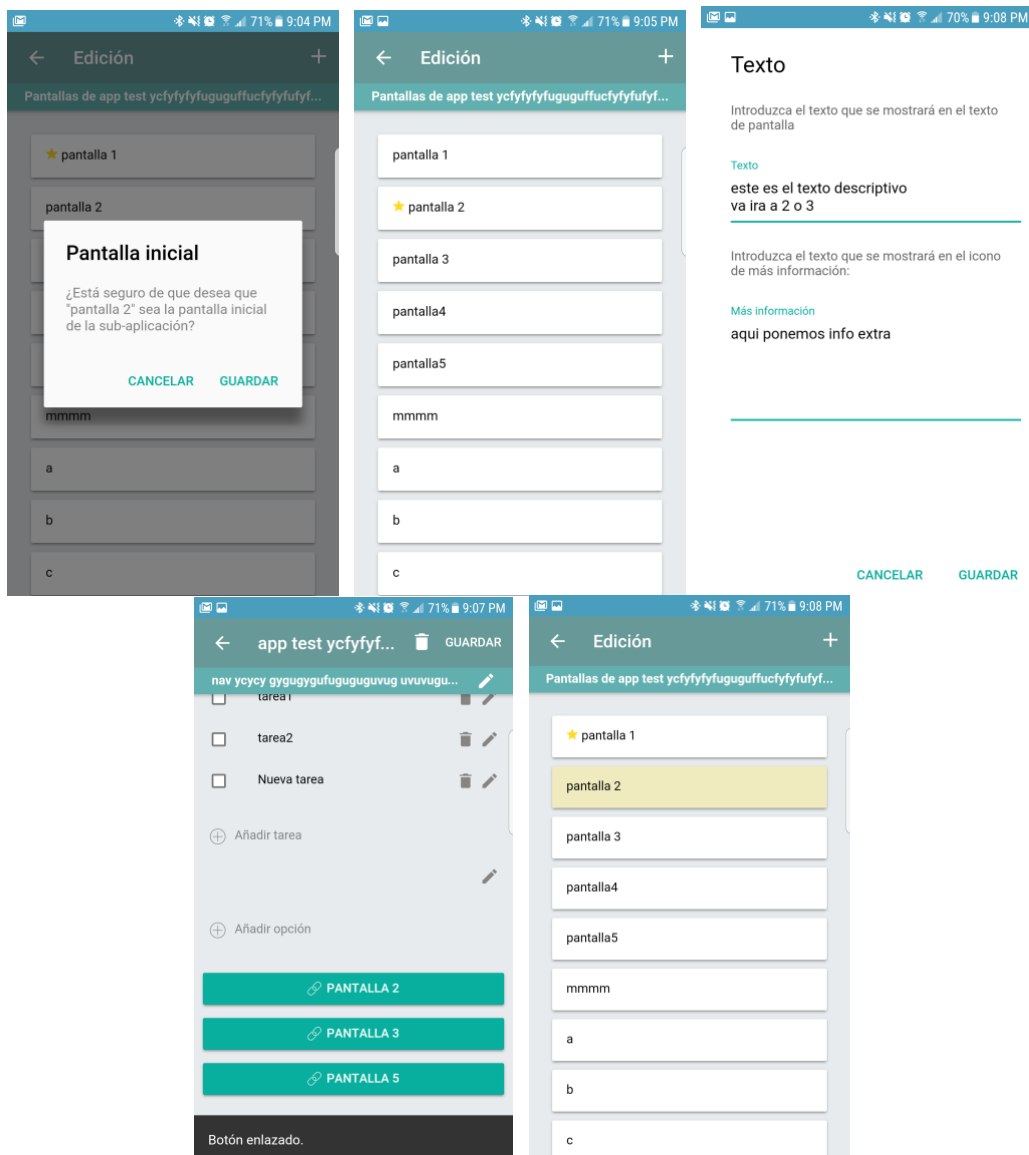
A la hora de seleccionar la pantalla a enlazar de un botón, no había manera de saber a qué pantalla estaba enlazado, por lo que se ha decidido sobresaltar el nombre de pantalla a la que está enlazado actualmente.

