Anexo I – Plan del Proyecto Software

**Lista de cambios**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Descripción** | **Autor** |
| 1.1 | 11/10/2011 | Esqueleto inicial | Romina Liuzzi |
| 1.2 | 11/05/2012 | Primera iteración con estimaciones | Romina Liuzzi |
| 1.3 | 24/06/2012 | Estimaciones finales y conclusiones | Romina Liuzzi |

Tabla de contenidos

[Introducción 7](#_Toc328348519)

[Estimación de esfuerzo con casos de uso 8](#_Toc328348520)

[Factor de peso de los actores sin ajustar 9](#_Toc328348521)

[Factor de peso de los caso de uso sin ajustar 10](#_Toc328348522)

[Casos de uso sin ajustar 11](#_Toc328348523)

[Factores de complejidad técnica 12](#_Toc328348524)

[Factores ambientales 15](#_Toc328348525)

[Cálculo de puntos de casos de uso ajustados 17](#_Toc328348526)

[Esfuerzo en horas-persona 17](#_Toc328348527)

[Definición de tareas 18](#_Toc328348528)

[Investigación previa sobre plataforma y conceptos teóricos 18](#_Toc328348529)

[Definición del proyecto 18](#_Toc328348530)

[Análisis 19](#_Toc328348531)

[Diseño del sistema 19](#_Toc328348532)

[Desarrollo del sistema 21](#_Toc328348533)

[Pruebas 21](#_Toc328348534)

[Manual de usuario 22](#_Toc328348535)

[Firma del .apk y publicación de la aplicación 22](#_Toc328348536)

[Planificación temporal del proyecto 23](#_Toc328348537)

[Tabla de tareas inicial 23](#_Toc328348538)

[Diagrama de Gantt planificación Inicial 25](#_Toc328348539)

[Tabla de tareas Final 26](#_Toc328348540)

[Diagrama de Gantt planificación Final 27](#_Toc328348541)

[Desviaciones 28](#_Toc328348542)

[Estudio de viabilidad del proyecto 33](#_Toc328348543)

[Análisis de costes 33](#_Toc328348544)

[Recursos humanos 33](#_Toc328348545)

[Material software y hardware y gastos generales 35](#_Toc328348546)

[Coste total del proyecto 37](#_Toc328348547)

[Análisis de beneficios 38](#_Toc328348548)

**Índice de figuras**

[Figura 1. Horario laboral definido 24](#_Toc328351355)

[Figura 2. Estimación inicial. Diagrama de Gantt 25](#_Toc328351356)

[Figura 3. Estimación real. Diagrama de Gantt 27](#_Toc328351357)

[Figura 4. gitHub. Gráfico de frecuencia de código 29](#_Toc328351358)

[Figura 5. gitHub. Gráfico commits por semana 30](#_Toc328351359)

[Figura 6. gitHub gráfico de tarjeta perforada. 31](#_Toc328351360)

[Figura 7. gitHub. Listado de commits 32](#_Toc328351361)

[Figura 8. Descargas App 39](#_Toc328351362)

**Índice de tablas**

[Tabla 1. Peso relativo de los actores 9](#_Toc328351383)

[Tabla 2. Tabla de factores que influyen en los casos de uso 10](#_Toc328351384)

[Tabla 3. Factores de complejidad técnica 12](#_Toc328351385)

[Tabla 4. Evaluación de factores 13](#_Toc328351386)

[Tabla 5. Valores asignados por su peso 14](#_Toc328351387)

[Tabla 6. Factores ambientales 15](#_Toc328351388)

[Tabla 7. Valores Hora-Persona 17](#_Toc328351389)

[Tabla 8. Estimación Inicial 23](#_Toc328351390)

[Tabla 9. Estimación final 26](#_Toc328351391)

[Tabla 10. Gastos software 36](#_Toc328351392)

[Tabla 11. Gastos hardware 36](#_Toc328351393)

[Tabla 12. Coste total del proyecto 37](#_Toc328351394)

**Índice de fórmulas**

[Fórmula 1. UAW 9](#_Toc328351421)

[Fórmula 2. UUCW 10](#_Toc328351422)

[Fórmula 3. UUCP 11](#_Toc328351423)

[Fórmula 4. TCF 13](#_Toc328351424)

[Fórmula 5. EF 15](#_Toc328351425)

[Fórmula 6.UCP 17](#_Toc328351426)

# Introducción

Una de las fases más importantes en el desarrollo de cualquier proyecto es su planificación. Ésta nos permite hacer un seguimiento del mismo, ayudando en la toma de decisiones y en la gestión de los recursos asignados, tanto humanos como temporales.

En la planificación se establecen cuales son las tareas a realizar, así como el tiempo que se ha estimado que durará su desarrollo. También se determinará de qué recursos se dispone para la realización de estas tareas. Estas estimaciones se hacen dentro de un marco de tiempo limitado al comienzo de un proyecto de software, y deberán actualizarse regularmente a medida que progrese el proyecto. Al final del proyecto se analiza las desviaciones producidas y los motivos de éstas.

Fruto de esta planificación se obtiene el coste económico del proyecto desglosado en diversos conceptos: recursos humanos, material hardware y software y gastos generales.

