

ANEXO I: PLAN DEL PROYECTO

Habitr - Aplicación para fomentar los hábitos saludables.

Trabajo de Fin de Grado

Ingeniería Informática



VNiVERSIDAD D SALAMANCA

Junio 2023

Autor:

Jorge Cruz García

Tutores:

André Filipe Sales Mendes

Gabriel Villarubia González

ÍNDICE

Tabla de figuras	3
Índice de tablas.....	4
1 Introducción	6
2 Estimación de esfuerzo	7
2.1 Complejidad de actores	7
2.2 Complejidad de casos de uso	8
2.3 Factores	11
2.4 Factores de complejidad técnica	11
2.5 Factores de complejidad del entorno	13
3 Planificación Temporal.....	16
4 Referencias.....	22

TABLA DE FIGURAS

Figura 1: Fases del Proceso Unificado	6
Figura 6: Estimación EZEstimate	15
Figura 2: Tareas (parte 1)	17
Figura 3: Tareas (parte 2)	18
Figura 4: Tareas (parte 3)	19
Figura 5: Diagrama de Gantt	22

ÍNDICE DE TABLAS

Tabla 1: ACT-01 Usuario no logueado.....	7
Tabla 2: ACT-02 Usuario logueado.....	7
Tabla 3: ACT-03 Sistema	8
Tabla 4: UC-01 Registro	8
Tabla 5: UC-02 Login.....	8
Tabla 6: UC-03 Logout	8
Tabla 7: UC-04 Recuperar contraseña	8
Tabla 8: UC-05 Ver perfil.....	8
Tabla 9: UC-06 Crear publicación	8
Tabla 10: UC-07 Borrar publicación.....	9
Tabla 11: UC-08 Ver feed.....	9
Tabla 12: UC-09 Dar like	9
Tabla 13: UC-10 Compartir una publicación.....	9
Tabla 14: UC-11 Enviar solicitud de amistad.....	9
Tabla 15: UC-12 Compartir código de amistad.....	9
Tabla 16: UC-13 Consultar solicitudes de amistad	9
Tabla 17: UC-14 Responder solicitudes de amistad	9
Tabla 18: UC-15 Desbloquear logro	9
Tabla 19: UC-16 Ver logros.....	9
Tabla 20: UC-17 Ver estadísticas	10
Tabla 21: UC-18 Crear rutina.....	10
Tabla 22: UC-19 Editar rutina	10
Tabla 23: UC-20 Borrar rutina.....	10
Tabla 24: UC-21 Ver rutinas públicas.....	10
Tabla 25: UC-22 Añadir rutina pública.....	10
Tabla 26: UC-23 Completar rutina	10
Tabla 27: UC-24 Programar notificaciones.....	10
Tabla 28: UC-25 Ver juego	10
Tabla 29: UC-26 Cambiar flores.....	10
Tabla 30: T1 Sistemas Distribuidos	11
Tabla 31: T2 Rendimiento	11
Tabla 32: T3 Eficiencia del Usuario Final.....	11
Tabla 33: T4 Procesamiento Interno Complejo	11
Tabla 34: T5 Reusabilidad.....	11
Tabla 35: T6 Facilidad de instalación.....	11
Tabla 36: T7 Facilidad de uso	12
Tabla 37: T8 Portabilidad	12
Tabla 38: T9 Facilidad a cambio	12
Tabla 39: T10 Concurrencia.....	12
Tabla 40: T11 Características de seguridad	12
Tabla 41: T12 Acceso directo a terceras partes	12
Tabla 42: T13 Entrenamiento especial del usuario	12
Tabla 43: E1 Familiaridad con UML.....	13

Anexo I: Plan del Proyecto

Tabla 44: E2 Experiencia en la aplicación	13
Tabla 45: E3 Experiencia con orientación a objetos	13
Tabla 46: E4 Experiencia de los analistas	13
Tabla 47: E5 Motivación	13
Tabla 48: E6 Estabilidad de los requisitos	13
Tabla 49: E67Trabajadores a tiempo parcial	14
Tabla 50: E8 Dificultad del lenguaje de programación	14

1 INTRODUCCIÓN

En este *Anexo I – Plan del proyecto* se va a llevar a cabo el estudio y explicación de los aspectos relativos a la planificación temporal del proyecto.

