

## **Proyecto y Grupo**



## **“Documento Arquitectura Tecnológica”**

**Hito: 2**

**Fecha entrega: 23-01-2017**

**Versión: 1**

### **Componentes:**

- **Emilio Maestre Hortal**
- **Jose Francisco Moreno Fernández**
- **Pablo López Riquelme**
- **Sergio Pérez Seré**

## Tecnologías utilizadas

Hemos utilizado **Node JS** en el Back-End, porque nos pareció la mejor opción para nuestro proyecto pues es una librería bastante de moda que está basada en JavaScript y que tiene un gran rendimiento. Lo hemos utilizado para las operaciones CRUD de



nuestra base de datos principalmente y el resto de acciones de la lógica de negocio de nuestro sistema. Además, tiene un excelente modelo de eventos, ideal para trabajar con programación asíncrona. JavaScript es un lenguaje omnipresente es decir que reduciremos la curva de aprendizaje pudiendo ahorrar tiempo en aprender a utilizar Node.

Por eso resumimos que hemos utilizado node por estas 4 ventajas:

1. El desarrollo es más rápido.
2. La ejecución de test de unidad se puede hacer más rápido.
3. Las aplicaciones son más rápidas y por tanto la experiencia de usuario es mejor.
4. Menor coste de infraestructura

express

**Express** lo hemos utilizado para el manejo de la Api y el control de los eventos y es una infraestructura de aplicaciones web Node.js

mínima y flexible que proporciona un conjunto sólido de características para las aplicaciones web y móviles.

Con miles de métodos de programa de utilidad HTTP y middleware a su disposición, la creación de una API sólida es rápida y sencilla.

Hemos utilizado el framework de **Angular 2** para desarrollar la aplicación de escritorio y la base de la aplicación móvil, y nos hemos decantado por este framework por que actualmente es uno de los más utilizados y uno de los más demandados por las empresas del sector al ser un framework



multiplataforma y que se puede utilizar para cualquier dispositivo. Como con Node JS nos hemos decantado por angular por la facilidad de programar y la rapidez de obtener resultados.

Y por la gran comunidad que hay detrás de este framework apoyado por Google, que cuenta con más material y más tutoriales que otros del mismo tipo.



**Ionic** es una herramienta, gratuita y open source, para el desarrollo de aplicaciones híbridas basadas en HTML5, CSS y JS. Está construido con Sass y optimizado con AngularJS.

Y la hemos elegido porque nos parecía buena idea trabajar con el modelo vista-controlador para tener bien organizado el código y por esta serie de características:

- 1.- Alto rendimiento
- 2.- Ionic trabaja con Angular Js por lo que se pueden anclar bien para hacer nuestra app más robusta.
- 3.- Ionic se inspira en las SDK de desarrollo móviles nativos más populares, por lo que es fácil de entender para cualquier persona que ha construido una aplicación nativa para iOS o Android. Lo interesante, como sabéis, es que desarrollas una vez, y compilas para varios
- 4.- Bonito diseño limpio, sencillo y funcional. Ionic ha sido diseñado para poder trabajar con todos los dispositivos móviles actuales. Con muchos componentes usados en móviles, tipografía, elementos interactivos, etc.
- 5.- Un potente CLI, con un sólo comando podrás crear, construir, probar y compilar tus aplicaciones en cualquier plataforma.

**MySQL** como gestor de base de datos de nuestro sistema.

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base datos open source más popular del mundo.



Hemos elegido utilizar MySQL porque era la que más se adapta a nuestras necesidades y mayor manejo tendríamos sobre ella por el conocimiento previo y sería la más óptima para nuestro sistema.



**WordPress** para la página web de E-learning en la que ofrecemos cursos para dar a conocer nuestro servicio, lo hemos elegido porque nos ofrecía una plantilla de web sobre cursos y moocs que se adapta perfectamente a nuestras necesidades, con total administración por nuestra parte.

**OAUTH** para para la autenticación de usuarios con Apis externas de otros servicios y nos hemos decantado por este porque es uno de los más utilizados y era compatible con firebase.

Oauth es un estándar abierto que fue lanzado hacia finales de 2007 que define un mecanismo para que una aplicación web (cliente) pueda acceder a la información de un usuario en otra (proveedor) sin tener que informar a la primera del usuario y contraseña.



## Internacionalización

En principio no tenemos pensado implementar la aplicación en varios idiomas, pero si la aplicación funcionase y se llevase a cabo de forma definitiva sí que se implementará en varios idiomas, pero lo dejamos como una propuesta de funcionalidad adicional.