# Estimación de esfuerzo con casos de uso

Para obtener una aproximación del número de horas requeridas en el desarrollo del sistema se utiliza un método de estimación creado por Gustav Karner (Rational Software Corporation) que permite realizar un estudio sobre los Casos de Uso del proyecto. Nos permitirá evaluar el tiempo que puede durar el proyecto en su etapa de desarrollo además del esfuerzo necesario.

Este proceso se lleva a cabo mediante una serie pasos empezando por obtener los casos de uso sin ajustar. Primeramente se toman en cuenta los actores (UAW) y los casos de uso (UUCW), y seguidamente se ajusta utilizando una serie de factores técnicos (TCF) y factores ambientales (EF). De esta forma se obtiene la complejidad del sistema, obteniendo una idea del número de horas-persona para el proyecto.

## Factor de peso de los actores sin ajustar

|  |  |  |
| --- | --- | --- |
| Tipo de Actor | Descripción | Peso |
| Simple | Otro sistema que interactúa con el sistema a desarrollar mediante un API. | 1 |
| Medio | Otro sistema interactuando a través de un protocolo ó una persona que interactúa con el sistema mediante una interfaz de texto. | 2 |
| Complejo | Una persona que interactua con el sistema mediante una interfaz gráfica. | 3 |

Tabla 1. Peso relativo de los actores

Se obtiene el factor de peso de los actores:

Fórmula 1. UAW

En nuestro caso obtenemos el factor de peso de los actores teniendo en cuenta que nuestro sistema cuenta con dos actores, un actor simple, que es la base de datos e interactúa con el sistema a partir de un API y otro actor complejo que corresponde a al perfil de usuario que utiliza el sistema. De esta forma obtenemos:

## Factor de peso de los caso de uso sin ajustar

Ahora se pasa a determinar el factor de complejidad en función del número de transacciones de cada caso de uso.

|  |  |  |
| --- | --- | --- |
| Tipo de caso de uso | Descripción | Factor |
| Simple | 3 transacciones o menos | 5 |
| Medio | 4 a 7 transacciones | 10 |
| Complejo | Más de 7 transacciones | 15 |

Tabla 2. Tabla de factores que influyen en los casos de uso

En nuestro caso tenemos en total de 17 casos de uso, categorizados:

* 3 medios
* 14 simples

Fórmula 2. UUCW

Con nuestros datos obtenemos:

## Casos de uso sin ajustar

Nos vale para conocer a grandes rasgos la complejidad de los casos de uso y de la interfaces tomando en cuenta los actores y los casos de uso.

Fórmula 3. UUCP

O lo que es lo mismo:

**PesoDeCasosDeUsoSinAjustar = PesoDeActores + PesoDeCasosDeUso**

En nuestro caso, con nuestros datos nos queda:

## Factores de complejidad técnica

Los factores técnicos cuantifican la dificultad del sistema en función de sus características internas, para esto nos valemos de 13 puntos:

|  |  |  |
| --- | --- | --- |
| Factor | Descripción | Peso |
| T1 | Sistema distribuido | 2 |
| T2 | Objetivos de performance o tiempo de respuesta | 1 |
| T3 | Eficiencia del usuario final | 1 |
| T4 | Procesamiento interno complejo | 1 |
| T5 | El código debe ser reutilizable | 1 |
| T6 | Facilidad de instalación | 0.5 |
| T7 | Facilidad de uso | 0.5 |
| T8 | Portabilidad | 2 |
| T9 | Facilidad de cambio | 1 |
| T10 | Concurrencia | 1 |
| T11 | Incluye objetivos especiales de seguridad | 1 |
| T12 | Provee acceso directo a terceras partes | 1 |
| T13 | Se requiere facilidades especiales de entrenamiento a usuario | 1 |

Tabla 3. Factores de complejidad técnica

Evaluados con una escala:

|  |  |
| --- | --- |
| Descripción | Valor |
| Irrelevante | De 0 a 2 |
| Medio | De 3 a 4 |
| Esencial | 5 |

Tabla 4. Evaluación de factores

Aplicando la fórmula obtenemos el factor de complejidad técnica:

Fórmula 4. TCF

Con nuestros datos obtenemos al aplicar los valores de la siguiente tabla, un total de

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Factor | Peso | Valor | Producto |  |
| T1 | 2 | 0 | 0 |  |
| T2 | 1 | 1 | 1 |  |
| T3 | 1 | 0 | 0 |  |
| T4 | 1 | 1 | 1 |  |
| T5 | 1 | 2 | 2 |  |
| T6 | 0.5 | 1 | 0.5 |  |
| T7 | 0.5 | 1 | 0.5 |  |
| T8 | 2 | 0 | 0 |  |
| T9 | 1 | 1 | 1 |  |
| T10 | 1 | 0 | 0 |  |
| T11 | 1 | 1 | 1 |  |
| T12 | 1 | 0 | 0 |  |
| T13 | 1 | 0 | 0 |  |

Tabla 5. Valores asignados por su peso

## Factores ambientales

Los factores sobre los cuales se realiza la evaluación constan de 8 puntos calificables con unos valores que van de 0 a 5.