Lo primero que se hará es una **definición de las tareas a desarrollar**, una estimación del tiempo y esfuerzo que llevarán dichas tareas, y después **se dividirán en bloques de distintas fases siguiendo el proceso unificado**.

Finalmente, **se generará un diagrama de Gantt** para ayudar a la visualización de la planificación temporal, el tiempo dedicado a cada tarea, y la visualización del **camino crítico** de la aplicación, que representa las dependencias entre tareas y los bloqueos que pueden surgir.

Como se mencionó anteriormente, la planificación temporal y el desarrollo del proyecto estarán basados en el **Proceso Unificado**, explicado en la asignatura Ingeniería del Software. El **Proceso Unificado** es un marco de desarrollo marcado por un **enfoque en los casos de uso**, y un proceso **iterativo e incremental de distintas fases**.

En la *Figura 1* podemos ver el solapamiento entre distintas partes del proceso unificado, su solapación y la diferencia de carga de las tareas entre varias fases.

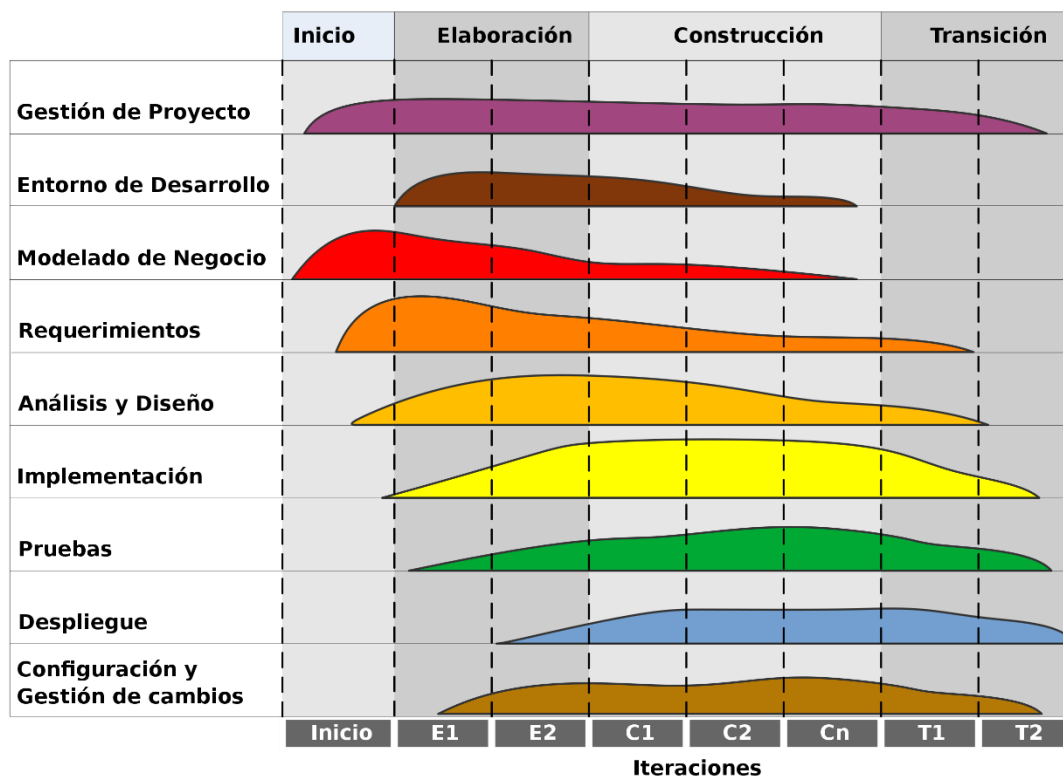


Figura 1: Fases del Proceso Unificado

2 ESTIMACIÓN DE ESFUERZO

Usando el modelo de **Puntos de Casos de Uso (UCP)** y la separación en casos de uso obtenido en el *Anexo II*, hemos realizado una estimación del esfuerzo que se tardaría en completar el proyecto.