Al utilizar la aplicación, se ha detectado que en caso de utilizar un “alert” para introducir el texto de pantalla, éste se hacía insuficiente, por lo que finalmente se ha decidido implementar modals para la introducción de datos.

Con el fin de evitar ediciones imprevistas en edición de pantalla, se ha incorporado un botón de guardado.

Finalmente, se ha detectado de que la primera pantalla creada sería siempre el inicio de la aplicación, y en caso de ser borrada la siguiente. Debido a esto, se ha añadido la opción de asignar una pantalla como la de inicio. Para ello, es necesario realizar un long press a la pantalla que se desee como inicial y aparecerá la opción. Pero, al igual que el

problema de no saber a qué pantalla estaba enlazado un botón, no se podía saber qué pantalla era la inicial, por lo que se ha añadido una estrella a ésta.



Complicaciones

La complicación principal, que ha sido completamente inesperada durante el desarrollo, ha sido la dificultad de la exportación de datos; en la que no se esperaba la necesidad de abarcar la funcionalidad de modos distintos según el dispositivo usado.

Otra complicación ha sido la de mostrar un área de texto, ya que éstos no disponen de autosize y era imposible ver correctamente el texto que se estaba introduciendo. Tras investigar el problema, finalmente se decidió utilizar el plugin **ionic2-autosize**. El cual supuso un breve problema con los botones inferiores pero que se pudo solucionar.

Android 4.4.2 e inferiores tienen desactivada la funcionalidad de HTML input file. Debido a ello, ha sido necesario detectar la versión y utilizar el plugin **file-chooser**, para obtener la URL del archivo seleccionado. Una vez obtenido, utilizando el plugin **file-path**, se obtiene el path local del archivo y, finalmente, mediante **File** se lee el fichero.

6. Pruebas

El objetivo de estas pruebas ha sido determinar el correcto funcionamiento y visualización de la aplicación en diferentes dispositivos, además de detectar posibles problemas en la navegación y usabilidad, y garantizar una correcta implementación del diseño centrado en el usuario

Se han realizado las pruebas con un Samsung Galaxy S7 Edge (Android 7.0), Samsung Galaxy S4 Mini (Android 4.4.2), un emulador de iPhone X (11.2), y el navegador Chrome para comprobar que la aplicación funcione correctamente en los distintos tamaños de pantalla.

PU-001: Pantalla principal				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Se accede a la aplicación	Presionar el icono de la aplicación	Se muestra la pantalla principal con las opciones de ejecutar y editar	Sí	
2. Ejecutar sub-aplicación	Presionar opción de ejecutar	Es re direccionado a la pantalla de ejecución	Sí	
3. Editar sub-aplicación	Presionar opción de editar	Es re direccionado a la pantalla de edición	Sí	

PU-002: Menú				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Se accede al menú	Presionar el icono de menú	Se muestra el menú de la aplicación con las opciones de ejecución, edición, manual y sobre tDec	Sí	
1.2 Ejecutar sub-aplicación	Presionar opción de ejecutar	Es re direccionado a la pantalla de ejecución	Sí	
1.3 Editar sub-aplicación	Presionar opción de editar	Es re direccionado a la pantalla de edición	Sí	
1.4 Manual	Presionar opción de manual	Es re direccionado a la pantalla de manual	No	No implementado
1.5 Sobre tDec	Presionar opción de sobre tDec	Es re direccionado a la pantalla de	No	No implementado

		sobre tDec		
--	--	------------	--	--

PU-003: Pantalla edición				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Se accede a la pantalla de edición	Acceder al presionar la opción	Pantalla vacía con el título de la sección y las opciones de crear o importar	Sí	
2. Se accede a la pantalla de edición	Acceder al presionar la opción	Lista de sub-apps	Sí	