|  |  |  |
| --- | --- | --- |
| Factor | Descripción | Peso |
| E1 | Familiaridad con el modelo de proyecto usado | 1.5 |
| E2 | Experiencia en la aplicación | 0.5 |
| E3 | Experiencia en orientación a objetos | 1 |
| E4 | Capacidad del analista líder | 0.5 |
| E5 | Motivación | 1 |
| E6 | Estabilidad de los requerimientos | 2 |
| E7 | Personal a tiempo parcial | -1 |
| E8 | Dificultad del lenguaje de programación | -1 |

Tabla 6. Factores ambientales

La fórmula a aplicar en este caso:

Fórmula 5. EF

De la aplicación a nuestro caso práctico resulta la siguiente tabla:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Factor | Descripción | Peso | Calificación | Producto |
| E1 | Familiaridad con el modelo de proyecto usado | 1.5 | 3 | 4.5 |
| E2 | Experiencia en la aplicación | 0.5 | 4 | 2 |
| E3 | Experiencia en orientación a objetos | 1 | 3 | 3 |
| E4 | Capacidad del analista líder | 0.5 | 3 | 1.5 |
| E5 | Motivación | 1 | 5 | 5 |
| E6 | Estabilidad de los requerimientos | 2 | 4 | 8 |
| E7 | Personal a tiempo parcial | -1 | 0 | 0 |
| E8 | Dificultad del lenguaje de programación | -1 | 1 | -1 |

Tras la aplicación de la fórmula anterior, obtenemos un valor de **EF = 0.71**.

## Cálculo de puntos de casos de uso ajustados

Se obtiene al aplicar la fórmula utilizando todos los coeficientes calculados hasta aquí.

Fórmula 6.UCP

En nuestro caso:

## Esfuerzo en horas-persona

Para obtener esta medida es preciso contar el número de factores ambientales del E1 al E6 que cuentan con una puntuación menor que 3, y el número de factores ambientales cuya puntación supere 3. En nuestro caso obtenemos 0.

Aplicando el criterio:

|  |  |
| --- | --- |
| Horas-Persona (CF) | Descripción |
| 20 | Valor menor o igual a 2 |
| 28 | Valor menor o igual a 4 |
| 36 | Valor mayor de 5 |

Tabla 7. Valores Hora-Persona

Obtenemos un coeficiente de 20.

Con lo cual nuestra estimación de horas asciende a: 990 horas/persona

# Definición de tareas

El desarrollo de este proyecto será dividido en unas fases que se pasan a enumerar y resumir a continuación.

## Investigación previa sobre plataforma y conceptos teóricos

La fase de investigación se estima al principio del proyecto ya que el desarrollo de la aplicación se realiza sobre una plataforma totalmente desconocida al estudiante. Es necesario fijar unos conceptos básicos antes de ser capaces de comenzar a desarrollar. El aprendizaje es iterativo y se entiende que estaremos formándonos durante todo el transcurso del proyecto, pero se reserva una bolsa de horas al comienzo para autoformación sobre la plataforma de desarrollo, IDE y todo el montaje previo necesario para llevar a cabo el desarrollo de la aplicación.

## Definición del proyecto

En esta fase se entiende, que tras la investigación estaremos en condiciones de presentar una primera versión del documento de especificación de requisitos con los casos de uso para su validación y aprobación.

También durante esta fase se resolverá el método de trabajo y se definirá el ritmo y sistema de entregas. Debido a que el proyecto es realizado a distancia, es importante ser riguroso con la metodología de trabajo. Es importante que esta fase se encuentre lo más avanzada posible, ya que se trata de una tarea bloqueante.

A grandes rasgos esta tarea está compuesta por las siguientes sub-tareas:

* Investigar y definir método de trabajo y herramientas (gestor de versiones, entregas periódicas, etc.)
* Configuración e integración de herramientas de trabajo
* Propuesta de proyecto
* Definición de requisitos

## Análisis

Durante esta fase se pretende tener una idea clara sobre las soluciones adoptadas para cumplir con los requisitos comprometidos en la fase anterior. En esta fase se evaluarán las posibles soluciones tecnológicas para cada uno de los problemas a resolver. Las fases de análisis, diseño y desarrollo están muy ligadas, y es posible que en algún caso se adopte un enfoque incremental, es decir abordar un problema completo, por ejemplo la especificación de un requisito o caso de uso, y realizar el análisis, diseño y desarrollo del mismo de forma paralela.

Algunas de las sub-tareas tratadas en esta fase:

* Análisis de requisitos
* Análisis de posibles soluciones
* Análisis de funcionalidades de la aplicación y permisos

## Diseño del sistema

La fase de diseño del sistema comprende tareas que van desde la clasificación en paquetes que compondrán la aplicación, el diseño del sistema de persistencia de datos, interfaces de usuario, comunicación entre vistas y controladores, etc. Es importante comprender que la realización del proyecto se lleva a cabo sobre un marco de tiempo limitado, y que tal limitación influirá directamente tanto sobre el resultado del análisis como las soluciones de diseño que resulten del mismo. Todo desarrollo es mejorable, el objetivo principal es contar con un diseño tan bueno como el tiempo permita.

