-

***PROLED***

**DAM 20129-2020**

Andoni Vidal, Karlos Perdiguero, Ion Ciganda, Gorka Apodaca

# **INTRODUCCIÓN**

## **¿Qué es Proled?**

Proled como abreviatura de “Proyecto de Leds”, es un proyecto compuesto de distintos juegos realizado por alumnos del grado Desarrollo de aplicaciones multiplataforma en adelante “DAM” para el curso 2019-2020.

Proled comprende el uso de un amplio abanico de herramientas multiplataforma que permite poner en práctica los conocimientos previos y los adquiridos en el proceso por los alumnos con el fin de completar los distintos módulos que componen el grado de DAM, todo ello encapsulado en un proyecto.

Principalmente se han desarrollado dos juegos para completar el proyecto:

* Tres en raya

Juego a turnos pulsando celdas cuyo objetivo es que uno de los dos jugadores complete un tres en raya.

* Simon

Juego de memoria que consiste en pulsar las mismas celdas en un orden propuesto aleatoriamente.

El tres en raya permite el uso de varias partidas simultáneas multijugador con dispositivos móviles. El Simon es un juego para 1 jugador. Opcionalmente se hace uso de un panel LED que ilumina los resultados sincronizados con los propios terminales de los jugadores.

## **¿Por qué Proled?**

La experiencia e ideas de los alumnos de cursos anteriores así como las presentaciones del año anterior, fueron claves para determinar que el proyecto además la propia aplicación de dispositivos móvil, se complementará con tecnologías IOT representadas en un panel de LEDs.

El poder representar tanto en los terminales como en los paneles llevó algo de más tiempo en decidir. Los primeros días de investigación de ideas, primero individualmente y posteriormente puestas en común, daban como resultado dos posibles usos para dar cuerpo al proyecto. Por un lado basar la aplicación en uno o más juegos, por otro lado orientar el proyecto a sistemas de iluminación de lugares programables desde terminal móvil. La interactividad entre usuarios entre otros factores decantó la balanza al lado de los juegos.

Entre los juegos, las ideas iniciales contemplaban un tres en raya, un tetris, el pong… Posteriormente se propuso hundir la flota y por último el simon. El tres en raya sería el punto de partida al catálogo de juegos que ofrecería el Proled.

# **OBJETIVOS**

## **Objetivos generales**

Uno de los objetivos era poner en práctica los conocimientos personales, los adquiridos en el curso de introducción de septiembre 2019 y los adquiridos a lo largo del propio proyecto.

El otro objetivo era el aprendizaje y puesta en práctica de la planificación y fases de un proyecto, el uso de metodologías ágiles, el trabajo en equipo, la disciplina y el respeto. Estos factores eran los pilares para que llegase a buen puerto tanto el proyecto del curso como cualquier proyecto empresarial en entorno laboral.

## **Objetivos de desarrollo**

Las necesidades de nuestro proyecto partían de una comunicación cliente-servidor en tiempo real para la simultaneidad de respuesta entre todos los terminales móviles que estuviesen participando en el mismo juego.

La biblioteca de JS **socket.io** respondía perfectamente para cumplir esa función. Adicionalmente, nos surgía la necesidad de profundizar en sus funcionalidades para la gestión de las salas de forma que nos permitiese jugar varias partidas multijugador de forma simultánea.

El servicio de *crud* para el almacenamiento y tratamiento de datos con protocolo REST los gestionaría un servidor **Springboot**. Esto requeriría planificar un tiempo de estudio de la tecnología para posteriormente llevarlo a la práctica.

Por la experiencia laboral de los componentes del grupo en el uso de bases de datos relacionales, incorporar una base de datos no-relacional basado en **MongoDB** supondría una mejora importante en los conocimientos y posibilidades para aplicar en proyectos futuros.

## **Objetivos de equipo**

El curso está orientado a guiar a los alumnos en los pasos necesarios para la consecución de un objetivo común.

El trabajo en equipo, compartir conocimientos, ayuda al compañero son la base para el éxito de cualquier proyecto independientemente del alcance ya sea en fase de enseñanza o en entorno laboral.

Las reuniones, la planificación a corto y medio plazo, la documentación compartida todo ellos supervisado por nuestros profesores dentro del marco de metodologías ágiles, nos permiten anticiparnos a la mayoría de problemas y dar respuesta rápida a las no previstas.