PU-004: Importar sub-aplicación				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Seleccionar archivo	Presionar icono de importación (nube con flecha)	Se muestra pantalla de selección de archivo	Sí	
2.1 Importación	Archivo no compatible seleccionado	Se vuelve a la pantalla de lista. Mensaje de error indicando que el formato es incorrecto	Sí	
2.2 Importación	Archivo compatible	Se vuelve a la pantalla de lista. Mensaje indicando que se ha importado correctamente. La sub-app importada aparece	Sí	

PU-005: Crear sub-aplicación				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Se accede a la opción de crear sub-aplicación	Desde edición, se presiona el icono +	Pantalla para introducir la información de la sub-aplicación	Sí	
2. Error al no introducir algún dato	Dejar algún campo vacío y presionar Guardar	Si algún dato no es introducido, aparece mensaje de error	Sí	
3. Crear sub-app	Introducir todos los datos y	Pantalla resumen de la sub-	Sí	

	presionar Guardar	aplicación		
4. Comprobar que aparece en la lista de sub-aplicaciones	Presionar botón retroceder	La nueva sub-app aparece en la lista de sub-aplicaciones	Sí	

PU-006: Edición sub-aplicación				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Acceder a edición de sub-app	Desde la lista de sub-aplicaciones de edición, presionar sobre la que se quiera acceder	Ficha resumen de la sub-app con las opciones de editar información, editar sub-app, descargar y eliminar	Sí	
2.3 Edición de información	Presionar el icono de edición	Re direccionar a la pantalla de crear/editar información	Sí	
2.4 Descargar sub-app	Presionar botón Descargar	Android con memoria externa: descargar e indicar la ruta. Android sin memoria externa: mostrar opciones de compartición. iOS: mostrar opciones de compartición (entre ellas descarga)	Sí	
2.5 Eliminar sub-app	Presionar botón eliminar	Mostrar mensaje de eliminación	Sí	
2.3.2 Cancelar	Presionar cancelar	No se elimina y se muestra un mensaje de cancelación	Sí	
2.3.3 Eliminar	Presionar Eliminar	Se elimina la sub-app, mostrando un mensaje y la lista actualizada sin la sub-app	Sí	
2.4 Editar pantallas	Presionar editar	Se muestra la pantalla de lista de pantallas	Sí	

PU-007: Editar sub-aplicación				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Se accede a la opción de editar sub-aplicación	Desde edición de sub-app se presiona el icono de edición	Pantalla para introducir la información de la sub-aplicación con la información actual	Sí	
2. Error al no introducir algún dato	Dejar algún campo vacío y presionar Guardar	Si algún dato no es introducido, aparece mensaje de error	Sí	
3. Editar sub-app	Introducir todos los datos y presionar Guardar	Pantalla resumen de la sub-aplicación con los datos cambiados	Sí	
4. Comprobar que aparece en la lista de sub-aplicaciones	Presionar botón retroceder	La nueva sub-app aparece en la lista de sub-aplicaciones con la información actualizada	Sí	

PU-008: Edición de pantallas sub-app				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Se accede a la lista de pantallas	Presionar el botón editar de edición de sub-app	Se muestra la pantalla con subtítulo el nombre la sub-app y el contenido vacío si no hay pantallas	Sí	
1. Se accede a la lista de pantallas	Presionar el botón editar de edición de sub-app	Se muestra la pantalla con subtítulo el nombre la sub-app y la lista de pantallas. Con una estrella se marca la pantalla de inicio de la sub-app	Sí	
2. Ir a edición pantalla	Presionar el	Es re direccionado	Sí	

	nombre de la pantalla a editar	a la pantalla de edición		
3. Indicar pantalla inicial	Mantener un rato pulsado el nombre de pantalla	Aparece un mensaje con la opción de guardar como pantalla inicial	Sí	
3.1 Cancelar	Presionar Cancelar	No cambia de pantalla inicial	Sí	
3.2 Guardar	Presionar Guardar	Se cambia de pantalla inicial. La pantalla seleccionada pasa a ser marcada con una estrella. La anterior pantalla inicial pierde la estrella	Sí	