Algunas de las sub-tareas asociadas a esta tarea son:

* Definición y diseño de pantallas
* Definición y diseño de entidades
* Definición de componentes que intervienen en la seguridad
* Definición de estructura de paquetes

## Desarrollo del sistema

Durante esta fase se lleva a cabo la implementación de la aplicación. Es aquí donde se aplican las soluciones diseñadas en la fase anterior. Desde estructura, comunicación entre capas, visibilidad entre clases, etc.

Algunas de las sub-tareas asociadas a esta fase son:

* Instaurar arquitectura de tres capas
* Implementar controladores para las vistas (pantallas)
* Depurar algoritmo de cálculo de calorías
* Implementar controladores de Google Maps
* Persistir datos
* Gestionar acceso a BBDD

## Pruebas

La consecución de esta fase vendrá fuertemente condicionada por la fase anterior. En esta fase entra la implementación de pruebas unitarias, funcionales, simulaciones, etc. Toda el testeo, detección de errores y resolución se estima dentro de esta fase.

En esta fase se encuentran las siguientes tareas:

* Estudio y configuración de herramientas (emuladores, etc.)
* Obtener rutas simuladas y depurar la aplicación
* Pruebas sobre terminal real

## Manual de usuario

Esta fase se realiza al finalizar la fase de pruebas y se documenta con capturas de pantallas de la aplicación terminada, por tanto es una de las últimas fases a realizar. También tiene un peso, desde mi punto de vista, menos prioritario que el resto.

## Firma del .apk y publicación de la aplicación

Es un objetivo del proyecto que la aplicación se encuentre publicada en la plataforma de distribución de apps de Google, Google Play. Para conseguir este propósito será preciso haber completado el desarrollo y ser capaces de asegurar una calidad aceptable a partir de pruebas unitarias, funcionales tanto simuladas sobre emuladores como sobre el terminal físico.

También será preciso llevar a cabo unas acciones requeridas por google para distribuir la aplicación en su Market. Estas acciones incluyen quitar los logs habilitados durante el proceso de desarrollo, testear la app, crear material promocianl, firmar la aplicación con un certificado personal correcto, etc. Todas estas tareas se estiman dentro de esta fase.

Finalmente en esta fase distinguimos las siguiente sub-tareas:

* Registro en Android developer console
* Preparación del material promocional (gráficos, videos, etc)
* Obtener e implantar certificado de desarrollo/release
* Publicar app en Google Play

# Planificación temporal del proyecto

## Tabla de tareas inicial

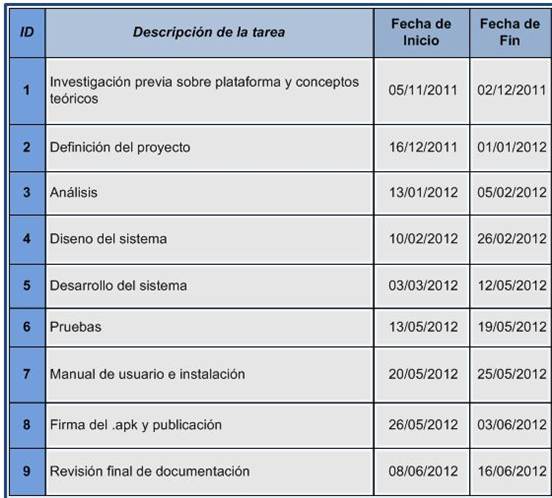


Tabla 8. Estimación Inicial

La estimación de la duración total del proyecto es de 990 horas, considerando una jornada laboral de 35 horas a la semana se obtiene una duración de 28 semanas, lo que traducido en meses resulta de aproximadamente 7 meses.

La fecha de inicio es del 5 de noviembre y se estima su finalización para el 16 de junio.

Como durante la realización del presente proyecto el alumno se encuentra trabajando, se estima el horario laboral tal como se enseña en la figura:



Figura 1. Horario laboral definido

Por otra parte los días laborales se estiman en función del tiempo real que es posible dedicar al proyecto, esto complica un poco las estimaciones ya que se trata de tardes, fines de semanas y días de vacaciones o festivos.

