

List2uS: Share with friends

Maria Magdalena Freixa Torres

Resumen— La rutina es parte de nuestra vida diaria, son acciones que realizamos día a día para llegar siempre al mismo objetivo. Un ejemplo entre muchos, es el de realizar la lista de la compra. Hoy en día con la modernización de prácticamente todas las tareas, se hace muy tedioso tener que cargar con un bloc de notas, bolígrafo y encontrar una superficie estable para poder escribir.

Con esta aplicación para dispositivos Android no sólo ponemos remedio a ese inconveniente, sino que también te ahorramos el paso de tener que hacer varias copias si tienes que compartir la lista con tu familia o amigos. List2uS te permite, en unos pocos pasos, crear y compartir en tiempo real cualquier lista que necesites.

Palabras clave— Gestión de listas de la compra, compartir entre familiares o amigos, aplicación Android, App.

Abstract— The routine is part of our daily life, there are actions we take every day to always reach the same goal. One example among many, is to make the shopping list. Today with the modernization of virtually all tasks, it becomes very tedious to have to carry a notebook, pen and find a stable surface to write.

With this application for Android, not only we put remedy to that problem, also save you the step of having to make multiple copies if you have to share the list with your family or friends. List2uS allows you, in a few steps, create and share in real time any list you need.

Index Terms— Management of shopping lists, shared among family or friends, Android application, App.



1 INTRODUCCION

LIST2US es un sistema con el que se gestionan listas de la compra. El sistema puede añadir, editar, eliminar y compartir listas de la compra, con cualquier tipo de usuario. Así como enviar y recibir notificaciones sobre las listas que comparte o que son compartidas contigo.

1.1 Objetivos

El objetivo principal es desarrollar una aplicación para dispositivos móviles Android para la gestión de listas de la compra.

Este sistema ha de permitir que cualquier usuario pueda iniciar sesión, registrarse y gestionar su cuenta de usuario.

También ha de poder gestionar sus listas de la compra, es decir, crear, eliminar, editar sus listas y poder compartirlas con otros usuarios, así como recibir notificaciones a través de correo electrónico.

Como objetivo secundario, se quiere que el sistema pueda gestionar recordatorios. Por ejemplo si una persona ha de comprar cierto producto cada cierto tiempo, que el sistema notifique de la compra que se ha de realizar.

A continuación en la Tabla 1 se muestran los diferentes objetivos iniciales del proyecto.

	Crítico	Prioritario	Secundario
Diseño de la gestión de usuarios	X		
Diseño de la gestión de listas de la compra	X		
Determinar de un método de comunicación Web Services	X		
Determinar la compartición de las listas de la compra	X		
Especificar, a quien se comparte una lista para poder diferenciarlas en el módulo de listas privadas	X		
Aplicación sencilla y amigable para todo tipo de usuarios		X	
Determinar la información necesaria del usuario		X	

- E-mail de contacto: mery.blood@gmail.com
- Mención realizada: Ingeniería del Software
- Trabajo tutorizado por: Yolanda Bénitez
- Curso 2015/16

	Crítico	Prioritario	Secundario
Especificar, para cada producto de la lista de la compra, el estado del producto, realizado/inacabado		X	
Gestión de recordatorios			X
Notificaciones por e-mail		X	

Tabla 1: Objetivos

2 ESTADO DEL ARTE

La utilización de listas de la compra es una tarea que prácticamente utilizamos en nuestro día a día, existen muchas soluciones ya implementadas, como más destacadas tenemos:

2.1 Wunderlist

Una de las soluciones más completas es Wunderlist con una puntuación de 4.5 y entre 5 millones y 10 millones de instalaciones. Es una aplicación para dispositivos con sistema operativo Android e iOS, también dispone de una versión web. La aplicación te permite planificar y compartir cualquier tarea, trabajo, lista de la compra, etc. A parte tiene la posibilidad de establecer recordatorios para las diferentes tareas.

El gran inconveniente de esta aplicación es que tiene un sistema de suscripción para poder utilizar completamente todas sus funcionalidades.