PU-009: Crear pantalla de sub-app				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Crear pantalla	Presionar icono + en edición de pantallas	Muestra una pop-up con un input para introducir el nombre	Sí	
2.1 Cancelar	Presionar opción Cancelar	No crea la pantalla. La lista se mantiene igual	Sí	
2.2.1 Error crear	No introducir texto y presionar Guardar	Aparece una alerta indicando del error	Sí	
2.2.2 Crear	Introducir el nombre y presionar guardar	Crea la pantalla y re direcciona a edición de pantalla	Sí	
2.2.3	Presionar flecha de volver	La pantalla aparece en la lista de pantallas	Sí	

PU-010: Editar pantalla de sub-app				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Editar pantalla	Presionar pantalla a editar o crear pantalla nueva	Muestra la información de la pantalla. En caso de ser recién creada, los campos están	Sí	

		vacíos		
2. Editar navbar	Presionar icono de edición de la navbar	Se muestra en un modal el texto a editar	Sí	
2.1 Cancelar	Presionar Cancelar	El texto no se modifica	Sí	
2.2 Guardar	Cambiar el texto y presionar Guardar	El texto se modifica	Sí	
3. Editar título	Presionar icono de edición del título (nombre de la pantalla)	Se muestra en un modal el texto a editar	Sí	
3.1 Cancelar	Presionar Cancelar	El texto no se modifica	Sí	
3.2 Guardar	Cambiar el texto y presionar Guardar	El texto se modifica	Sí	
4. Editar texto principal	Presionar icono de edición	Se muestra un modal para introducir el texto y más información	Sí	
4.1 Cancelar	Presionar Cancelar	El texto no se modifica	Sí	
4.2.1 Guardar solo texto	Cambiar texto rellenando únicamente el campo texto	El texto se modifica y no se muestra icono de más información	Sí	
4.2.2 Guardar más info	Cambiar texto rellenando el campo de más información	El texto se modifica y se muestra icono de más información	Sí	
5. Editar título tareas	Presionar icono de edición	Se muestra un modal para introducir el texto y más información	Sí	
5.1 Cancelar	Presionar Cancelar	El texto no se modifica	Sí	
5.2.1 Guardar solo texto	Cambiar texto rellenando únicamente el campo texto	El texto se modifica y no se muestra icono de más información	Sí	
5.2.2 Guardar más info	Cambiar texto rellenando el campo de más información	El texto se modifica y se muestra icono de más información	Sí	

6. Añadir tarea	Presionar opción de añadir tarea	Aparece una nueva tarea con el título “Nueva tarea”	Sí	
7. Editar tarea	Presionar icono de edición de la tarea en cuestión	Se muestra un modal para introducir el texto y más información	Sí	
7.1 Cancelar	Presionar Cancelar	El texto no se modifica	Sí	
7.2.1 Guardar solo texto	Cambiar texto rellenando únicamente el campo texto	El texto se modifica y no se muestra icono de más información	Sí	
7.2.2 Guardar más info	Cambiar texto rellenando el campo de más información	El texto se modifica y se muestra icono de más información	Sí	
8. Eliminar tarea	Presionar icono de eliminar	Mensaje indicando si quiere proceder	Sí	
8.1 Cancelar	Presionar cancelar	No elimina	Sí	
8.2 Eliminar	Presionar eliminar	Elimina y desaparece de la lista	Sí	
9. Editar título opciones	Presionar icono de edición	Se muestra un modal para introducir el texto y más información	Sí	
9.1 Cancelar	Presionar Cancelar	El texto no se modifica	Sí	
9.2.1 Guardar solo texto	Cambiar texto rellenando únicamente el campo texto	El texto se modifica y no se muestra icono de más información	Sí	
9.2.2 Guardar más info	Cambiar texto rellenando el campo de más información	El texto se modifica y se muestra icono de más información	Sí	
10. Añadir opción	Presionar opción de añadir opción	Aparece una nueva opción con el título “Nueva opción”	Sí	
11. Editar opción	Presionar icono de edición de la opción en cuestión	Se muestra un modal para introducir el texto y más información	Sí	