## Diagrama de Gantt planificación Inicial

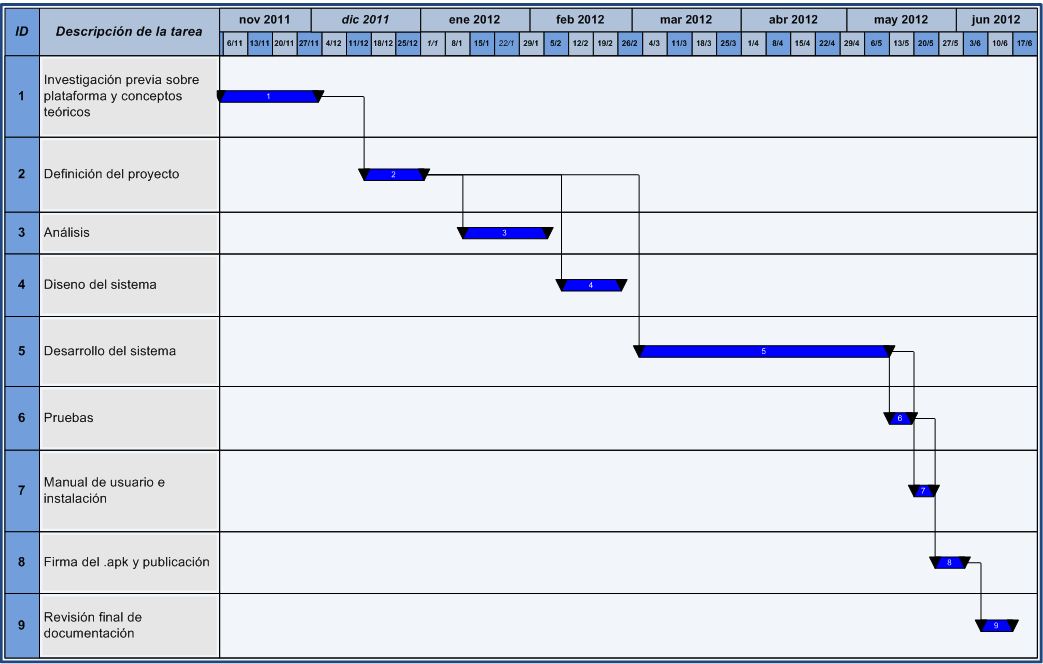


Figura 2. Estimación inicial. Diagrama de Gantt

## Tabla de tareas Final

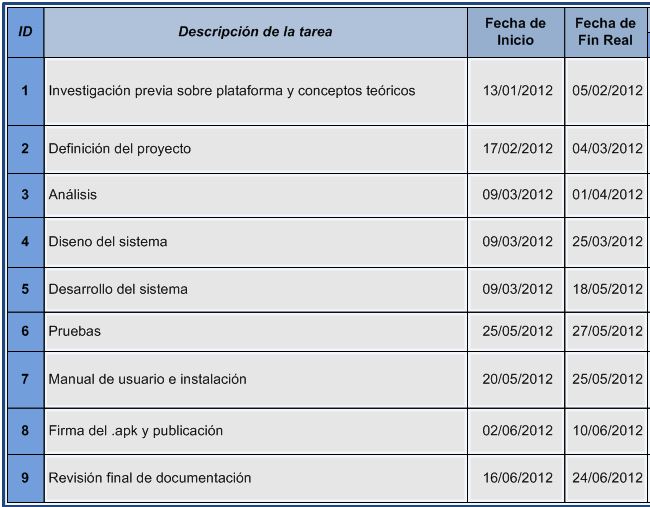


Tabla 9. Estimación final

## Diagrama de Gantt planificación Final

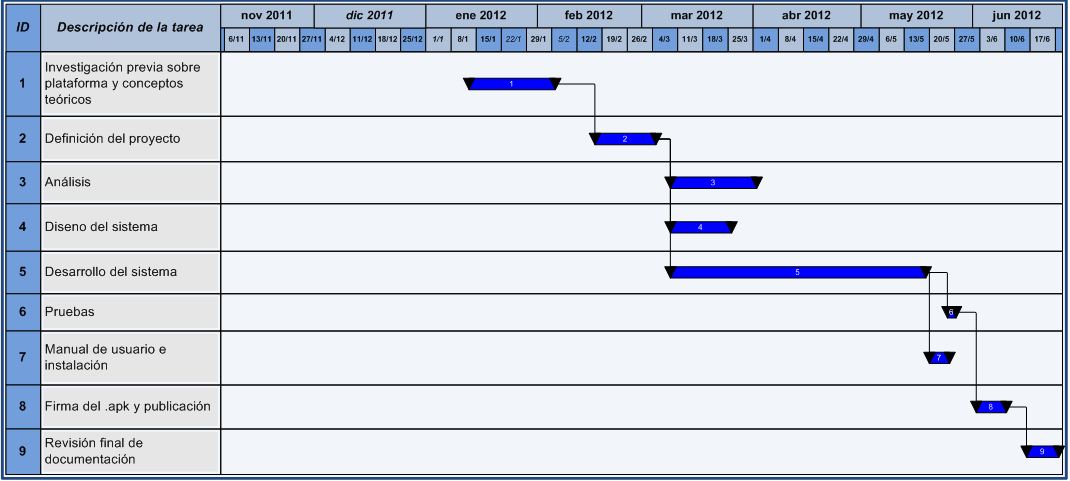


Figura 3. Estimación real. Diagrama de Gantt

## Desviaciones

La realización del proyecto se vio fuertemente condicionada por el calendario laboral, fue posible dedicar más horas en épocas de trabajo más liviano y fue casi imposible sacar tiempo durante los meses de noviembre y diciembre cuando coincidió el cierre de proyectos con fechas y presiones externas muy fuertes.

Se intentó que el diagrama de Gantt refleje esta situación, donde da la impresión de que las tareas se están paralelizando a pesar de contar con un único recurso para llevarlas a cabo. Esta situación se puede entender si tenemos en cuenta que el número de horas dedicadas al proyecto aumento durante este periodo, permitiendo avanzar con mayor velocidad. También es procedente tener en cuenta que estas tres fases en muchos casos se realizaron de forma conjunta.