La métrica UCP se basa en determinar un número de casos de uso sin ajustar (**UUCP**), que vienen dados por el número de actores y casos de uso de nuestro sistema y su complejidad, que dan paso a los factores de **UUCW (Unadjusted Use Case Weight)** y **UAW (Unadjusted Actor Weight)**

Una vez tenemos dichos UUCP, calculamos una serie de pesos que intentan definir la complejidad técnica (**TCF, Technical Complexity Factor**) y del entorno de nuestro sistema (**ECF, Environmental Complexity Factor**).

Teniendo estos tres datos, llegamos a la estimación del esfuerzo con las siguientes dos ecuaciones:

$$UCP = UUCP * TCF * ECF$$

$$Esfuerzo = UCP * F$$

donde F es un factor de conversión que define la cantidad de tiempo que se invertirá por unidad de UCP, y que en los proyectos suele situarse sobre 20 horas por UCP (pero cuyo valor es modificable según la previsión del analista).

2.1 COMPLEJIDAD DE ACTORES

La complejidad de los actores se determina observando la manera en la que interactúan con el sistema.

- **Simple:** Interactúan con el sistema a través de una API o funciones.
- **Medio:** Interactúan con el sistema a través de un protocolo.
- **Complejo:** Interactúan con el sistema a través de una interfaz gráfica

Con este conocimiento, la complejidad de nuestros actores es la siguiente:

ACT-01	Usuario no logueado
Complejidad	Complejo
Comentario	Interactúa a través de una interfaz gráfica.

Tabla 1: ACT-01 Usuario no logueado

ACT-02	Usuario logueado
Complejidad	Complejo
Comentario	Interactúa a través de una interfaz gráfica.

Tabla 2: ACT-02 Usuario logueado

ACT-03	Sistema
---------------	----------------

Complejidad	Simple
Comentario	Interactúa con el sistema a través de funciones.

Tabla 3: ACT-03 Sistema

2.2 COMPLEJIDAD DE CASOS DE USO

Para determinar la complejidad de los casos de uso, se contarán las transacciones que requieren los pasos de cada caso de uso.

Una transacción se define como un conjunto de operaciones atómicas, que o se ejecutan todas juntas, o no se ejecuta ninguna. No necesariamente van ligadas al número de pasos que tiene el caso de uso, si no a la complejidad de ellos.

Usando las transacciones como medida de complejidad obtendremos que los casos de uso pueden ser:

- **Simple: 3 o menos transacciones**
- **Medio: 4 a 7 transacciones**
- **Complejo: 7 o mas transacciones**