2.2 Lista de la compra

Otra de las soluciones más usada y disponible en Google Play es Lista de la compra con una puntuación de 4.5 y entre 1 millón y 5 millones de instalaciones. Con esta aplicación se pueden crear listas de la compra, recetas y tareas pendientes, dispone de sincronización automática y recordatorios para las compras.

Uno de los mayores inconvenientes es que no se pueden utilizar todas las funcionalidades de forma gratuita, de las cuales, las más importantes son la sincronización de las listas e introducir recordatorios.

2.3 Lista de compras

Por último, tenemos lista de compras con una puntuación de 4.1 y entre 1 millón y 5 millones de instalaciones. Con esta aplicación se pueden crear listas de la compra y añadir productos mediante voz o código de barras, es una aplicación muy intuitiva y fácil de utilizar.

Como inconveniente tenemos que no proporciona ningún tipo de sincronización, por lo tanto nuestras listas se perderían al cambiar de terminal o cuenta.

3 METODOLOGÍA

La metodología que se ha seguido para la realización del proyecto ha sido una versión adaptada de la metodología SCRUM [1], ya que es una metodología ágil e iterativa, donde si se produce cualquier cambio inesperado es más sencillo de tratar.

En este proyecto, ya que no existe un cliente específico, el rol de *Product Owner* lo ha representado la tutora del proyecto.

Así pues, los *sprints* han sido determinados por la planificación de la asignatura ‘Trabajo Final de Grado’, donde en cada sprint se ha realizado el seguimiento del proyecto, presentando los cambios, avances y planificando el trabajo a realizar para el siguiente sprint.

4 PLANIFICACIÓN

List2uS	
Sprint 0	
Propuesta de aplicación	
Objetivos iniciales	
Documentación	
Preparación de la documentación	
Primera reunión	
Sprint 1	
Investigación de aplicaciones similares	
Especificación de los objetivos	
Metodología	
Planificación	
Documentación	
Requisitos iniciales	
Informe inicial	
Sprint 2	
Creación del servidor externo	
Diseño de la base de datos	
Diseño de la Interfaz de la aplicación	
Comenzar la implementación	
Documentación	
Informe de progreso I	
Sprint 3	
Continuación de la implementación	
Testing	
Documentación	
Informe de progreso II	
Sprint 4	
Testing	
Depuración de errores	
Documentación	
Propuesta del artículo	
Defensa del proyecto	
Documentación	
Entregas finales	
Poster	
Preparación de la defensa	
Presentación de la defensa	

Tabla 2: Planificación

La Tabla 2 corresponde a la planificación inicial del proyecto para realizar y cumplir tanto los objetivos como los requisitos de la aplicación.

Cabe destacar que la planificación inicial se ha ido siguiendo a lo largo de todo el proyecto, sin embargo algunos objetivos han requerido más tiempo del planificado retrasando otros.

5 REQUISITOS

5.1 Requisitos funcionales

Registro

1. La aplicación debe permitir al usuario introducir sus datos en el formulario del registro.

- a. Nombre de usuario.
- b. Correo electrónico.
- c. Contraseña.

2. El sistema validara los datos introducidos por el usuario.

3. Si alguno de los datos introducidos es inválido, el sistema mostrará un mensaje de error.

4. Si la validación de los datos introducidos es correcta, el sistema se encargará de introducir todos los datos del usuario nuevo a la base de datos.

5. El sistema enviara un correo electrónico conforme se ha registrado en la aplicación.

6. Una vez registrado correctamente el usuario, el sistema le redirigirá a la pantalla principal de la aplicación.

Inició de sesión

1. Para poder acceder a la aplicación el usuario deberá identificarse introduciendo:

- a. Correo electrónico.
- b. Contraseña.

2. En el caso de que el usuario ya se haya identificado con anterioridad la identificación será automática cada vez que se inicie la aplicación.