El equipo dispone y hace uso de herramientas de comunicación, control de versiones de código, tableros de administración de tareas y unidades de almacenamiento compartido en cloud para cubrir todas las posibles necesidades que pudieran surgir durante la ejecución del proyecto.

# **DOCUMENTACIÓN**

## **Diseño**

Después de decantarnos por el inicio del proyecto con el tres en raya. Las primeras reuniones consisten en identificar las funcionalidades que tendría el juego respetando siempre los requisitos de proyecto que establece el profesor para este proyecto.

Con todo ello, el resultante sobre el papel nos queda una arquitectura separada por capas y orientada a servicios.

Respetando las metologías ágiles, acotamos todas las funcionalidades que tendría el proyecto final reduciéndolo a una versión 1 precaria para poder sobrellevar posibles nuevos cambios de criterios del Scrum master haciendo planificaciones más a corto-medio plazo.

Sintetizadas ya las funcionalidades que contemplan la versión 1 de la aplicación, realizamos un análisis funcional y un análisis técnico de las características.

De la misma forma identificamos las piezas necesarias para completar la parte del panel LED.

## **Fases**

**Fase 1:**

En una primera fase, montamos todas las herramientas de las que dispondrá el equipo para la realización del proyecto.

Entornos virtuales.

Servidores (producción y test) en Heroku

Repositorios (producción y test) en Bitbucket

Comunicacion con Slack

Planificación con Trello

Control de versiones con GIT.

**Fase 2:**

En la fase 2 damos cuerpo a las funcionalidades acotadas para la versión 1 del proyecto.

Programamos una versión limitada del cliente sin dedicar un tiempo excesivo al aspecto de la aplicación. Ignoramos de la misma forma la propia lógica del tres en raya, en consecuencia tampoco hay validaciones ni restricciones de ningún tipo. El test es puramente funcional a nivel de elementos.

Del lado servidor, limitamos por cógido los usuarios de la partida así como los movimientos a almacenar.

**Fase 3:**

En la fase 3 se implementa la lógica de la aplicación.

En el lado cliente y una vez capturados todos los eventos, testeamos el flujo correcto del tres en raya.

En el lado servidor implementamos la gestión de las salas y testeamos la comunicación enviada y recibida por los sockets. Se implementa la validación de la lógica de la partida a nivel servidor.

Paralelamente se prueba la comunicación con el panel LED montado.

**Fase 4:**

En la fase 4 se avanza a una siguiente versión de la aplicación.

Se implementa una validación a nivel de cliente.

Se define el objeto partida. Esto nos permite ampliar por un lado la posibilidad de jugar varias partidas en paralelo y por otro lado definimos la estructura para un posterior almacenamiento y tratamiento de datos.

Ajustamos las validaciones de todos los endpoints desde que se pulsan celdas en el terminal móvil hasta que se encienden las luces del panel y viceversa para evitar respuestas no-lógicas que se salten pasos.

**Fase 5:**

En la fase 5 entra en escena la base de datos y la autentificación.

Se implementa un login basado en Firebase, (Plataforma que permite integrar Bases de datos, autenticaciones, almacenamiento, hosting,etc.. en las APP).

En este caso se ha elegido la autenticación con google para entrar a proled.

Hay que crear un proyecto en Firebase, habilitar la autenticación deseada, modificar ciertos archivos como build.gradle, mainactivity.java en la App. Firebase generará un archivo google-services.json que deberemos integrar también en la APP y por último también es necesario generar los certificados de firma de depuración (SHA-1) para el desarrollo.

Incorporamos una base de datos en Mongodb para el almacenamiento de los datos del jugador, la partida, ranking etc…

En el lado de cliente se desarrolla un panel exclusivo para la gestión del usuario y el tratamiento y consulta de datos de las partidas de Proled.

**Fase 6:**

En la fase 6, convertimos el Proled en un proyecto multijuegos.

Se añade un panel previo al tres en raya que nos permite seleccionar el juego al que queremos jugar.

Se desarrolla el juego “Simon”.

Al carecer del tiempo suficiente de desarrollo y dadas las circunstancias excepcionales que hemos vivido se ha definido el juego para un sólo jugador, que tratará de pulsar las secuencias de colores que se van representando automáticamente, aumentando la dificultad con un color más cada nivel, hasta que el usuario falle.

**Fase 7:**

Por último se trata de representar las partidas de los juegos mencionados sobre un panel de leds RGB, con un modelo de panel LED WS2812 de 16x16 con un total de 256 leds RGB.