UC-01	Registro
Transacciones	3
Complejidad	Simple

Tabla 4: UC-01 Registro

UC-02	Login
Transacciones	3
Complejidad	Simple

Tabla 5: UC-02 Login

UC-03	Logout
Transacciones	1
Complejidad	Simple

Tabla 6: UC-03 Logout

UC-04	Recuperar contraseña
Transacciones	3
Complejidad	Simple

Tabla 7: UC-04 Recuperar contraseña

UC-05	Ver perfil
Transacciones	1
Complejidad	Simple

Tabla 8: UC-05 Ver perfil

UC-06	Crear publicación
Transacciones	3
Complejidad	Simple

Tabla 9: UC-06 Crear publicación

Anexo I: Plan del Proyecto

UC-07	Borrar publicación
Transacciones	1
Complejidad	Simple

Tabla 10: UC-07 Borrar publicación

UC-08	Ver feed
Transacciones	1
Complejidad	Simple

Tabla 11: UC-08 Ver feed

UC-09	Dar like
Transacciones	1
Complejidad	Simple

Tabla 12: UC-09 Dar like

UC-10	Compartir una publicación
Transacciones	1
Complejidad	Simple

Tabla 13: UC-10 Compartir una publicación

UC-11	Envíar solicitud de amistad
Transacciones	3
Complejidad	Simple

Tabla 14: UC-11 Envíar solicitud de amistad

UC-12	Compartir código de amistad
Transacciones	1
Complejidad	Simple

Tabla 15: UC-12 Compartir código de amistad

UC-13	Consultar solicitudes de amistad
Transacciones	1
Complejidad	Simple

Tabla 16: UC-13 Consultar solicitudes de amistad

UC-14	Responder solicitudes de amistad
Transacciones	1
Complejidad	Simple

Tabla 17: UC-14 Responder solicitudes de amistad

UC-15	Desbloquear logro
Transacciones	1
Complejidad	Simple

Tabla 18: UC-15 Desbloquear logro

UC-16	Ver logros
Transacciones	1
Complejidad	Simple

Tabla 19: UC-16 Ver logros

UC-17	Ver estadísticas
--------------	-------------------------

Anexo I: Plan del Proyecto

Transacciones	1
Complejidad	Simple

Tabla 20: UC-17 Ver estadísticas

UC-18	Crear rutina
Transacciones	3
Complejidad	Simple

Tabla 21: UC-18 Crear rutina

UC-19	Editar rutina
Transacciones	3
Complejidad	Simple

Tabla 22: UC-19 Editar rutina

UC-20	Borrar rutina
Transacciones	1
Complejidad	Simple

Tabla 23: UC-20 Borrar rutina

UC-21	Ver rutinas públicas
Transacciones	1
Complejidad	Simple

Tabla 24: UC-21 Ver rutinas públicas

UC-22	Añadir rutina pública
Transacciones	1
Complejidad	Simple

Tabla 25: UC-22 Añadir rutina pública

UC-23	Completar rutina
Transacciones	1
Complejidad	Simple

Tabla 26: UC-23 Completar rutina

UC-24	Programar notificaciones
Transacciones	1
Complejidad	Simple

Tabla 27: UC-24 Programar notificaciones

UC-25	Ver juego
Transacciones	1
Complejidad	Simple

Tabla 28: UC-25 Ver juego

UC-26	Cambiar flores
Transacciones	2
Complejidad	Simple

Tabla 29: UC-26 Cambiar flores

2.3 FACTORES

En este apartado se detallarán los valores de la **complejidad técnica (TCF)** y de la **complejidad del entorno (ECF)**, explicando los motivos de las elecciones que se han realizado.

2.4 FACTORES DE COMPLEJIDAD TÉCNICA

Los factores de complejidad técnica consisten en una serie de 13 parámetros cuyo valor varía entre 0 (factor irrelevante) hasta 5 (factor esencial):

T1	Sistemas Distribuidos
Valor	1
Comentario	El único aspecto distribuido de nuestra aplicación será la BBDD.

Tabla 30: T1 Sistemas Distribuidos

T2	Rendimiento
Valor	1
Comentario	Es una aplicación móvil, y por lo general no deberá ser difícil de optimizar.

Tabla 31: T2 Rendimiento

T3	Eficiencia del Usuario Final
Valor	3
Comentario	Se desea que la aplicación sea de fácil uso.

Tabla 32: T3 Eficiencia del Usuario Final

T4	Procesamiento Interno Complejo
Valor	0
Comentario	El sistema no tiene operaciones complejas, y mayormente usa operaciones CRUD.

Tabla 33: T4 Procesamiento Interno Complejo

T5	Reusabilidad
Valor	2
Comentario	La reutilización del código no es una prioridad, pero sin embargo si se intentará tener en cuenta a la hora del desarrollo.

Tabla 34: T5 Reusabilidad

T6	Facilidad de instalación
Valor	0
Comentario	Euna aplicación móvil, y por tanto su instalación no está en nuestras manos y será un simple click.

Tabla 35: T6 Facilidad de instalación

T7	Facilidad de uso
-----------	-------------------------

Anexo I: Plan del Proyecto

Valor	4
Comentario	La aplicación deberá de ser relativamente fácil de usar, para aumentar su uso entre toda la población.

Tabla 36: T7 Facilidad de uso

T8	Portabilidad
Valor	0
Comentario	El sistema está planteado como portable, sin embargo, la responsabilidad de la portabilidad no recae sobre el programador si no sobre el SDK de Flutter.

Tabla 37: T8 Portabilidad

T9	Facilidad a cambio
Valor	3
Comentario	Se pretende añadir nuevas funcionalidades y iterar rápidamente sobre el desarrollo.

Tabla 38: T9 Facilidad a cambio

T10	Concurrencia
Valor	1
Comentario	El sistema debe de poder soportar peticiones concurrentes.

Tabla 39: T10 Concurrencia

T11	Características de seguridad
Valor	3
Comentario	El sistema deberá almacenar la información y contraseñas de sus usuarios.