Por otra parte el número real de horas dedicadas no difiere mucho del estimado. Dado que durante el transcurso del proyecto se utilizó un gestor de versiones, es posible, observar esta situación a partir de las estadísticas generadas.

El siguiente gráfico demuestra que la mayoría de los cambios a nivel código (corresponde con la fase de implementación y pruebas) se llevó a cabo durante los meses de marzo y abril. El gráfico toma lugar entre el 11 de marzo y el 10 de junio tal como enseña el eje horizontal temporal. La gráfica original se encuentra disponible en gitHub.com con mejor resolución.

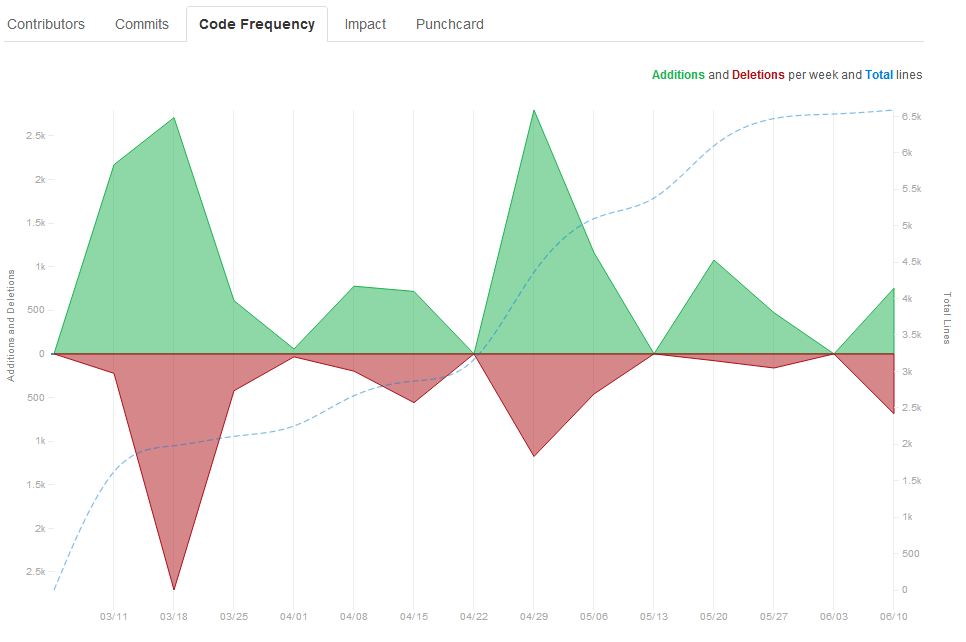


Figura 4. gitHub. Gráfico de frecuencia de código

Quizás el gráfico de commits semanales resulta más visual en este aspecto. Donde se selecciona una semana del año del gráfico de barras superior y se actualiza el gráfico inferior pintando el número de commits realizados por día de la semana.

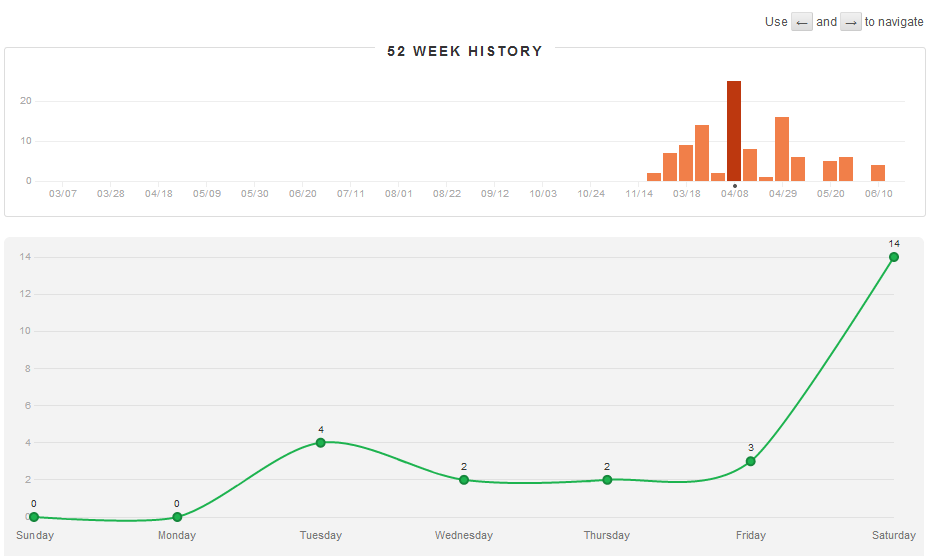


Figura . gitHub. Gráfico commits por semana

Por otra parte, en el siguiente gráfico, autogenerado a partir de los commits y extraído del mismo gestor de versiones se puede observar que también se dedicaban horas fuera de los días inicialmente planificados (tardes de los viernes y fines de semana).

Esta gráfica la denominan “Punchcard” del inglés tarjeta perforada. Hace referencia al antiguo sistema de fichaje de los trabajadores. El eje lateral representa el día de la semana y el eje horizontal la hora del día en la que se realizó la actividad. Adicionalmente el tamaño del punto representa el número de commits realizados en ese par hora/día.

