

DOCUMENTO DE PLANIFICACIÓN

Proyecto CMS - Escuela de Baile Urbano

1. ENTORNO DE TRABAJO

1.1 Herramientas Colaborativas

Discord

Uso: Videollamadas, reuniones y compartir pantalla. Justificación:

- Llamadas sin límite de tiempo
- Compartir pantalla para mostrar el WordPress desde un único portátil
- Canales organizados por temas
- Todos los miembros ya lo usan habitualmente

Casos de uso específicos:

- Reuniones de planning y retrospectivas
- Sesiones de trabajo conjunto
- Demostración del WordPress al equipo antes de presentar al profesor
- Resolución de problemas técnicos en tiempo real

Microsoft Teams

Comunicación diaria, almacenamiento de borradores y planificación. Justificación:

- Chat estructurado por conversaciones
- Almacenamiento de archivos con versiones
- Calendario/agenda integrado para seguimiento de tareas
- Chat general: comunicación diaria entre miembros, coordinación de reuniones y dudas.

GitHub

Control de versiones, repositorio oficial y referencia principal. Justificación:

- Control de versiones profesional
- Almacenamiento ordenado de documentos
- Organización clara por entregas
- Historial completo de cambios (commits)

Estructura del repositorio:

cms-escuela-baile/

- README.md (descripción del proyecto)
- Sprint-1/
 - 01-Requisitos.pdf
 - 02-Analisis.pdf
 - 03-Diseño.pdf
 - 04-Implementacion.pdf
 - 05-Planificacion.pdf
- docs/
 - guia-entrevista.pdf
 - backlog.md
- assets/
 - mockups/
 - diagramas/
 - capturas-wordpress/

Política de subida:

- Solo se sube la versión FINAL de cada documento
- Los borradores permanecen en Teams

1.2 Flujo de Trabajo Centralizado

WordPress en un Único Portátil

Todo el desarrollo de WordPress se realizará en un solo portátil del equipo.

Razones:

1. Evitar problemas de sincronización entre entornos locales




2. Configuración única y consistente
3. Facilita mostrar el trabajo al profesor desde un mismo equipo
4. No hay conflictos de versiones de plugins o base de datos

Proceso:

1. Instalación inicial: Un miembro instala WordPress en su portátil (MAMP)
2. Sesiones de trabajo:
 - o Reuniones por Discord compartiendo pantalla
 - o Uno trabaja en el portátil mientras otros revisan
 - o Rotación del portátil en reuniones presenciales
3. Backups regulares:
 - o Export de base de datos (.sql) → Subir a Teams y GitHub
 - o Carpeta wpdb completa → Subir comprimida a Teams
 - o Screenshots de progreso → Subir a Teams y GitHub

Traspaso de Información entre Miembros

Ejemplo práctico:

1.  Person crea mockups en Balsamiq → Subir .png a Teams/Borradores
2.  Person los descarga → Implementa las páginas en WordPress → Hace screenshots
3.  Person documenta en Word → Sube borrador a Teams → Revisa equipo
4. Versión final → Se sube a GitHub como PDF
5. Profesor revisa en GitHub y da feedback

1.3 Comunicación según Tipo de Tarea

Tipo de comunicación	Herramienta	Ejemplo
Reuniones planificación	Discord (videollamada)	Sprint Planning
Trabajo conjunto	Discord (compartir pantalla)	Configurar plugins juntos
Chat diario	Teams (chat)/Discord/Whatsapp	"He terminado los mockups"
Almacenar borradores	Teams (archivos)	Versiones de trabajo de requisitos

Tipo de comunicación	Herramienta	Ejemplo
Seguimiento de tareas	Teams (calendario)	Ver qué está pendiente
Versión final oficial	GitHub	PDF final para el profesor
Historial de cambios	GitHub (commits)	"Actualizado diagrama de clases"

1.4 Herramientas de Desarrollo

MAMP

Uso: Servidor local para WordPress (instalado en UN portátil del equipo)

Componentes:

- Apache (servidor web)
- MySQL (base de datos)
- PHP (lenguaje de WordPress)

Ventajas de un único portátil:

- Configuración única
- Base de datos centralizada

Balsamiq/Figma

Uso: Creación de mockups Proceso:

1. Diseñar mockups
2. Exportar como PNG
3. Subir a Teams/Borradores
4. Versión final → GitHub/assets/mockups/

Draw.io

Uso: Diagramas de clases y arquitectura Proceso:

1. Crear diagrama online

2. Exportar como PNG y XML
3. Subir a Teams/Borradores
4. Versión final → GitHub/assets/diagramas/

2. METODOLOGÍA SCRUM

Software utilizado: Taiga

Justificación: Software especializado en la utilización de SCRUM, backlogs, burndown chart. Gratuito y permite exportar el contenido.