3. El sistema se encargara de validar los datos introducidos para la identificación.

4. En caso que la validación de los datos no sea la correcta, el sistema mostrara un mensaje de error.

5. Una vez identificado correctamente el usuario, el sistema le redirigirá a la pantalla principal de la aplicación.

Cierre de sesión

1. Cualquier usuario identificado debe poder finalizar sesión en la aplicación.

2. En el caso de que el usuario pulse el botón de cerrar sesión, el sistema mostrará un mensaje para asegurarse que el usuario quiere cerrar su sesión.

3. En caso de cierre de sesión el sistema redirigirá al usuario a la pantalla inicial de identificación.

Perfil de usuario

1. Se dispondrá de una pantalla de perfil de usuario, donde se permita al usuario consultar o modificar sus datos de cuenta.

2. Si se modifica algún dato de la cuenta, el sistema lo validará.

3. En caso de que la validación sea correcta, el sistema procederá a actualizar la base de datos.

Menú lateral

1. A partir del menú lateral se podrá acceder a las siguientes áreas de la aplicación:

- a. Listas privadas
- b. Listas compartidas
- c. Recordatorios
- d. Perfil de usuario
- e. Cerrar sesión

2. Se podrá visualizar la foto de perfil, el nombre y correo electrónico del usuario identificado.

Listas privadas

1. La aplicación debe permitir al usuario añadir, editar y eliminar listas.

2. En el caso que se quiera eliminar una lista, el sistema mostrará un mensaje para asegurar que el usuario desea eliminar la lista.

3. Al seleccionar una lista el sistema redirigirá al usuario dentro de la lista donde se podrá visualizar los productos introducidos.

4. En todos los casos, el sistema actualizara la base de datos con la información pertinente.

Productos

1. El sistema debe permitir al usuario añadir, editar y eliminar productos de la lista seleccionada con anterioridad.

2. El sistema también debe permitir compartir dicha lista mediante un botón de compartir.

3. En el caso que se quiera compartir la lista de productos, el sistema pedirá el correo electrónico del usuario, que validará y actualizará la base de datos con la nueva información.

4. En caso de que se produzca la compartición de la lista, el sistema añadirá automáticamente dicha lista en la sección de Listas compartidas.

5. En todos los casos, el sistema actualizara la base de datos con la información pertinente.

Listas compartidas

1. El sistema debe permitir eliminar una lista compartida.

2. En el caso que se quiera eliminar una lista, el sistema mostrará un mensaje para asegurar que el usuario desea eliminar la lista.

3. El sistema debe permitir añadir, modificar, eliminar productos.

4. En el caso que se quiera eliminar un producto, el sistema mostrará un mensaje para asegurar que el usuario desea eliminar dicho producto.

5. En todos los casos, el sistema actualizara la base de datos con la información pertinente.

Recordatorios

1. El usuario podrá añadir, modificar y eliminar recordatorios.
2. En el caso que se quiera eliminar un recordatorio, el sistema mostrará un mensaje para asegurar que el usuario desea eliminarlo.
3. El sistema deberá notificar al usuario en la fecha indicada por el del recordatorio introducido.

Notificaciones

1. El sistema deberá lanzar alarmas o notificaciones en los casos siguientes:

- a. Avisos al añadir nuevos productos a la lista compartida.
- b. Avisos al modificar productos a la lista compartida.
- c. Avisos al eliminar productos a la lista compartida.
- d. Avisos de los recordatorios introducidos por el usuario.

5.2 Requisitos no funcionales

Interfaz y usabilidad

1. La aplicación debe disponer de una interfaz sencilla, intuitiva y atractiva para el usuario.
2. La introducción de datos debe estar estructurada para una mayor comodidad para el usuario.

Rendimiento

1. Los accesos a base de datos no suponen una gran carga para el dispositivo.
2. Se esperan tiempos de respuesta no superiores a 2 segundos respecto a las diferentes peticiones al servidor como a las consultas y/o modificaciones de la base de datos.

Seguridad

1. Para poder usar la aplicación se debe estar registrado en el sistema.
2. Por cada dispositivo solo será necesario un inicio de sesión.
3. Los datos de usuario serán cifrados.

6 PRESUPUESTO

6.1 Costes

En la Tabla 3 se muestran los costes de personal, teniendo en cuenta que el proyecto consta de unas 1060 horas.