Tabla 40: T11 Características de seguridad

T12	Acceso directo a terceras partes
Valor	0
Comentario	El sistema sólo será usado por gente que tenga la aplicación instalada y tenga una cuenta. Por tanto, no hay terceras partes.

Tabla 41: T12 Acceso directo a terceras partes

T13	Entrenamiento especial del usuario
Valor	0
Comentario	No se realizará entrenamiento especial del usuario

Tabla 42: T13 Entrenamiento especial del usuario

2.5 FACTORES DE COMPLEJIDAD DEL ENTORNO

Los factores de complejidad del entorno consisten en una serie de 8 parámetros, cuyos valores varían entre 0 (sin experiencia / no es relevante) hasta 5 (experto / muy relevante):

E1	Familiaridad con UML
Valor	3
Comentario	Se ha estudiado UML a nivel teórico y se han realizado proyectos de prueba, sin embargo, nunca se ha utilizado de forma práctica.

Tabla 43: E1 Familiaridad con UML

E2	Experiencia en la aplicación
Valor	1
Comentario	Se tiene experiencia con UML, sin embargo no se tiene con Flutter.

Tabla 44: E2 Experiencia en la aplicación

E3	Experiencia con orientación a objetos
Valor	4
Comentario	Se han desarrollado múltiples proyectos con orientación a objetos a lo largo de los años.

Tabla 45: E3 Experiencia con orientación a objetos

E4	Experiencia de los analistas
Valor	3
Comentario	Los analistas tienen experiencia con UML y han realizado proyectos de prueba, pero nunca uno completo.

Tabla 46: E4 Experiencia de los analistas

E5	Motivación
Valor	5
Comentario	La motivación es muy alta dado que se escogió el tema del proyecto voluntariamente.

Tabla 47: E5 Motivación

E6	Estabilidad de los requisitos
Valor	4
Comentario	Los requisitos están bien definidos y no cambiarán mucho a lo largo del proyecto.

Tabla 48: E6 Estabilidad de los requisitos

E7	Trabajadores a tiempo parcial
Valor	0

Anexo I: Plan del Proyecto

Comentario	Se tiene un solo programador que tiene dedicación completa
-------------------	--

Tabla 49: E7 Trabajadores a tiempo parcial

E8	Dificultad del lenguaje de programación
Valor	0
Comentario	Dart es un lenguaje de alto nivel, con muchas librerías y mucho parecido a otros lenguajes de OOP como Java.

Tabla 50: E8 Dificultad del lenguaje de programación

Finalmente, hemos obtenido la siguiente estimación:

The screenshot shows the EZEstimate application window with the following data:

- Module:** Gestión autenticación
- Summary:**
 - Total Modules: 9
 - Use cases: Simple 26, Average 0, Complex 0
 - Actors: Simple 1, Average 0, Complex 2
- Estimation Summary:**
 - UAW: 7
 - UUCW: 130
 - UUCP = UAW + UUCW: 137
 - TFactor: 18
 - EFactor: 23
 - TCF = 0.6 + (.01*TFactor): 0,78
 - EF = 1.4 + (-0.03*EFactor): 0,71
 - UCP = UUCP*TCT*EF: 75,8706
 - Total Effort@ 15 Hrs/UCP: 1138,059
- Use case / Actor List:**

Id	Module	Type	Name	complexity
20	Gestión autenti...	Usecase	Login	Simple
29	Gestión autenti...	Actor	Usuario logueado	Complex
15	Gestión estadist...	Usecase	Ver estadísticas	Simple
22	Gestión juego	Usecase	Cambiar flores	Simple
10	Gestión logros	Usecase	Ver logros	Simple
11	Gestión rutinas	Usecase	Ver rutinas	Simple
12	Gestión rutinas	Usecase	Crear rutina	Simple
13	Gestión rutinas	Usecase	Editar rutina	Simple
14	Gestión rutinas	Usecase	Borrar rutina	Simple
16	Gestión rutinas	Usecase	Añadir rutina pú...	Simple
25	Gestión rutinas	Usecase	Programar notifi...	Simple
19	Gestión social	Usecase	Crear publicación	Simple
4	Gestión social	Usecase	Borrar publicaci...	Simple
5	Gestión social	Usecase	Ver feed	Simple
6	Gestión social	Usecase	Dar like	Simple
7	Gestión social	Usecase	Compartir public...	Simple
3	Gestión usuarios	Usecase	Ver perfil	Simple

Figura 2: Estimación EZEstimate

Se bajó el factor F a 15 horas por UCP, porque muchos de los casos de uso del proyecto son repetitivos y operaciones CRUD, cuya implementación es fácil.