2.1 Adaptación al Proyecto

- Scrum Master: Facilita reuniones
- Product Owner: Prioriza backlog
- Developers: Todos los miembros participan en desarrollo y documentación

Sprint Planning

- Cuándo: Inicio de cada sprint
- Dónde: Discord (videollamada)
- Objetivo: Seleccionar tareas del Product Backlog para el Sprint Backlog

Daily Standup (adaptado)

- Dónde: Teams
- Formato: Cada uno escribe:
 - ¿Qué hice hoy?
 - ¿Qué haré mañana?
 - ¿Tengo impedimentos?

Sprint Review

- Cuándo: Día de entrega o día de antes
- Dónde: Laboratorio (presencial) o Discord
- Objetivo: Demostrar trabajo al profesor (cada Lunes en horario lectivo)

Sprint Retrospective

- Cuándo: Después de cada sprint
- Dónde: Discord (videollamada)
- Objetivo: ¿Qué funcionó? ¿Qué mejorar?

3. PRODUCT BACKLOG

Lista completa de tareas del proyecto, ordenadas por prioridad.

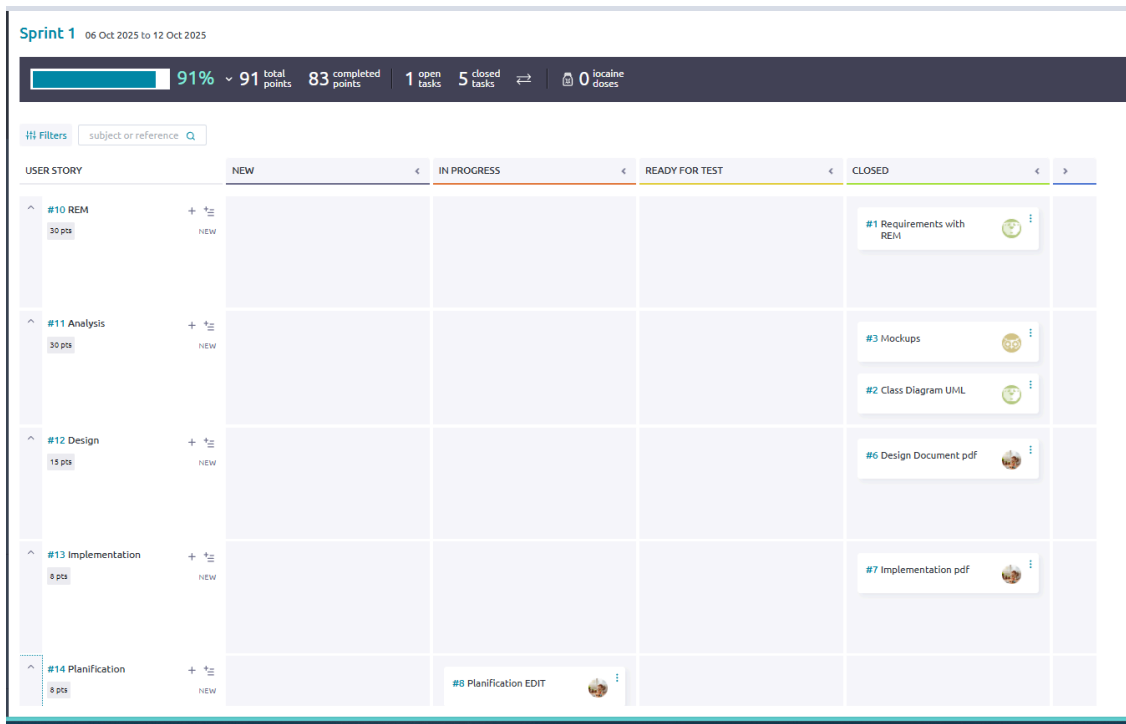
Leyenda de Estados

- New - No iniciada
- In progress - Alguien trabajando
- Ready for test - revisión
- Closed - Terminada y revisada
- Needs info - poca información para realizar la tarea



4. SPRINT 1 BACKLOG

Tareas seleccionadas para completar en el primer sprint (hasta el **Date**). Objetivo del Sprint 1: Documentar requisitos, análisis, diseño, instalar WordPress e informar sobre planificación.



Total horas Sprint 1: 22 horas Fecha inicio: Date Fecha fin: Date Duración: 7 días

5. BURNDOWN CHART - SPRINT 1

