



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática
misCompras**



Presentado por Roberto Arasti Blanco
en Universidad de Burgos — 25 de junio
de 2021

Tutor: Carlos Pardo Aguilar



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. Carlos Pardo Aguilar, profesor del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Roberto Arasti Blanco, con DNI 71307060E, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado misCompras.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 25 de junio de 2021

Vº. Bº. del Tutor:

D. Carlos Pardo Aguilar

Resumen

“misCompras” es una aplicación móvil de seguimiento de gastos en compras que genera estadísticas a partir de los datos recogidos para ayudar al usuario a tomar decisiones.

Descriptores

aplicación móvil, compras, android, seguimiento de gastos

Abstract

“misCompras” is a shopping cost tracking mobile application that generates statistics from the retrieved data to help the user make decisions.

Keywords

mobile application, shopping, android, cost tracking

Índice general

Índice general	iii
Índice de figuras	iv
Introducción	1
Objetivos del proyecto	3
Técnicas y herramientas	5
Aspectos relevantes del desarrollo del proyecto	7
4.1. Ciclo de vida	7
4.2. Análisis	9
4.3. Diseño	9
4.4. Implementación	11
Trabajos relacionados	13
Conclusiones y Líneas de trabajo futuras	15
Bibliografía	17

Índice de figuras

4.1. Modelo en espiral[1]	8
---------------------------	---

Introducción

Este trabajo consiste en una aplicación móvil de seguimiento de compras. La memoria resume el desarrollo del proyecto y los anexos contienen información más concreta sobre este como la estructura del proyecto y su manual de uso.

Objetivos del proyecto

El objetivo principal de este trabajo es desarrollar una aplicación móvil que permita al usuario registrar lo que compra para mostrarle información al respecto y así ayudarle a tomar decisiones sobre cómo realizar sus compras. Como partes de este objetivo principal queremos:

- Poder introducir uno a uno los artículos comprados junto a su precio.
- Poder generar estadísticas sobre los gastos registrados.
- Poder generar un csv de los datos introducidos para sortear las limitaciones de las estadísticas generadas en la aplicación (e.g. Hacer gráficos o tablas dinámicas).
- Poder establecer un presupuesto mensual para uso en estadísticas y notificar cuando se sobrepasa.
- Una pantalla de inicio.
- Un menú para acceder al resto de funciones.

Los objetivos técnicos de este trabajo para conseguir los objetivos son:

- Tener una base de datos persistente en la que almacenar y acceder a los datos.
- Poder generar un archivo csv a partir de una búsqueda en la base de datos.
- Poder almacenar el presupuesto como una preferencia de la aplicación.

Técnicas y herramientas

El desarrollo de esta aplicación ha sido en espiral. La principal ventaja de este enfoque es el control del riesgo. Se ha comenzado con la especificación de requisitos y casos de uso aparte de la creación de un prototipo. Después se ha ido añadiendo mejoras al prototipo y requisitos nuevos.

Herramientas de desarrollo utilizadas:

- StarUML: he usado esta herramienta de modelado para hacer algunos de los diagramas.
- Android Studio: el entorno de desarrollo integrado oficial para desarrollo Android. Tiene un emulador incorporado pero también permite probar la aplicación desde un dispositivo android mediante una conexión cableada.
- Room: librería de persistencia para Android. Me ha facilitado el almacenamiento y el acceso a los datos en la aplicación.

Aspectos relevantes del desarrollo del proyecto

4.1. Ciclo de vida

El desarrollo del software ha seguido un modelo en espiral, se ha partido de unos requisitos base y un prototipo al que se han ido añadiendo funcionalidades entre reunión y reunión con el tutor. Hay cuatro fases que se repiten de forma cíclica^[7]:

- **Análisis de los objetivos:** en esta fase se eligen los requisitos a implementar en el ciclo actual. Esta fase incluye la planificación inicial únicamente en el primer ciclo.
- **Evaluación de riesgos:** se definen alternativas respecto a cómo abordar la implementación y se elige la menos arriesgada.
- **Desarrollo y pruebas:** se implementan los requisitos que se tienen como objetivos y se ejecutan pruebas sobre la implementación. No he realizado pruebas automatizadas porque el automatizar pruebas para una aplicación móvil es complicado porque tendría que instalar y aprender de cero a utilizar un framework de pruebas para interfaces móviles. En cambio, he definido una serie de pruebas manuales que verifican el correcto funcionamiento de la aplicación (ver sección “Pruebas del sistema” en el anexo C “Manual del programador”).
- **Planificación:** se muestra el producto al cliente (en este caso el tutor) y se decide el siguiente curso de acción.