11.1 Cancelar	Presionar Cancelar	El texto no se modifica	Sí	
11.2.1 Guardar solo texto	Cambiar texto rellenando únicamente el campo texto	El texto se modifica y no se muestra icono de más información	Sí	
11.2.2 Guardar más info	Cambiar texto rellenando el campo de más información	El texto se modifica y se muestra icono de más información	Sí	
12. Eliminar opción	Presionar icono de eliminar	Mensaje indicando si quiere proceder	Sí	
12.1 Cancelar	Presionar cancelar	No elimina	Sí	
12.2 Eliminar	Presionar eliminar	Elimina y desaparece de la lista	Sí	
13. Añadir botón	Presionar opción de añadir botón	Muestra un nuevo botón con el texto "Nuevo botón"	Sí	
14. Editar botón	Presionar el botón a editar	Aparece un menú en la parte inferior de la pantalla con las opciones: añadir/editar enlace, editar texto, eliminar enlace, eliminar botón, cancelar.	Sí	
14.1 Cancelar	Presionar Cancelar	Cierra el menú y el botón no es modificado	Sí	
14.2 Añadir/Editar enlace	Presionar añadir/editar enlace	Muestra una pantalla con la lista de pantallas de la sub-app	Sí	
14.3 Eliminar enlace	Presionar eliminar enlace	Muestra mensaje de confirmación	Sí	
14.3.1 Cancelar	Presionar cancelar	No elimina el enlace	Sí	
14.3.2 Eliminar	Presionar eliminar	Elimina el enlace del botón y desaparece el icono de enlace	Sí	
14.4 Eliminar botón	Presionar eliminar botón	Muestra mensaje de confirmación	Sí	
14.4.1 Cancelar	Presionar	No elimina el	Sí	

	cancelar	botón		
14.4.2 Eliminar	Presionar eliminar	Elimina el botón y desaparece de la lista	Sí	
14.4 Editar texto botón	Presionar editar texto	Se muestra en un modal el texto a editar	Sí	
14.4.1 Cancelar	Presionar Cancelar	El texto no se modifica	Sí	
14.4.2 Guardar	Cambiar el texto y presionar Guardar	El texto se modifica	Sí	
15. Guardar cambios	Presionar opción GUARDAR	Se muestra mensaje conforme se ha guardado. Se vuelve a la lista de pantallas. Si se ha cambiado el nombre éste se ve modificado.	Sí	
16. Volver sin guardar	Se realizan cambios y sin presionar guardar se vuelve hacia atrás	Los cambios no son guardados. Si se ha cambiado el nombre de la pantalla, éste no se ve modificado en la lista	Sí	
17. Eliminar pantalla	Presionar icono de eliminar	Se muestra mensaje de confirmación	Sí	
17.1 Cancelar	Presionar Cancelar	No se elimina la pantalla	Sí	
17.2	Presionar Eliminar	Se elimina la pantalla mostrando un mensaje, se vuelve a la lista de pantallas y ésta no aparece en la lista	Sí	

PU-011: Enlazar botón a pantalla				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Pantalla enlace	Presionar opción de enlazar de un botón en edición de pantalla	Muestra pantalla con la lista de pantallas de la sub-app y la	Sí	

		opción de crear nueva. Si el botón estaba previamente enlazado, éste se sobresalta con un color amarillo/marrón.		
2. Enlazar a nueva pantalla	Presionar icono +	Aparece un pop-up con un input para introducir el nombre de la nueva pantalla	Sí	
2.1 Cancelar	Presionar cancelar	No se crea la pantalla y por lo tanto no es enlazado. Se continúa mostrando la lista de pantallas	Sí	
2.2 Error	No introducir texto y presionar Guardar	Muestra mensaje de error	Sí	
2.3 Crear	Introducir texto y Guardar	Crea la nueva pantalla. El botón es enlazado a ésta. Es re direccionado a la edición de pantalla del botón. El botón se muestra con un icono de enlace	Sí	
3. Enlazar con pantalla	Presionar el nombre de la pantalla con el que se quiere realizar el enlace	El botón es enlazado a la pantalla. Es re direccionado a la edición de pantalla del botón. El botón se muestra con un icono de enlace	Sí	