Cargo	Horas	Euros/Hora	Coste
Jefe de proyecto	100	28€/h	2800€
Programador Java	300	10€/h	3.000€
Programador Android	300	18€/h	5.400€
Analista	160	20€/h	3.200€
Diseñador	50	18€/h	900€
Total:			15.300€

Tabla 3: Costes

6.2 Coste directo

En la Tabla 4 se muestra una estimación del coste directo del proyecto.

Concepto	Horas proyecto	Coste
Nomina personal	1060 horas	15.300€
Seguridad social (30,9%)	1060 horas	4.727,7€
Coste directo total:		20.027,7€

Tabla 4: Coste directo

6.3 Coste indirecto

En la Tabla 5 se muestra una estimación del coste indirecto del proyecto.

Concepto	Coste
Compas Software	145,14€
Gasto mantenimiento	1.000€
Licencias específicas	200€
Material	4.300€
Coste indirecto Total	5.645,14€

Tabla 5: Coste indirecto

Como compra de Software tenemos Adobe Photoshop CC 2015 por 6 meses a 24,19€/mes, como material tenemos 4 ordenadores con todos sus componentes y periféricos, junto con una impresora y tres dispositivos Android para las pruebas, y a parte contamos con las 4 licencias de Windows 7 Professional.

6.4 Estimación del margen de beneficio

En la Tabla 6 se muestra una estimación total, incluyendo el margen del beneficio del proyecto.

Concepto	Coste
Coste directo	20.027,7€
Coste indirecto	5.645,14€
Margen de beneficio (10%)	2.567,3€
Coste total del proyecto:	28.240,2€

Tabla 6: Estimación margen de beneficio

7 IMPLEMENTACIÓN

En este apartado se explicará la implementación, las diferentes funcionalidades y el diseño final de la aplicación List2uS.

7.1 Inicio de sesión



Fig. 1: Inicio de sesión

La Fig. 1 hace referencia al inicio de sesión del usuario, es la pantalla que se mostrará cuando un usuario entre a la aplicación y no haya iniciado sesión anteriormente.

Como se puede observar para poder iniciar sesión el usuario deberá introducir el correo electrónico y la contraseña que habrá utilizado anteriormente para registrarse en la aplicación.

Si el usuario introduce la información correctamente y los datos son válidos, se abrirá la actividad principal que podemos ver en la Fig. 3.

Si el usuario no está registrado deberá hacer clic en registrarse donde se abrirá la actividad de registro que se puede ver en la Fig. 2

7.2 Registro



Fig. 2: Registro

La Fig. 2 hace referencia al registro de un nuevo usuario, para eso se deberá introducir un nombre de usuario, un correo electrónico, una contraseña y repetir la misma contraseña para poder verificar la misma.

Si el usuario ha introducido correctamente la información, es decir, si el nombre de usuario y el correo electrónico existen en la base de datos y tienen el formato correcto y las 2 contraseñas coinciden se procederá al registro del usuario y el sistema abrirá la actividad principal que se puede ver en la Fig. 3.

Si el usuario se ha equivocado al darle clic al botón de registrarse, podrá volver a la actividad de inicio de sesión haciendo clic al botón de cancelar.

7.3 Actividad principal

Esta actividad es la encargada de controlar los fragmentos [2] del resto de la aplicación.

Un fragmento es un objeto pensado para ser cargado desde una actividad y sustituirla. De este modo, lo que hacemos es “añadir” un fragmento a la actividad, que es el que muestra la interfaz gráfica e incluye la implementación de esta nueva vista.

Esto lógicamente tiene muchas implicaciones, pero la principal ventaja que tiene con respecto al modelo Vista-actividad es que dentro de una sola actividad, en nuestro caso la actividad principal, podemos lanzar varios fragmentos que interactúan entre ellos a través de dicha actividad, y de este modo nos ahorraremos tener que estar todo el rato cambiando entre distintas actividades, con el consecuente gasto de memoria.