Se obtuvo un resultado de 1138 horas de trabajo, o lo que es lo mismo, unos **140 días laborales**.

3 PLANIFICACIÓN TEMPORAL

La planificación temporal explica todas las tareas necesarias para el desarrollo del proyecto, definiendo una aproximación del tiempo que se empleará en cada una de ellas.

Seguiremos el Proceso Unificado cómo explicamos previamente, de manera que nuestra planificación se divide en las siguientes etapas principales:

- **Especificación de requisitos:** Se fijan los objetivos del proyecto y los requisitos que tiene que cumplir nuestra aplicación.
- **Análisis:** Se llevará a cabo el análisis detallado de los requisitos que hayamos establecido previamente.
- **Diseño:** Se explicará el funcionamiento determinado de los componentes del sistema.
- **Implementación:** Se programará un primer prototipo con código.
- **Pruebas:** Se realizarán pruebas unitarias y del sistema completo.

Cada una de estas etapas se dividirá a su vez en varias subetapas.

En las *figuras 2, 3 y 4*, que se presentan a continuación, podemos ver la división de tareas prevista para nuestro proyecto, junto al tiempo estimado.

Anexo I: Plan del Proyecto

→	▸ Habitr	116 days	Tue 10/01/23	Tue 20/06/23	
→	▸ Especificación de requisitos	12 days	Tue 10/01/23	Wed 25/01/23	
→	Fijar objetivos	3 days	Tue 10/01/23	Thu 12/01/23	
→	Definición de requisitos funcionales	5 days	Fri 13/01/23	Thu 19/01/23	3
→	Definición de requisitos no funcionales	3 days	Fri 13/01/23	Tue 17/01/23	3
→	Realización del documento "Anexo II: especificación de requisitos"	5 days	Wed 18/01/23	Wed 25/01/23	5
→	Hito: Fin de la especificación de requisitos	0 days	Wed 25/01/23	Wed 25/01/23	2
→	▸ Investigación	10 days	Tue 10/01/23	Mon 23/01/23	
→	Investigación de la literatura académica	10 days	Tue 10/01/23	Mon 23/01/23	
→	Comparación con aplicaciones parecidas	10 days	Tue 10/01/23	Mon 23/01/23	
→	Hito: Fin investigación	0 days	Mon 23/01/23	Mon 23/01/23	8
→	▸ Análisis	23 days	Thu 26/01/23	Mon 27/02/23	8;2
→	Modelo del dominio	5 days	Thu 26/01/23	Wed 01/02/23	
→	Realización de casos de uso - Análisis	6 days	Thu 02/02/23	Thu 09/02/23	13
→	Clases Análisis	3 days	Fri 10/02/23	Tue 14/02/23	14
→	Arquitectura del modelo de análisis	2 days	Wed 15/02/23	Thu 16/02/23	15
→	Realización del documento "Anexo III: Análisis de requisitos"	7 days	Fri 17/02/23	Mon 27/02/23	16
→	Hito: Fin del análisis de requisitos	0 days	Mon 27/02/23	Mon 27/02/23	12

Figura 3: Tareas (parte 1)