PU-012: Pantalla ejecución				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Se accede a la pantalla de ejecución	Acceder al presionar la opción	Pantalla vacía con el título de la sección	Sí	

1. Se accede a la pantalla de ejecución	Acceder al presionar la opción	Lista de sub-apps disponibles	Sí	
2. Seleccionar sub-app	Seleccionar sub-app a ejecutar	Es re direccionado a la pantalla de información de la sub-app	Sí	

PU-013: Pantalla ejecución sub-app (info)				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Información sub-app	Seleccionar sub-app a ejecutar desde la lista de sub-apps	Información de la sub-app y opción de ejecutar	Sí	
2. Ejecutar	Presionar botón Ejecutar	Es re direccionado a la pantalla inicial de la sub-app	Sí	

PU-014: Pantalla ejecución sub-app (ejecución)				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Ejecutar	Presionar botón Ejecutar en la pantalla de información	Muestra la pantalla inicial de la sub-app	Sí	
2. Seleccionar botón	Si hay opciones y tareas hacer las selecciones pertinentes para que se puedan usar los botones. Presionar un botón	Es re direccionado a la pantalla a la que estaba enlazado	Sí	
3. Home	Presionar icono Home	Vuelve a la pantalla principal de la aplicación	Sí	
4. Más información	Presionar el icono de más información	Se muestra un modal con el texto de más información del campo en cuestión	Sí	
4.1 Cerrar	Presionar cerrar	Cierra el modal	Sí	

PU-015: Verificación pantalla inicial				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Edición	Seleccionar editar sub-app, seleccionar de la	La nueva pantalla es marcada con una estrella	Sí	

	lista la nueva pantalla inicial			
2. Ejecución	Seleccionar ejecutar sub-app	La primera pantalla es la pantalla que hemos indicado en la edición	Sí	

PU-016: Verificación modificar pantalla				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Edición	Seleccionar editar pantalla de sub-app, cambiar datos y guardar	Se guardan los cambios	Sí	
2. Ejecución	Seleccionar ejecutar sub-app, continuar app hasta llegar a la pantalla modificada	La pantalla contiene los cambios que se habían hecho en la edición	Sí	

PU-017: Verificación enlace				
Descripción	Entrada	Salida esperada	OK?	Observaciones
1. Edición	Seleccionar editar pantalla de sub-app, cambiar enlace de botón y guardar	Se guardan los cambios	Sí	
2. Ejecución	Seleccionar ejecutar sub-app, continuar app hasta llegar a la pantalla modificada. Presionar botón modificado en edición	Se re direcciona a la pantalla seleccionada durante la edición de enlace	Sí	

7. Conclusiones

Tras la realización de este trabajo se ha conseguido soltura al desarrollar aplicaciones híbridas mediante el framework de Ionic. Gracias a ello, las próximas aplicaciones o mejoras en la propia aplicación, podrán realizarse más rápidamente al no necesitar investigación previa de conceptos básicos. También se ha aprendido a generar aplicaciones para las plataformas de Android y iOS a dispositivos y emuladores, así como a utilizar nuevos plugins, y escoger el mejor mecanismo a utilizar para realizar una tarea en un proyecto multidispositivo (almacenamiento de datos, descarga de ficheros...).