7.4 Fragmento principal

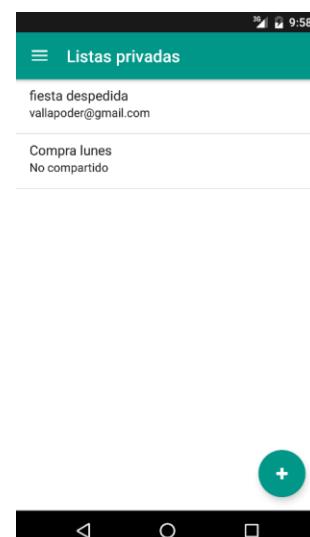


Fig. 3: Fragmento principal

La Fig. 3 corresponde al fragmento principal que se mostrará cuando el usuario haya iniciado sesión, registrado, o ya haya iniciado sesión con anterioridad.

Esta es la sección de listas privadas donde podremos observar nuestras listas y aquellas que hayamos compartido con otro usuario. Para diferenciarlas tenemos un subtítulo con el correo electrónico de la persona con la que hemos compartido la lista, o no compartido si esa lista no ha sido compartida con ningún usuario.

Si hacemos clic en una de las listas nos llevará al fragmento de la Fig. 5 donde tendremos los productos de dicha lista.

Si hacemos clic en el *FloatingActionButton(+)* [3], se abrirá un dialogo para poder añadir una lista nueva, y al hacer clic en aceptar se añadirá automáticamente al resto de listas.

En cambio, si hacemos un *LongClick* (un clic largo), nos aparecerá otro dialogo con las opciones que podemos realizar con una lista, editar, eliminar y compartir.

Si hacemos clic en eliminar nos mostrará un dialogo, para confirmar que queremos eliminar la lista definitivamente, si hacemos clic en editar nos mostrara el dialogo, para poder cambiar el nombre de dicha lista.

Finalmente, si hacemos clic en compartir nos mostrará un dialogo, donde deberemos introducir el correo electrónico del usuario con el cual queremos compartir la lista.

Cabe añadir que tanto al compartir una lista con otro usuario o eliminar una lista ya compartida, el usuario afectado recibirá un correo electrónico con las acciones realizadas en la lista que tiene compartida.

7.5 Listas compartidas

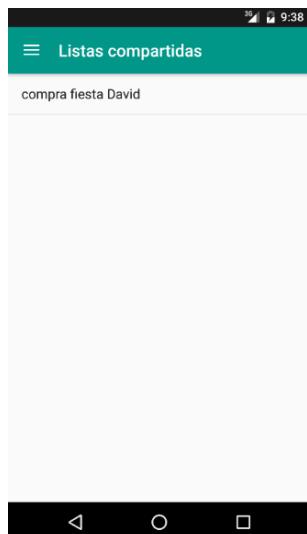


Fig. 4: Listas compartidas

La Fig. 4 corresponde al fragmento de listas compartidas,

donde se mostrarán las listas que han compartido con nosotros, estas listas podrán ser editadas o eliminadas haciendo un *LongClick* (un clic largo), que nos mostrará un diálogo con las opciones correspondientes.

Si hacemos clic en eliminar nos mostrará un diálogo, para confirmar que queremos eliminar la lista definitivamente, si hacemos clic en editar nos mostrará el diálogo, para poder cambiar el nombre de dicha lista.

En cambio, si hacemos clic en una de las listas nos llevará al fragmento de los productos de dicha lista.

Cabe añadir que para la eliminación la de las listas compartidas se enviará un correo electrónico al usuario afectado.

7.6 Productos

Este fragmento contiene los productos de la lista seleccionada como podemos observar en la Fig. 5



Fig. 5: productos

Si hacemos un clic en uno de los productos, este se modificará de forma que se pueda visualizar que el producto ha sido comprado como se puede ver en la Fig. 6

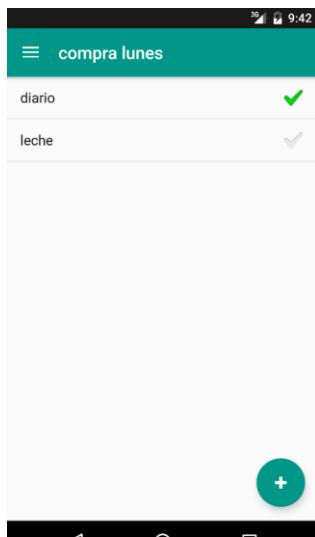


Fig. 6: Producto comprado

Si hacemos clic en el *FloatingActionButton*(+), se abrirá un dialogo para poder añadir un nuevo producto a la lista y al hacer clic en aceptar se añadirá automáticamente al resto de productos.