Anexo I: Plan del Proyecto

→	▸ Diseño	34 days	Tue 28/02/23	Fri 14/04/23	12
→	Clases de diseño	7 days	Tue 28/02/23	Wed 08/03/23	
→	Diseño de la base de datos	7 days	Thu 09/03/23	Fri 17/03/23	20
→	Arquitectura del modelo de diseño	5 days	Mon 20/03/23	Fri 24/03/23	21
→	Realización de casos de uso - Diseño	6 days	Mon 27/03/23	Mon 03/04/23	22
→	Modelo de despliegue	2 days	Tue 04/04/23	Wed 05/04/23	23
→	Realización del documento "Anexo IV: Diseño"	7 days	Thu 06/04/23	Fri 14/04/23	24
→	▸ Implementación	37 days	Mon 17/04/23	Tue 06/06/23	19
→	Diseño de la interfaz de la aplicación	7 days	Mon 17/04/23	Tue 25/04/23	19
→	Implementación de funcionalidades básicas	14 days	Mon 17/04/23	Thu 04/05/23	19
→	Implementación interfaz principal	7 days	Wed 26/04/23	Thu 04/05/23	27
→	▸ Autenticación	7 days	Fri 05/05/23	Mon 15/05/23	28;29
→	Implementación autenticación	4 days	Fri 05/05/23	Wed 10/05/23	28
→	Implementación interfaz autenticación	3 days	Thu 11/05/23	Mon 15/05/23	31
→	▸ Rutinas	6 days	Fri 05/05/23	Fri 12/05/23	28;29
→	Implementación de la interfaz de rutinas	2 days	Fri 05/05/23	Mon 08/05/23	29
→	Implementación de la gestión de rutinas	6 days	Fri 05/05/23	Fri 12/05/23	28
→	Implementación de la gestión de rutinas públicas	3 days	Fri 05/05/23	Tue 09/05/23	28

Figura 4: Tareas (parte 2)

Anexo I: Plan del Proyecto

→	▸ Usuario y amigos	8 days	Tue 16/05/23	Thu 25/05/23	30
→	Implementación de la gestión de usuarios	4 days	Tue 16/05/23	Fri 19/05/23	28
→	Implementación de la interfaz de usuarios	3 days	Tue 16/05/23	Thu 18/05/23	29
→	Implementación de la gestión de amigos	4 days	Mon 22/05/23	Thu 25/05/23	38;28
→	Implementación de la interfaz de amigos	3 days	Fri 19/05/23	Tue 23/05/23	39;29
→	▸ Social	6 days	Fri 26/05/23	Fri 02/06/23	37
→	Implementación gestión social	4 days	Fri 26/05/23	Wed 31/05/23	37
→	Implementación interfaz social	2 days	Thu 01/06/23	Fri 02/06/23	43
→	▸ Estadísticas	8 days	Fri 26/05/23	Tue 06/06/23	33;37
→	Implementación estadísticas	5 days	Fri 26/05/23	Thu 01/06/23	33;37
→	Implementación vista estadísticas	3 days	Fri 02/06/23	Tue 06/06/23	46
→	▸ Logros	8 days	Fri 26/05/23	Tue 06/06/23	33;37
→	Implementación logros	5 days	Fri 26/05/23	Thu 01/06/23	33;37
→	Implementación vista logros	3 days	Fri 02/06/23	Tue 06/06/23	49
→	▸ Juego	20 days	Fri 05/05/23	Thu 01/06/23	28
→	Implementación juego	15 days	Fri 05/05/23	Thu 25/05/23	28
→	Implementación vista de juego	5 days	Fri 26/05/23	Thu 01/06/23	52
→	▸ Pruebas	10 days	Wed 07/06/23	Tue 20/06/23	26
→	Pruebas de componentes	10 days	Wed 07/06/23	Tue 20/06/23	26
→	Pruebas del sistema completo	10 days	Wed 07/06/23	Tue 20/06/23	26
→	Hito: Fin de la primera iteración	0 days	Tue 20/06/23	Tue 20/06/23	54
→	Realización del documento "Memoria"	10 days	Wed 21/06/23	Tue 04/07/23	57

Figura 5: Tareas (parte 3)

Anexo I: Plan del Proyecto

Cómo hemos podido apreciar en las figuras anteriores, tenemos una estimación actual de **116 días laborales** de trabajo para completar el proyecto, y se acabaría aproximadamente el **4 de Julio**.

Finalmente, en el diagrama de Gantt de *figura 5, que se encuentra a continuación*, podemos ver la relación entre las distintas actividades y tareas del proyecto. En rojo, se destacan las tareas que forman parte del **camino crítico**.

