



SPRINT PLANNING SPRINT X

FISIO FIND - Grupo 6 - #X

Alberto Carmona Sicre, Antonio Macías Ferrera, Benjamín

Ignacio Maureira Flores, Francisco Capote García, Daniel Alors

Romero, Daniel Fernández Caballero, Daniel Ruiz López,

Daniel Tortorici Bartús, Daniel Vela Camacho, Delfín Santana

Rubio, Guadalupe Ridruejo Pineda, Julen Redondo Pacheco,

Miguel Encina Martínez, Francisco Mateo Villalba, Pablo

Fernández Pérez, Ramón Gavira Sánchez, Rafael Pulido

Cifuentes

04/02/2025

Índice

1. OBJETIVOS DEL SPRINT	3
2. SPRINT BACKLOG	3
3. METODOLOGÍA INTERNA	3
3.1. Gestión de Tareas en GitHub Project	3
3.2. Organización de las Tareas	4
3.3. Flujo de Trabajo	4
3.4. Definición de Hecho (DoD) de una Historia de Usuario	4
3.5. Gestión de la Configuración	5

Ficha del documento

- **Nombre del Proyecto:** FISIO FIND
 - **Número de Grupo:** Grupo 6
 - **Entregable:** #DP
 - **Miembros del grupo:** Alberto Carmona Sicre, Antonio Macías Ferrera, Benjamín Ignacio Maureira Flores, Francisco Capote García, Daniel Alors Romero, Daniel Fernández Caballero, Daniel Ruiz López, Daniel Tortorici Bartús, Daniel Vela Camacho, Delfín Santana Rubio, Guadalupe Ridruejo Pineda, Julen Redondo Pacheco, Miguel Encina Martínez, Francisco Mateo Villalba, Pablo Fernández Pérez, Ramón Gavira Sánchez, Rafael Pulido Cifuentes.
 - **Autores:** Antonio Macías Ferrera
 - **Fecha de Creación:** dd/mm/aaaa
 - **Versión:** v1.0
-

Histórico de Modificaciones

Fecha	Versión	Realizada por	Descripción de los cambios
04/02/2025	v1.0	Antonio Macías Ferrera	Elaboración de la plantilla del documento.

Participantes

Nombre completo	Rol	Contacto
Alberto Carmona Sicre	–	albcarsic@alum.us.es
Antonio Macías Ferrera	–	antmacfer1@alum.us.es
Benjamín Ignacio Maureira Flores	–	benmauflo@alum.us.es
Francisco Capote García	–	fracapgar1@alum.us.es
Daniel Alors Romero	–	danalorom1@alum.us.es
Daniel Fernández Caballero	–	danfercab@alum.us.es
Daniel Ruiz López	–	danruilop1@alum.us.es
Daniel Tortorici Bartús	–	dantorbar1@alum.us.es
Daniel Vela Camacho	–	danelcam@alum.us.es
Delfín Santana Rubio	–	delsanrub@alum.us.es
Guadalupe Ridruejo Pineda	–	guaridpin@alum.us.es
Francisco Mateo Villalba	–	framatvil@alum.us.es
Pablo Fernández Pérez	–	pablofp.33@gmail.com
Ramón Gavira Sánchez	–	ramgavsan@alum.us.es
Rafael Pulido Cifuentes	–	rafpulcif@alum.us.es

1. OBJETIVOS DEL SPRINT

El propósito de este informe es definir los objetivos a lograr durante el Sprint #X y describir la metodología para alcanzarlos. Se analizarán el proceso de **Sprint Planning**, la gestión de tareas con **GitHub Project**, y el cumplimiento de las estimaciones iniciales.

Los siguientes **objetivos** del *Sprint* harán referencia a las HU desglosadas en la plataforma *GitHub Project*.

- **Objetivo 1:** [Descripción breve del objetivo]
- **Objetivo 2:** [Descripción breve del objetivo]
- **Objetivo 3:** [Descripción breve del objetivo]

2. SPRINT BACKLOG

Objetivo	ID	Funcionalidad
Objetivo 1	HU-XXX	Implementación del login con autenticación
Objetivo 1	HU-XXX	Integración de API externa
Objetivo 2	HU-XXX	Diseño del dashboard de usuario
Objetivo 2	HU-XXX	CRUD 1
Objetivo 3	HU-XXX	Datos de clases

3. METODOLOGÍA INTERNA

3.1. Gestión de Tareas en GitHub Project

El equipo utiliza *GitHub Project* como herramienta de gestión de tareas donde las actividades están organizadas en distintas columnas que reflejan su estado dentro del flujo de trabajo. Esta herramienta cuenta con un **tablero Kanban** para facilitar el seguimiento de las tareas, generación de **gráficas Burn-down** que nos serán útiles en las retrospectivas, y asignación y **estimación de tareas** además de otras funciones que procurarán una buena organización del trabajo.

3.2. Organización de las Tareas

Todas las tareas a ejecutar en el *Sprint* se encontrarán inicialmente en la columna “Product Backlog”, habiendo sido previamente asignadas y estimadas por el equipo *Scrum*.

3.3. Flujo de Trabajo

Cada miembro del equipo será responsable de gestionar el estado de sus tareas ateniéndose al siguiente procedimiento:

1. Inicio de la Tarea

- El desarrollador selecciona una tarea de la columna “Product Backlog” y la traslada a “Todo”.
- Esta acción indica que la tarea ha sido priorizada para su ejecución.

2. Trabajo en Progreso

- Cuando se comienza a trabajar en la tarea, se mueve a la columna “In Progress”.
- Se debe registrar el tiempo de trabajo en **Clockify** de acuerdo al protocolo y la política de nombrado especificada en el **Plan De Gestión De La Configuración**.

3. Revisión de Código: Revisión por pares

- Al finalizar la implementación, el responsable de la tarea crea una *Pull Request (PR)* y traslada la tarea a la columna “In Review”.
- El otro miembro del equipo asignado se encarga de analizar el código y verificar su calidad.
- Si la revisión es satisfactoria, el revisor aprueba la PR y fusiona los cambios.
- Si se identifican errores o mejoras necesarias, la tarea se devuelve a “In Progress”, notificando los ajustes requeridos.

3.4. Definición de Hecho (DoD) de una Historia de Usuario

Para que una historia de usuario (HU) se considere terminada, debe cumplir con los siguientes requisitos:

- La funcionalidad **debe** estar completamente desarrollada y *cumplir con los requisitos* especificados en la HU.
- Se deben **satisfacer** las **expectativas** del producto en términos de **comportamiento y usabilidad**.
- El código **debe seguir las buenas prácticas** establecidas por el equipo.

- Se debe **garantizar** la **legibilidad, mantenibilidad** y escalabilidad del código fuente.
- Todo el código **debe ser revisado por al menos un miembro distinto** al desarrollador original.
- El revisor debe verificar que el código funciona correctamente y cumple con los estándares definidos.
- Cada issue debe contar con al menos **un comentario positivo** de otro miembro del equipo antes de su aprobación final.

3.5. Gestión de la Configuración

Desde la **política de versionado** de documentos y de código, hasta la **política de nombrado de ramas**, pasando por el **criterio de mensajes de commits** y el **flujo** de trabajo **GitHub Project - GitHub - Clockify** se encuentra definido en detalle en el ***Plan De Gestión De La Configuración***.

Aprobado por:

Scrum Master: Antonio Macías Ferrera

Rol: [Nombre Apellidos]