En cambio, si hacemos un *LongClick* (un clic largo), nos aparecerá otro dialogo con las opciones que podemos realizar con un producto, editar y eliminar.

Si hacemos clic en eliminar nos mostrará un diálogo, para confirmar que queremos eliminar el producto de la lista, y si hacemos clic en editar nos mostrará el dialogo, para poder cambiar el nombre del producto.

7.7 Perfil

Como último fragmento tenemos el perfil de usuario, que podemos ver en la Fig. 7, donde podremos gestionar nuestra cuenta, cambiando el correo electrónico o la contraseña de nuestra cuenta.



Fig. 7: Gstión de usuario

Si hacemos clic en el botón de cambiar correo electrónico nos aparecerá un diálogo donde deberemos introducir el nuevo correo electrónico.

En cambio, si hacemos clic en el botón de cambiar contraseña aparecerá otro dialogo, donde deberemos introducir la nueva contraseña.

7.8 NavigationView

Para poder navegar por toda la aplicación se utiliza un *NavigationView* [4], que se puede ver en la Fig. 8.

El *NavigationView*, es simplemente un menú lateral de navegación que se hace visible desde el margen izquierdo de la aplicación. Este menú es accesible desde cualquier fragmento de la aplicación.

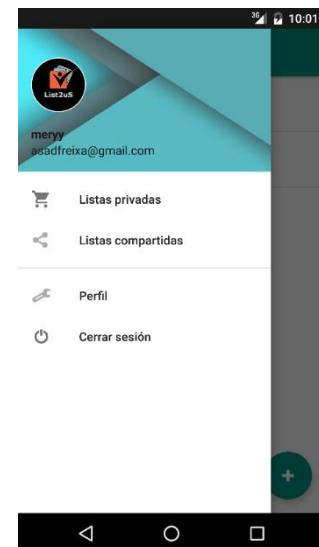


Fig. 8: NavigationView

En la parte superior podremos ver nuestro nombre de usuario y nuestro correo electrónico, a continuación el menú de navegación que nos permitirá movernos por toda la aplicación, entrar en nuestro perfil para gestionar nuestra cuenta o cerrar nuestra sesión.

7.9 Notificaciones

Para poder avisar al usuario de los diferentes cambios, novedades etc. Se envían notificaciones en forma de correo electrónico a los usuarios afectados por estas. En la Fig. 9 podemos ver un ejemplo de notificación cuando un usuario comparte una lista con otro usuario.



Fig. 9: Notificación por correo electrónico

Para poder implementar las notificaciones por correo electrónico se utiliza la API JavaEmail [5], que facilita el envío de correos electrónicos dentro de una aplicación sin que el usuario deba interactuar.

7.10 Acceso y tratamiento de los datos

Toda la información de la aplicación está ubicada en un servidor externo que contiene una base de datos MySQL con los datos de usuario, listas, productos etc.

Para acceder a dicha información desde la aplicación utilizamos la librería Volley [6] de Google, porque optimiza el envío de peticiones HTTP desde la aplicación hasta el servidor externo.

Primero de todo, creamos la petición desde la aplicación donde le pasaremos los parámetros necesarios para la misma, como ejemplo, para la petición de las listas de la compra de un usuario necesitaremos el correo electrónico.

Una vez tengamos la petición será enviada a la URL correspondiente, que contendrá el código PHP para procesarla, en este caso se cogerá el parámetro correo electrónico y se creará una petición a la base de datos para consultar todas las lista de la compra con ese mismo correo.

Seguidamente, PHP cogerá el resultado de la petición SQL y transformará los datos a formato JSON que serán retornados a la aplicación.