Todos los objetivos que se habían planteado inicialmente se han cumplido. El único objetivo que no se ha cumplido cien por cien como se había especificado es la opción de descarga de la sub-aplicación. No todos los dispositivos permiten almacenar archivos en el sistema de ficheros, por lo que, según el dispositivo en el que se esté utilizando la aplicación, se toma una aproximación u otra: Android descarga el fichero en la carpeta Descarga de la memoria externa y, si no dispone de ella, permite compartir el archivo. iOS permite compartir el archivo (opción que incluye almacenaje). Y finalmente, el navegador permite descargar el archivo.

La planificación del trabajo y la metodología especificada ha sido seguida a lo largo del desarrollo del trabajo.

Al disponer de tareas tan específicas y de tiempo limitado, el método “waterfall” ha permitido desarrollar cada uno de los componentes y realizar sus pruebas tal y como se habían planificado.

La planificación ha variado un poco desde su especificación inicial. Se ha añadido el desarrollo de la pantalla principal, que no se había tenido en cuenta en un principio, se ha movido el desarrollo de importar/exportar sub-aplicación a la fase de implementación, y se han añadido nuevas funcionalidades: indicar pantalla principal e indicar pantalla enlazada (fase de implementación), sobre tDec y manual (fase final de implementación).

El mover la funcionalidad de importar/exportar a una fase previa se realizó ya que al investigar sobre el tema se detectó que podría traer algún problema, por lo que se pensó que sería más adecuado realizarlo al principio, con tiempo, que al final del desarrollo.

Se añadió la funcionalidad de indicar la pantalla inicial de la sub-aplicación ya que, al realizar pruebas, se detectó que tendría más sentido poder indicar qué pantalla es la inicial, a forzar que fuera la primera que se creó.

Indicar si un botón se encuentra enlazado se añadió ya que al realizar pruebas se detectó que no era posible saber si un botón se encontraba o no enlazado, ni a qué pantalla lo estaba.

Finalmente, los apartados sobre tDec y manual, se añadieron para dar información sobre la aplicación y explicar cómo utilizarla.

Futuras mejoras

La aplicación puede incorporar una gran cantidad de mejoras y de nuevas funcionalidades que podrían hacer de esta aplicación un muy buen producto en el mercado:

Añadir un servidor donde se registrarían los usuarios y se almacenarían sus sub-aplicaciones (permitiría el uso multidispositivo).

Sistema real de compartición de sub-aplicaciones (sin necesidad de importar y exportar). La sub-aplicación se compartiría directamente mediante usuarios de tDec dentro de la propia aplicación.

Mejorar el sistema de creación de la sub-aplicación en navegadores y tabletas. Las pantallas se podrán crear y enlazar de forma más gráfica (mediante rectángulos y flechas).

Sistema de monetización in-app para la compartición y almacenamiento de sub-aplicaciones (freemium).

Incorporación de estilos en la sub-aplicación (color y fuente de la sub-aplicación).

“No hay límites salvo los impuestos por sus propios pensamientos”.

Frases de **Brian Tracy**

8. Glosario

tDec: Aplicación que permite la creación de sub-aplicaciones basadas en decisión (preguntas/respuestas) y ejecutarlas en ella como si de una aplicación se tratara.

sub-aplicación: aplicación generada dentro de una aplicación principal.

stakeholders: personas u organizaciones involucradas de manera formal e informal con la solución de negocio ofrecida en la experiencia del cliente.

Zingtree: aplicación capaz de generar programas online para la toma de decisiones y se basa en un modelo de negocio de suscripción mensual.

ionic: es un SDK HTML5, construido sobre Angular, que ayuda a desarrollar aplicaciones móviles nativas utilizando tecnologías web como HTML, CSS y Javascript.

Framework: estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software.

waterfall: modelo de desarrollo del software donde existen fases muy marcadas. El desarrollo se interpreta como el agua que va cayendo de un estanque al siguiente.

MVC: patrón de arquitectura (Modelo Vista Controlador), el cual separa los datos y la lógica de la interfaz de usuario.

JSON: acrónimo de JavaScript Object Notation, es un formato de texto ligero para el intercambio de datos.

JS (Javascript): lenguaje de programación interpretado. Se utiliza principalmente en el lado del cliente.

