



Actividad 1. Primera fase de desarrollo, revisión y ajuste de los Sprints

Adrián González, Luis Lira, José Ma. Gutiérrez.

Licenciatura en Desarrollo de Sistemas Web

Proyecto II

Enlace React

- <https://github.com/LuisLiraC/SprintPlanning>

Enlace trello

- <https://trello.com/b/8JFLHnzv/proyecto-api>

Avances de los sprints

Actividades completadas (Sprint #1 – En curso)

- **D01: Definición de tecnologías**
 - Se escogieron las tecnologías con base a los conocimientos del equipo de desarrollo, las cuales están más enfocadas hacia Javascript.
 - Para el desarrollo del backend se decidió usar Express.js, un framework de Node.js bastante usado para la creación de API Rest con Javascript.
 - Para almacenamiento, se escogió una base de datos NoSQL que fue MongoDB, debido a que hay un servicio gratuito llamado MongoAtlas.
 - Para el desarrollo del frontend se escogió Pug/Jade como motor de plantillas de Express, para crear las vistas si necesidad de crear un frontend por separado.
 - Heroku fue la plataforma elegida para el despliegue de la aplicación debido a que también es un servicio gratuito para ejecutar aplicaciones de Node.js, además de contar con otro servicio gratuito de MongoDB.
 - También, se escogió la librería CodeMirror, para la creación de la documentación y dar una apariencia más profesional.
- **D02: Diseño de datos**
 - El diseño de los datos se planteó desde una perspectiva de archivos JSON, ya que son el tipo de archivos con los que MongoDB trabaja por defecto.
 - Los datos elegidos, fueron escogidos según al nivel de importancia de información de los personajes y de qué tanto cambian con el tiempo.
- **D03: Creación de servidor**

- Se creó el servidor en local para comenzar el desarrollo y se corrió con Node.js.
- **D04: Creación de mockup base:**
 - De acuerdo a los datos que fueron elegidos en la fase de estructura de datos, se creó un archivo JSON vacío de la estructura que tendrán la información de los campeones.
- **D05: Creación de petición GET all:**
 - Se creó el endpoint para para petición GET de todos los campeones, como en este caso aún no existe información creada en la base de datos, ni una conexión a la base de datos, se está trabajando con un variable de tipo objeto nativa de Javascript, y por ahora esa es la respuesta que está devolviendo el servidor.

Ajustes al proyecto:

- **D10: Creación de Web Scraper (Agregado al product backlog):**
 - Se decidió crear un Web scraper, ya que la tarea de extraer la información de cada campeón de los sitios oficiales de manera manual podrá representar inconvenientes en el tiempo de entregar del requisito.
 - Un web scraper es una aplicación que se encarga de analizar sitios web y extraer información de los mismos.
 - Para realizar un web scraper primero debe analizar la estructura de la página para poder indicar por medio de programación los datos que extraerá.
 - Se realizará con Node.js y Puppeteer. Puppeteer es una librería especializada para creación de scraper, Node.js se usará para la ejecución del scraper y para crear el archivo .json de cada campeón al finalizar el scraping.