Si la petición ha salido correctamente, se procederá al parseo de la información recibida y se mostrará en la interfaz en este caso las listas de la compra del usuario.

8 PRUEBAS

Durante la realización de la aplicación se han ido haciendo pruebas para cada una de las funcionalidades que se implementaban.

Las pruebas se han realizado en 3 *smartphones* Android, un Motorola MotoG G 4G con Android 5.1 (Lollipop), un Sony Ericsson Xperia Neo V con Android 4.0.3 (Ice Cream Sand-

wich) y un Samsung Galaxy S3 mini con Android 4.1.2 (Jelly Bean).

A continuación se muestran algunas pruebas:

Registro de usuario

Entrada	Salida esperada	Salida
Registrarse	Registro de usuario en la base de datos	<input checked="" type="checkbox"/>
Registrarse sin indicar algún campo	No permite el registro	<input checked="" type="checkbox"/>
Registrarse con un correo electrónico sin el formato correcto	No permite el registro	<input checked="" type="checkbox"/>
Intento de registro de un usuario ya existente	No permite el registro	<input checked="" type="checkbox"/>
Intento de registro con contraseñas que no coinciden	No permite el registro	<input checked="" type="checkbox"/>

Tabla 7: Pruebas registro usuario

Inicio de sesión

Entrada	Salida esperada	Salida
Inició de sesión	Permite el inicio de sesión en la aplicación	<input checked="" type="checkbox"/>
Intento de inicio de sesión sin indicar algún campo	No permite el inicio de sesión	<input checked="" type="checkbox"/>
Intento de inicio de sesión con usuario y contraseña que no coinciden	No permite el inicio de sesión	<input checked="" type="checkbox"/>
Intento de Inicio de sesión con un usuario y contraseña que no existen	No permite el inicio de sesión	<input checked="" type="checkbox"/>

Tabla 8: Pruebas inicio de sesión

Gestión de usuarios

Entrada	Salida esperada	Salida
Cambio de correo electrónico del usuario	Actualización de la base de datos y cierre de sesión	<input checked="" type="checkbox"/>
Cambio de contraseña del usuario	Actualización de la base de datos y cierre de sesión	<input checked="" type="checkbox"/>
Intento de Cambio de correo electrónico con un formato incorrecto	No permite el cambio de correo electrónico	<input checked="" type="checkbox"/>
Cierre de sesión	Cierre de sesión abriendo la actividad de inicio de sesión	<input checked="" type="checkbox"/>

Tabla 9: Pruebas gestión de usuarios

Listas

Entrada	Salida esperada	Salida
Añadir una lista nueva	Adición de la nueva lista a la base de datos	<input checked="" type="checkbox"/>
Editar nombre de una lista	Actualización de la base de datos con el nombre de la nueva lista	<input checked="" type="checkbox"/>
Eliminar una lista	Actualización de la base de datos y si es compartida enviar un correo electrónico al usuario afectado	<input checked="" type="checkbox"/>
Compartir una lista	Actualización de la base de datos y envío del correo electrónico al usuario afectado	<input checked="" type="checkbox"/>

Tabla 10: Pruebas listas

Productos

Entrada	Salida esperada	Salida
Añadir un producto nuevo (lista no compartida)	Adición del nuevo producto a la base de datos	<input checked="" type="checkbox"/>
Añadir un producto nuevo (lista compartida)	Adición del nuevo producto a la base de datos y envío del correo electrónico al usuario afectado	<input checked="" type="checkbox"/>
Editar un producto (lista no compartida)	Actualización de la base de datos con el nuevo nombre del producto	<input checked="" type="checkbox"/>
Eliminar un producto (lista no compartida)	Actualización de la base de datos	<input checked="" type="checkbox"/>
Eliminar un producto (lista compartida)	Actualización de la base de datos y envío del correo electrónico al usuario afectado	<input checked="" type="checkbox"/>
Realización de un producto (lista no compartida)	Actualización de la base de datos y cambio de estado en el check de la aplicación	<input checked="" type="checkbox"/>
Realización de un producto (lista compartida)	Actualización de la base de datos, cambio de estado en el check de la aplicación datos y envío del correo electrónico al usuario afectado	<input checked="" type="checkbox"/>

Tabla 11: Pruebas productos

9 CONCLUSIONES

En la realización de este proyecto se ha desarrollado una aplicación para sistemas Android, para la gestión de listas de la compra, donde se pueden crear, editar, eliminar y compartir con otros usuarios.