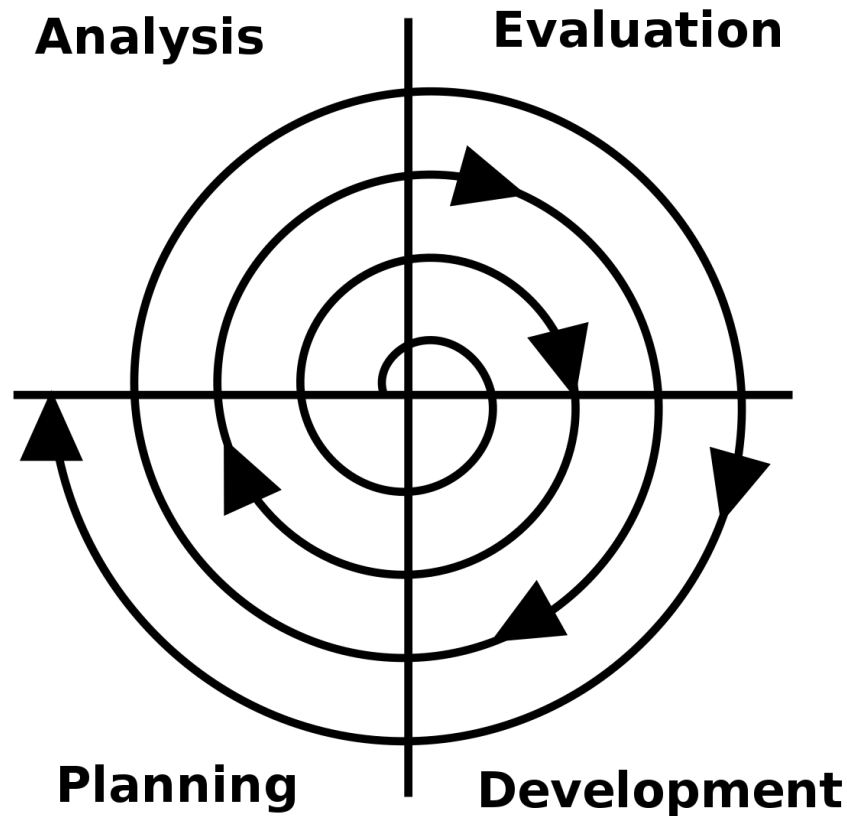


Figura 4.1: Modelo en espiral[1]

La fase de análisis está fuertemente influenciada por la fase de planificación del ciclo anterior y se definen casi por completo en cada reunión con el tutor así que la mayoría del tiempo del desarrollo nos encontramos en las fases de evaluación y desarrollo. También el hecho de que las reuniones marquen el principio y el fin de cada ciclo puede servir de referencia.

A juzgar por la cantidad de reuniones realizadas desde la creación del prototipo, he realizado cuatro ciclos completos. Mostré el primer prototipo a finales de abril y a partir de entonces empezaron las iteraciones.

Si hubiese sido capaz de reunirme con más frecuencia, el control del riesgo que proporciona el modelo en espiral hubiera sido más efectivo, por ejemplo, si en una reunión el cliente puede tirar por tierra todo el avance

realizado en el ciclo actual, reducir el tamaño de los ciclos aumentando la frecuencia de las reuniones minimiza esas pérdidas.

4.2. Análisis

En la especificación de requisitos base se recogieron solo los cuatro primeros objetivos: guardar uno a uno los artículos comprados, generar estadísticas, exportar a csv y marcar un presupuesto.

Después de la reunión posterior se añadieron los requisitos de las pantallas de inicio y de menú

Casi al final del desarrollo se añadieron los requisitos de poder borrar las filas de las diferentes tablas de la base de datos y mover productos en la jerarquía de categorías.

4.3. Diseño

Código

En Android Studio, las pantallas se definen mediante clases, así que para cada pantalla de la aplicación es necesario definir la clase correspondiente. Para la persistencia de datos, hay que definir las clases de los objetos persistentes que necesito, una clase abstracta de base de datos y una interfaz del objeto que se encarga de acceder a la base de datos de cuya implementación se encarga la librería. Por último, también decidí crear una clase adicional para facilitar el tratamiento de las filas de la tabla de gastos devueltas por la base de datos.

Base de datos

Para acelerar algunas acciones del usuario y poder generar estadísticas de más formas, he decidido crear una jerarquía de categorías, subcategorías y productos y que cada entrada de coste contenga la id del producto, el precio, y la fecha generada a la hora de crear la entrada.