PhoneGap: Framework para el desarrollo de aplicaciones móviles producido por Nitobi. Permite a los programadores desarrollar aplicaciones para dispositivos móviles utilizando herramientas genéricas tales como Javascript, HTML5, CSS3.

Cordova: Apache Cordova (versión de código abierto de PhoneGap) permite, a los programadores de software, desarrollar aplicaciones para dispositivos móviles utilizando Javascript, HTML5, CSS3.

AngularJS: framework de JavaScript de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página.

sqlite: sistema de gestión de base de datos relacional compatible con ACID.

webSQL: API para almacenar datos en bases de datos que pueden ser obtenidos mediante queries de variantes de SQL.

local storage: almacenamiento local de aplicaciones web.

Promise: abstracción de programación asíncrona que provee el valor o la excepción de la función.

BLOB (Binary Large Object): objetos binarios de gran tamaño

9. Bibliografía

IBM, Native, web or hybrid mobile-app development, IBM Software, WSW14182-USEN-01

<ftp://public.dhe.ibm.com/software/pdf/mobile-enterprise/WSW14182USEN.pdf>

Functional requirement: Wikipedia (06/10/2017)

https://en.wikipedia.org/wiki/Functional_requirement

Non-functional requirement: Wikipedia (06/10/2017)

https://en.wikipedia.org/wiki/Non-functional_requirement

Zingtree (07/10/2017):

<https://zingtree.com/>

Ionic v2 (09/10/2017)

<https://ionicframework.com/docs/>

Andrew McGivern. Structure of an Ionic class (26/10/2017)

<http://mcgivern.com/structure-of-an-ionic-app/>

MVC. Wikipedia (27/10/2017)

<https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>

MVC Diagram. Wikimedia (27/10/2017)

[https://commons.wikimedia.org/wiki/File:MVC_Diagram_\(Model-View-Controller\).svg](https://commons.wikimedia.org/wiki/File:MVC_Diagram_(Model-View-Controller).svg)

Eva Patricia Gil, Eva de Lera Tatjer i Antònia Monjo Palau. *Usuarios y sistemas interactivos*. UOC PID_00143566

Robert Ramírez Vique. *Métodos para el desarrollo de aplicaciones móvil*. UOC PID_00176755

Sara Salmerón i Roger Montserrat. *Modelos de negocio y marketing basados en dispositivos móviles*. UOC

Ionic Storage (05/11/2017)

<https://ionicframework.com/docs/storage/>

Ionic File (05/11/2017)

<https://ionicframework.com/docs/native/file/>

Ionic Social Sharing (19/11/2017)

<https://ionicframework.com/docs/native/social-sharing/>

Ionic2 Autosize (18/11/2017)

<https://www.npmjs.com/package/ionic2-autosize>

Ionic Status Bar (03/12/2017)

<https://ionicframework.com/docs/native/status-bar/>

Sitepoint: Storing Local Data in a Cordova App

<https://www.sitepoint.com/storing-local-data-in-a-cordova-app/>

OurCodeWorld: How to save a PDF from a base64 string on the device with Cordova
(19/11/2017)

<https://ourcodeworld.com/articles/read/230/how-to-save-a-pdf-from-a-base64-string-on-the-device-with-cordova>

Ionic File Chooser (18/12/2017)

<https://ionicframework.com/docs/native/file-chooser/>

Ionic File Path (18/12/2017)

<https://ionicframework.com/docs/native/file-path/>

Ionic In App Browser (22/12/2017)

<https://ionicframework.com/docs/native/in-app-browser/>

10. Anexos

Junto con el documento se encuentra disponible una carpeta llamada anexos, en la cual se podrá encontrar:

- Diagrama de Gantt completo de la planificación del trabajo.
- design: Diseño de las pantallas principales en Sketch.
- Manual de usuario: manual explicativo del funcionamiento de la aplicación.
- Ejemplo de sub-aplicación que puede ser importada en tDec: tDec-subapp-Diagnóstico SCAEST.tdec