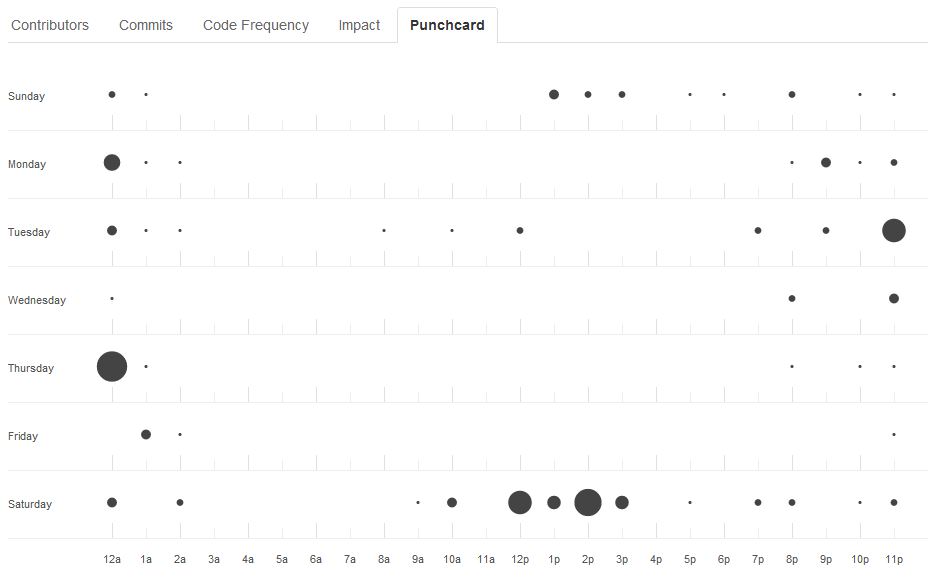


Figura 6. gitHub gráfico de tarjeta perforada.

Finalmente cada commit queda asociado a una lista de cambios identificada con un nombre, por lo que cada cambio queda registrado de forma permanente en este sistema. Tal como enseña la lista de cambios, la primera fase de la implementación consistió en la reestructuración del código heredado de la primera versión del sistema.

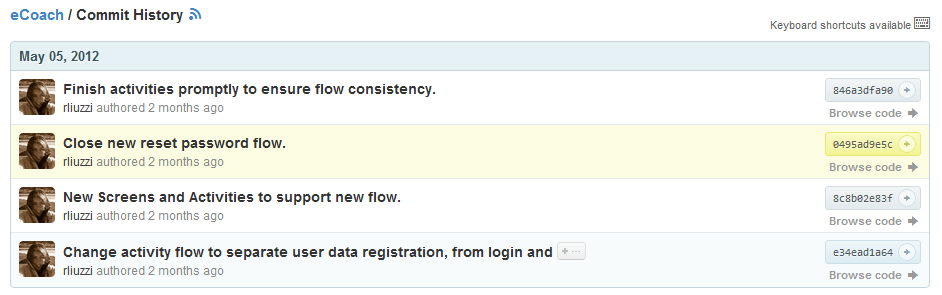


Figura . gitHub. Listado de commits

Se dedicó una fuerte carga de trabajo a la reestructuración aplicando arquitectura de tres capas al sistema, corrección de errores, re-implementación de funcionalidades enteras, adición de nuevas funcionalidades, etc. Esto se corresponde con la estimación de que la mayor carga de horas recaería sobre las fases de Implementación y Pruebas.

# Estudio de viabilidad del proyecto

En el estudio económico debemos tener en cuenta los distintos recursos utilizados. Por un lado debemos cuantificar los costes de personal, teniendo en cuenta los tiempos estimados de realización del proyecto. Por otro lado deberemos tener en cuenta los costes del material software y hardware utilizados.

A continuación se muestran los costes del proyecto desglosados según su categoría. Finalmente se añaden los gastos generales y se establece el coste total del proyecto.

## Análisis de costes

### Recursos humanos

**Salario**

El equipo consta de una única persona que se supone, tiene una experiencia laboral básica. En este caso se establecerá un sueldo de 10 € la hora.

El tipo de horario sería de más de 30 horas semanales por lo que habría que aplicar un cómputo de jornada completa. El salario mínimo interprofesional facilitado por la seguridad social es de 624 € mensuales. En este caso, aplicaremos un sueldo de 1600€ mensuales que equivaldrían a 10€/Hora.

**Seguridad Social**

La ley de Seguridad Social marca un porcentaje de sueldos que deben ser abonados al estado en concepto de impuestos.

Los empleados se encontrarían en el grupo 2: Ingenieros Técnicos, Peritos y Ayudantes Titulados cuyas bases de cotización en contingencias comunes oscilarían entre: 843,30 - 3.166,20 Euros/mes

El sueldo estimado asciende a 1600€/mes que entra dentro de la base de cotización establecida. Por tanto habría que aplicar un 23,60% de retención para la seguridad social.