Las tablas tienen definidas que al borrar uno de los elementos, se borren también sus hijos y a la hora de introducir dos veces el mismo objeto se reemplaza el viejo.

Disposición de elementos en pantalla

Aparte del código, cada clase correspondiente a una pantalla tiene asociada un archivo que indica qué elementos aparecen en pantalla y que características tienen como por ejemplo restricciones de distancia a los bordes o el ancho.

La pantalla de inicio contiene el nombre de la aplicación y dos botones, el botón menú lleva a la pantalla de menú y el botón demo datos carga en la base de datos un conjunto muy básico de datos utilizado para demostración en las reuniones y para probar yo mismo que las funcionalidades que voy implementando no estropeen otras partes de la aplicación.

La pantalla de menú contiene las opciones de guardar datos, generar estadísticas, exportar datos, marcar presupuesto, borrar elementos y mover elementos. Cada opción lleva a la pantalla correspondiente, pero la opción de generar estadísticas comprueba antes si hay datos sobre los que generar estadísticas y si no los hay lanza un mensaje.

La pantalla de Guardar datos tiene 3 menús desplegables utilizados para navegar con rapidez por la jerarquía de categorías, subcategorías y productos. Las ruedas se usan para introducir el precio del artículo y al seleccionar el producto toman automáticamente el valor del último gasto que hubo con ese producto, esto reduce la necesidad de tocar mucho las ruedas en caso de precios constantes o con fluctuaciones pequeñas (optimizar las operaciones[8]). Pulsar el botón aceptar guarda el gasto de ese producto con ese precio en la fecha actual. Los botones con el símbolo + al lado de cada menú desplegable llevan a las pantallas de Añadir categoría, Añadir subcategoría, y Añadir producto respectivamente y usan las elecciones marcadas en los menús desplegables para reducir aún más la cantidad de pulsaciones necesaria. El botón de borrar entradas al lado de aceptar lleva a la pantalla de Borrar entradas, he decidido ponerlo aquí porque facilita al usuario corregir un error que pueda haber cometido.

Las pantallas de Añadir categoría, Añadir subcategoría y Añadir producto tienen un espacio para escribir y un botón aceptar que introduce el elemento en caso de que no cause conflictos. Si hay un conflicto aparece una alerta avisando del problema, en caso de que no sea un conflicto grave que podría ser un olvido o intencional (e.g. Raquetas en Deporte, Tennis y en Alimentación, Repostería) se pregunta en la alerta si el usuario está seguro de realizar la acción.

La pantalla de borrar entradas muestra un menú desplegable con todas las entradas realizadas y un botón borrar.

La pantalla de generar estadísticas muestra otra vez tres menús desplegables pero a diferencia de los anteriores estos permiten dejar vacíos el segundo y tercer menú para seleccionar que se quieren generar estadísticas sobre una categoría, una subcategoría, o un producto. El botón generar estadísticas genera 3 estadísticas para un producto: su precio medio, el gasto en ese producto durante el mes actual y el porcentaje que supone del total gastado ese mes. En caso de hacer lo anterior con una categoría o subcategoría, se generan solo las dos últimas estadísticas. El botón estadísticas generales muestra el total de gasto en el mes actual, el gasto mensual medio, y el porcentaje utilizado del presupuesto marcado.

La pantalla de exportar datos tiene un botón que genera un archivo en formato csv con todos los costes y muestra por pantalla la ubicación del archivo.

La pantalla de marcar presupuestos utiliza 5 ruedas y un botón marcar presupuesto para introducir el presupuesto (por defecto está puesto en 500€).

La pantalla de borrar elementos muestra tres menús desplegables que funcionan igual que los de generar estadísticas y un botón borrar elemento que borra el elemento seleccionado después de avisar al usuario que se borrarán todos los elementos dependientes en la jerarquía.

4.4. Implementación

En el prototipo, solo implementé las pantallas de inicio, menú y guardar datos sin los botones adicionales (para añadir elementos y borrar entradas), después de enseñar el prototipo en la reunión ví que era recomendable añadir el botón de demo datos de la pantalla de inicio y que debía cambiar la base de datos porque la tenía mal, lo que nos lleva al siguiente problema, al cambiar la base de datos es necesario indicar el cambio incrementando el parámetro de versión en la anotación de la clase de base de datos y ofrecer una estrategia ante cómo transformar los datos de una versión con la vieja base de datos a la nueva, por suerte, a la hora de declarar la base de datos en la pantalla de inicio se puede añadir un método para indicar que la estrategia es deshacerse de los datos antiguos; esto no es un problema porque nunca he publicado la aplicación con la versión 1 de la base de datos.