Además de haber aprendido a desarrollar un proyecto de principio a fin en solitario, pasando por todas sus fases, he adquirido nuevos conocimientos, o ampliado muchos de ellos sobre Android, Java, PHP y SQL.

9.1 Desviación del proyecto

Como este proyecto tiene una duración limitada, y al haber alargado la duración de las tareas siguientes:

- Notificaciones vía correo electrónico
- Visualización de las listas de la compra
- Compartición de las listas de la compra

Aproximadamente, se alargaron las tareas unas 3 semanas y tomé la decisión de no realizar el objetivo de los recordatorios, ya que al ser un objetivo secundario los objetivos principales no se ven afectados por esta decisión, y se ha añadido a las líneas futuras de la aplicación.

Todos los demás objetivos se han realizado como se planearon al inicio del proyecto.

10 LÍNIAS FUTURAS

Hay funcionalidades que tanto la tutora como yo hemos ido recopilando para poder implementar si disponía de tiempo, pero debido a que el tiempo es limitado no se han podido realizar. Todas esas funcionalidades son:

- Crear una nueva sección para recordatorios, donde el usuario pueda introducir un producto y cada cuanto tiempo quiere que la aplicación le avise para comprarlo.
- Crear una lista predefinida de productos que son habituales en la vida diaria, para facilitar el uso al usuario.
- Poder iniciar sesión con las diferentes redes sociales que existen, así como la compartición de las listas mediante ellas.
- Introducir también notificaciones push y que el usuario pueda desactivarlas si lo desea.
- Realización de una página web donde tenga información de la aplicación, puedes descargarla y puedas hacer lo mismo que en la aplicación.
- Posibilidad de realizar la aplicación para el sistema operativo iOS.
- Poder compartir una lista de la compra con varios usuarios.
- Implementar la posibilidad de introducir la cantidad, precio, categoría de los productos.
- Implementar la posibilidad de localización de los centros más cercanos para comprar un cierto producto.

AGRADECIMIENTOS

Me gustaría agradecer toda la ayuda y consejos de mi tutora para realizar este proyecto. También me gustaría agradecer todo el apoyo de mi familia y amigos.

BIBLIOGRAFIA

- [1] Instituto Nacional de Tecnologías de la Comunicación. (2009). Ingeniería del software: metodologías y ciclos de vida. Disponible en:
https://www.incibe.es/file/N85W1ZWPHifRgUc_oY8_Xg
[Fecha de consulta: 20/06/2016]
- [2] Referencia oficial de Google sobre fragments:
<http://developer.android.com/intl/es/guide/components/fragments.html>
[Fecha de consulta: 20/06/2016]
- [3] Referencia oficial de Google sobre FloatingActionButton:
<http://developer.android.com/intl/es/reference/android/support/design/widget/FloatingActionButton.html>
[Fecha de consulta: 20/06/2016]
- [4] Referencia oficial de Google sobre NavigationView:
<http://developer.android.com/intl/es/reference/android/support/design/widget/NavigationView.html>
[Fecha de consulta: 20/06/2016]
- [5] Explicación y página oficial de Oracle JavaEmail API:
<http://www.oracle.com/technetwork/java/javamail/index.html>
[Fecha de consulta: 20/06/2016]
- [6] Ejemplo y explicación de peticiones POST con la librería Volley:
<http://www.hermosaprogramacion.com/2015/02/android-volley-peticiones-http/>
[Fecha de consulta: 20/06/2016]
- [7] Prototipo interactivo inicial de la aplicación:
<https://popapp.in/w/projects/5704e66bfe3706e6222d3cab/preview/5704e9e34396b42e547cb86c>
[Fecha de consulta: 20/06/2016]