* Retención Mensual S.S. = 1600 x 23,60% = 377,6 euros mensuales
* Gastos de personal mes = Sueldo Bruto + Seguridad Social
* Gasto mensual por persona = 1600 + 377,6 = 1977,6 €

El gasto total sería entonces los 7 meses estimados de duración del proyecto por el gasto mensual de los empleados:

**Gasto personal estimado: 1977,6 x 7 = 13.843,20 €**

### Material software y hardware y gastos generales

El conjunto de herramientas software utilizadas incluye una amplia gama de herramientas de libre distribución, que a efectos de estos cálculos, no serán tenidas en cuenta.

Para la realización del proyecto también se han utilizado herramientas no gratuitas. Estas herramientas tienen un coste determinado, pero este coste no será el que tenemos que imputar al proyecto. Lo que haremos será calcular el coste de cada herramienta, en función del tiempo de duración del proyecto y de su coste total.

Supondremos que todo el material software y hardware tiene una vida útil de 2 años, que es el tiempo que se tarda aproximadamente en renovarlo, ya que para entonces ya se dispondrá de nuevas versiones y actualizaciones del producto.

Los dos años de vida útil corresponden a 278 días laborables por año (eliminando fines de semana, festivos y días vacacionales). La jornada laboral es de 8 horas diarias con lo cual este material se debe amortizar en:

**2 años \* 278 días laborables \* 8 horas/día = 4448 horas.**

Sin embargo la realización de este proyecto conlleva solo 990 horas. Por lo tanto, calcularemos el coste del material en función de las horas empleadas. Pongamos un ejemplo.

Una de las herramientas utilizadas es la suite Microsoft Office 2007, que tiene un coste de 502 €. Como la aplicación se debe amortizar en dos años (4448 horas) calculamos el precio por hora de la herramienta:

**502 € / 4448 horas = 0.11 €/hora**

Si lo multiplicamos por el tiempo total de desarrollo del proyecto (ya que es difícil estipular el tiempo real durante el que se ha utilizado la herramienta) obtenemos el predio que debemos imputarle al proyecto:

**0.11 € \* 990 horas = 108.90 €**

A continuación obtenemos el cálculo del coste total de los recursos software utilizado:

|  |  |  |
| --- | --- | --- |
|  | Coste del material | Coste del material en el proyecto |
| Windows 7 professional edition | 348 € | 69.64 € |
| MS Office 2007 | 502 € | 108.90 € |

Tabla . Gastos software

A continuación obtenemos el cálculo del coste total de los recursos hardware utilizado:

|  |  |  |
| --- | --- | --- |
|  | Coste del material | Coste del material en el proyecto |
| PC entorno de desarrollo y documentación | 490 € | 96.05 € |
| Terminal físico Android (HTC Hero) | 120 € | 26.70 € |

Tabla . Gastos hardware

**Gastos generales:**

A parte de los gastos ya establecidos se incurre en otro tipo de gastos de carácter general para llevar a cabo el desarrollo de este proyecto. Entre éstos se encuentra el material de oficina (CDs, impresiones, material de oficina, etc) y la licencia mensual del gestor de versiones (aproximadamente 5 € al mes). Dado que son gastos mínimos en relación al grueso del proyecto, no se ha llevado a cabo un control exhaustivo sobre ellos, sin embargo para no excluirlos del cómputo final se ha estimado un gasto de 250 € en este concepto.

### Coste total del proyecto

A partir del desglose de gastos recogidos en los apartados anteriores obtenemos el coste total del proyecto:

|  |  |
| --- | --- |
|  | Coste del material del proyecto |
| RRHH | 13.843,20 € |
| Material SW | 178,54 € |
| Material HW | 122,75 € |
| Gastos generales | 250,00 € |
| TOTAL | **14.394,49 €** |

Tabla . Coste total del proyecto

Se puede apreciar que los recursos humanos aglutinan la mayor parte del coste total.

## Análisis de beneficios

Ganar dinero con el proyecto es lo que hace factible para su realización. Nuestro proyecto está dentro del campo de las aplicaciones móviles, por lo que su precio de comercialización no debería ser demasiado excesivo. Rara vez el coste por descarga de una aplicación para móviles dentro del área del entretenimiento supera los 99 céntimos.

La aplicación fue lanzada al mercado unos días antes de la presentación del presente proyecto para ser descargada de forma gratuita. Una opción de percibir beneficios a partir de su distribución radica en distribuirla de forma gratuita y percibir un arancel a cambio de presentar banners de publicidad dentro de la misma aplicación. La versión sin banners puede ser descargada con un coste de distribución.

A partir de los datos recogidos durante estos días podemos realizar un cálculo del tiempo que supondría recuperar la inversión desembolsada para el desarrollo de este proyecto y comenzar a percibir beneficios.

Suponiendo que se mantenga la tendencia (aproximadamente 25 descargas por semana) y que se perciba 1 euro por cada descarga, teniendo en cuenta que el coste total del proyecto ascendió a un total de **14.394,49 €** tardaríamos un total de 11 años en recuperar la inversión. Si se completan los ingresos con los percibidos a partir de la publicidad y se tiene en cuenta que gran parte del código podrá reutilizarse en el desarrollo de más aplicaciones, entonces el tiempo se vería reducido.

En todo caso se trata de una aplicación de final de carrera cuyo objetivo principal radica en el aprendizaje, desde este punto de vista, la inversión resulta claramente rentable.



Figura . Descargas App