Se descarga el programa ARDUINO y se configura correspondientemente para que acepte nuestro controlador ESP32, ya que son placas diferentes con pines analógicos y digitales en distintas salidas.

Hay dos grandes partes diferenciadas en la lógica de ésta representación visual de las partidas. una sería la lógica de parte del servidor NODE, donde se configuran las conexiones con el servidor MQTT de AWS y después de las escuchas de las pulsaciones de la parte cliente se mandan los topics o ‘mensajes’ que interpretará después el controlador mencionado.

Y la segunda parte es programar el sketch de arduino, éste estará debidamente configurado con la IP de AWS y la conexión WIFI donde queramos operar, ya que el ESP32 recibirá los mensajes enviados desde MQTT por dicho WIFI.

El panel led representará los movimientos programados según el mensaje que reciba, el circuito de la placa es sencillo, se conecta un cable a la corriente de 5v (positivo), otro al ground (negativo) y otro será el pin de entrada de datos.

La función Callback controlará cada TOPIC y actuará en función del PAYLOAD recibido.

## **Dificultades**

La principal dificultad en el proyecto ha sido la coordinación de los propios compañeros de equipo en las fases iniciales.

Por un lado, la gran cantidad de ideas y la experiencia de programadores da como resultado una amplia gama de soluciones propuestas para afrontar un mismo problema. En algún momento la falta de comunicación ha originado que la resolución de algunas funcionalidades desemboca en la incompatibilidad de las mismas ya sea en la misma capa de cliente o servidor o entre ambas.

Por otro lado, en la planificación de tareas, el orden de escoger las tarjetas del trello tampoco ha ayudado en el correcto desarrollo en tiempo de algunas metas. Problemas como tener funcionalidades del lado cliente terminadas sin poder testear por que nadie se ha ocupado del lado servidor que enlazaban, han sido frecuentes.

Por último, a veces el no seguir las directrices encomendadas por el Scrum master y Product Owner ha resultado que parte del código se ha tenido que rehacer por no cumplir unos mínimos de idoneidad de lógica de programación, rendimiento bajo de la aplicación o mala experiencia del usuario.

## **Soluciones**

Las soluciones a las dificultades encontradas se han basado en aumentar las reuniones periódicas, mejorar la comunicación entre compañeros tanto como con el Scrum master y Product owner.

La priorización de las tareas planificadas del Scrum master en Trello ha ayudado a seleccionar mejor las tareas de forma que ninguna tarea obstruya a otra.

**EVALUACIÓN**

## **Objetivos logrados**

“Proled” nos ha permitido crear una aplicación multijuegos en la que el aprendizaje por el camino ha sido continuo.

* Hemos profundizado los conocimientos adquiridos en el curso preliminar de Septiembre 2019 con ReactNative y Node Js.
* Hemos utilizado protocolos de comunicación vía Socket.
* Hemos incorporado funcionalidades Iot.
* Hemos integrado la autentificación con Firebase.
* Hemos gestionado bases de datos no-relacionales.
* Hemos utilizado herramientas para comunicación y gestión de planificaciones modernas.
* Hemos llevado a la práctica las metodologías ágiles.
* Hemos aprendido a usar el protocolo MQTT con AWS
* Hemos aprendido a configurar Arduino y explorar sus posibilidades con el controlador ESP32, en éste caso un panel LED RGB de 256 LEDS.

## **Objetivos no logrados**

Algunos juegos tanto planteados por nosotros como por el Scrum master no han sido posible incorporarlas por falta de tiempo por eventualidades ajenas a la enseñanza.

Respecto a funcionalidades o teconologías a implementar.

* Autentificación con JWT.
* Montar la API con protocolo rest en un servidor alternativo Springboot.
* Aplicación multididioma.
* Partidas multijugador vía Bluetooth contra el panel LED.

## **Conclusiones**

Por circunstancias excepcionales, por un lado el curso de ciberseguridad y por otro la pandemia global del COVID19 impactando en el día a día en los miembros del equipo, las horas disponibles para dedicar al proyecto se han visto reducidas notablemente dejando pendiente muchas ideas que teníamos pensadas llevar a cabo.

A pesar de ello, no ha impedido que valoremos el proyecto positivamente ya que en el camino hemos profundizado o descubierto en algunos casos, tecnologías y formas de trabajar desconocidas por nosotros hasta ahora y que están copando la actualidad del software en los entornos de trabajo.

De la misma forma esperamos que en futuros cursos y proyectos se pueda continuar por este camino.