

### 1.3. Planificación

Para la planificación y la distribución de las tareas se usará la plataforma de Taiga, un software de organización de proyectos.

La estructura será la siguiente:

#### → Épicas

Ordenadas en fases, son la agrupación y división por temas de los distintos puntos a entregar.

- **Definición del proyecto y primeros esquemas** (Código de color: rojo)

Definición inicial del proyecto, necesaria para la creación base de este.

Se dará una descripción, un estudio de mercado previo, se debe definir los objetivos iniciales que pueden variar según avance el proyecto.

Creación de los diagramas necesarios para la estructura de la base de datos, definir sus elementos y relaciones.

Diseño de las pantallas web: Prototipado inicial y responsive.

Realizar pruebas con tecnología no vista durante el curso: El prototipado con Adobe XD.

- **Despliegue de la aplicación** (Código de color: amarillo)

Despliegue de la aplicación en un servidor web. Al ser una aplicación en Jakarta, se realizará mediante Wildfly y Docker.

- **Arquitectura de la aplicación** (Código de color: naranja)

Backend de la aplicación. La estructura será realizada en Jakarta EE.

- **Diseño de la aplicación** (Código de color: verde)

Frontend del proyecto. Siguiendo con los prototipos creados previamente se buscará el diseño estructurado de los siguientes puntos:

- Página principal.
- Login y Signup para los alumnos.
- Perfil del alumno.
- Listados de Mundos y Niveles.
- Test a realizar

También se buscará un Login diferenciado para los profesores, con el perfil también personalizado para estos. Avisos dinámicos de logros conseguidos, pruebas superadas, etc.

- **Testing JUnit + Jasmine** (Código de color: azul)

Testing de las funciones de la aplicación, tanto para el Backend como para el Frontend.

- **Memoria del proyecto** (Código de color: **morado**)

Redacción de los puntos a entregar + Guías de usuario y despliegue.

- **Revisión final** (Código de color: **negro**)

Comprobación de funciones y despliegue correcto final. Documentación redactada por completo.

### → Sprints

Los tiempos estipulados para realizar determinadas partes del proyecto. Son un total de 6 intervalos, cada uno engloba unas determinadas historias de usuario:

- Planificación inicial - Prototipos - Esquemas
- Despliegue de la aplicación
- Primera versión básica de la web
- Necesidades avanzadas de la web
- Despliegue final - Revisión de funciones
- Revisión final

### → Historias de Usuario y Tareas

Los objetivos y puntos necesarios para el desarrollo del proyecto.

Las historias de usuario son requisitos que pediría el supuesto cliente para la aplicación.

Están relacionados a una Épica en concreto, y a su vez delimitados en un Sprint. Cada una de ellas cuenta con tareas por realizar. Hay tareas sin una historia concreta, sobretudo los esquemas y documentaciones pertinentes para el proyecto estarían catalogadas aquí.

- Planificación inicial - Prototipos - Esquemas
  - **Realización de diagramas de BD:** \*

Estructura inicial de la base de datos.

- **Prototipado inicial de la aplicación:** \*

Prototipos de la pantalla de inicio, login y signup, perfiles y listados.

- **Tareas sin historia:**

Diagramas de base de datos, E/R y de clases. Puntos del 1.1 al 1.4 de la memoria del proyecto.

- Despliegue de la aplicación

- **Tareas sin historia:**

Despliegue inicial de la aplicación.

- Primera versión básica de la web
  - **Pantalla principal:** \*
  - **Listado de Mundos/Niveles:** \*\*
  - **Login + Signup:** \*\*
  - **Tarea sin historia:**

Diseño inicial de las pantallas. Estructura de la aplicación en Spring.

- Necesidades avanzadas de la web
  - **Tarea sin historia:**
- Despliegue final - Revisión de funciones
  - **Tarea sin historia:**

- Revisión final
  - **Demo final:** \*

Demostración del producto final.

- **Tarea sin historia:**

Revisión de la documentación, código e interfaz.