La siguiente funcionalidad que implementé fue la de generar estadísticas, una incidencia que ocurrió al implementarla fue que la forma de obtener la fecha y hora que uso se añadió con Android 8, es decir, que la versión mínima de Android en la que funciona mi aplicación es Android 8 (Oreo).

Implementé Añadir categorías, Añadir subcategorías y Añadir productos.

Después implementé el exportar a un archivo csv pero con algunos fallos.

Implementé Marcar presupuesto guardando el presupuesto establecido en un archivo de preferencias compartidas.

Antes de implementar Borrar entradas, sabía que tenía que arreglar la forma en la que generaba el archivo exportado.

Implementé borrar elementos.

Añadí archivos de disposición de pantalla para poder girar el móvil y que los elementos no se saliesen de la pantalla.

Añadí Mover elementos.

Hice algunos cambios para hacer la aplicación tolerante a fallos (en general, evitar la acción con datos incorrectos y avisar al usuario).

Por último, añadí en la pantalla de estadísticas, un botón para generar un gráfico interactivo de sectores.

Trabajos relacionados

He buscado en la tienda de aplicaciones Google Play aplicaciones similares a la mía y he encontrado estas entre muchísimas otras:

PriceSpy[2]: permite comparar los precios de un producto en distintas tiendas. También permite comparar especificaciones de cada producto. Ayuda a tomar decisiones de otra índole sobre la compra.

Okticket[3]: permite almacenar tickets de compra a partir de una foto y puede realizar informes de gastos.

Control de gastos diarios y mensuales[4]: lleva control del saldo de tus cuentas y puede generar gráficos de sectores de ingresos o de gastos.

Monefy[5]: lleva control de gastos e ingresos pero solo genera el gráfico para gastos.

Cost Track[6]: otra aplicación de seguimiento de ingresos y costes.

Solo con entrar en la tienda de aplicaciones y buscar seguimiento de costes se puede ver que este concepto de aplicación es muy frecuente.

Conclusiones y Líneas de trabajo futuras

Estoy satisfecho con el resultado de este trabajo al conseguir un producto acabado, aun así, queda espacio de mejora.

Mis errores propios en distintos momentos me han hecho ver que todas las fases de desarrollo de software son cruciales para obtener un producto de calidad.

En la asignatura de 2º de Interacción Hombre-Máquina utilizamos Android Studio para algunas de las prácticas y al ver los resultados que obteníamos supuse que era una herramienta muy limitada, pero el realizar este proyecto me ha hecho darme cuenta de su potencial.

He visto que el trabajo se me ha vuelto cuesta arriba comparado con otros trabajos de asignaturas que he cursado durante la realización de este y que el trabajo en equipo marca la diferencia tanto en motivación como en calidad del producto final.

Futuras líneas de trabajo

Estas son tareas que considero realizaría si algún día continúo con este proyecto:

- Añadir una jerarquía de productos básica para que el cliente no tenga que introducir cada producto nuevo que compre.
- Añadir un icono de la aplicación para reemplazar el que Android Studio pone por defecto.

- Permitir acceder desde la pantalla de inicio al manual de usuario.

Bibliografía

- [1] Wikipedia. Software_development_spiral.svg, 2009. [Internet; descargado 23-junio-2021]. https://es.m.wikipedia.org/wiki/Archivo:Software_Development_Spiral.svg
- [2] Prisjakt Sverige AB. PriceSpy - Free price comparison, 2013. [App móvil]. <https://play.google.com/store/apps/details?id=se.prisjakt.pricespy>
- [3] okticket. Okticket - Gestión de gastos profesionales, 2018. [App móvil]. <https://play.google.com/store/apps/details?id=es.okticket.app>
- [4] SimpleProjectDevs. Control de gastos diarios y mensuales - Gratis, 2020. [App móvil]. https://play.google.com/store/apps/details?id=com.simpleprojectdevs.simple_savings
- [5] Aimbity. Monefy - App de control de gastos e ingresos, 2014. [App móvil]. <https://play.google.com/store/apps/details?id=com.monefy.app.light>
- [6] Artezio. Cost Track - Expense Tracker & Budget Manager, 2018. [App móvil]. <https://play.google.com/store/apps/details?id=com.artezio.costtrack.android>
- [7] Ian Sommerville. Ingeniería del software séptima edición, 2005. [pg. 68].
- [8] Wilfred J. Hansen. User engineering principles for interactive systems, 1971. [Fall Joint Computer Conference, AFIPS Press].