Anexo I: Plan del Proyecto

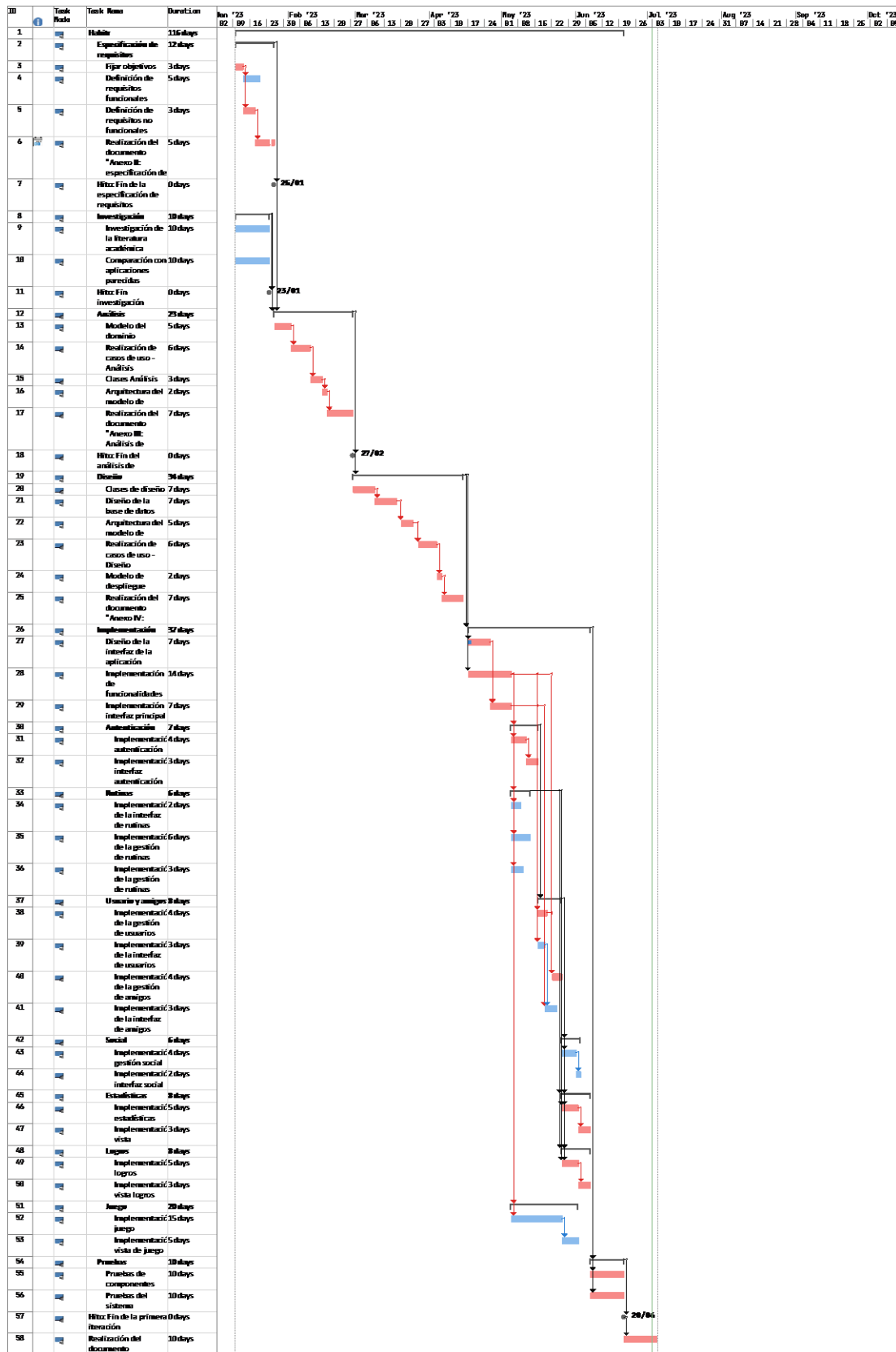


Figura 6: Diagrama de Gantt

En total, según el software, se estima un tiempo de **116 días laborales** de desarrollo.

Comparando con la estimación hecha a través de puntos de casos de uso, que eran **140 días laborales**, siendo ambas estimaciones hechas por métodos independientes y estando ambas en el mismo rango general, podemos confiar en que la estimación hecha es relativamente fiable, y que el desarrollo del proyecto debería de llevar entre 5 y 7 meses de tiempo real.

4 REFERENCIAS

- Pressman, R. S. - "Ingeniería del Software: Un Enfoque Práctico". 7ª Edición. McGraw-Hill. 2010.