

RESUMEN

El proyecto *MediApp* consiste en el diseño y desarrollo de una aplicación web para la gestión médica basado en 3 tipos de usuario: administradores, especialistas y pacientes. Para ello se han utilizado los *frameworks* de Angular para *frontend* y de ExpressJS para *backend*.

Las funcionalidades principales de la aplicación la gestión de especialidades médicas y usuarios por parte de los usuarios administradores, la gestión de informes médicos y prescripciones de medicamentos por parte de los usuarios especialistas y la solicitud de nuevas citas, almacenamiento de mediciones y consulta de sus datos médicos por parte de los pacientes.

Durante estas semanas hemos afrontado diferentes dificultades, la mayoría de ellas debido al reto que supone desarrollar el propio proyecto a la vez que se aprende el manejo de tecnologías completamente nuevas como los dos *frameworks* arriba mencionados, el uso de JSON Web Tokens para la autenticación y autorización, el manejo de SCSS o el uso de nuevas herramientas como *Postman*.

Es por ello por lo que como principal conclusión podemos decir que el proyecto nos ha servido como método de aprendizaje no sólo de estas herramientas y tecnologías sino también de una nueva forma de arquitectura y organización de software.

También hemos aprendido que, durante el ciclo de vida del desarrollo de software, la implementación de nuevas funcionalidades puede provocar la aparición de *bugs* en funcionalidades ya desarrolladas que deben ser arreglados a la vez que se desarrollan estas nuevas funcionalidades para que la aplicación continúe con su desarrollo de forma óptima.

El código fuente de este proyecto puede ser encontrado en este [repositorio de GitHub](#) bajo licencia GNU GPLv3.

ABSTRACT

The MediApp project involves the design and development of a web application for medical management based on three types of users: administrators, specialists, and patients. For this purpose, we have used Angular for the frontend and ExpressJS for the backend framework.

The main functionalities of the application include the management of medical specialties and users by administrator users, the management of medical reports and medication prescriptions by specialist users, and the scheduling of new appointments, storage of measurements, and consultation of their medical data by patients.

During these weeks, we have faced various challenges, most of them due to the difficulty of developing the project while simultaneously learning how to use completely new technologies such as the two frameworks mentioned above, the use of JSON Web Tokens for authentication and authorization, the handling of SCSS, and the use of new tools like Postman.

Therefore, our main conclusion is that the project has served as a learning method not only for these tools and technologies but also for a new form of software architecture and organization.

We have also learned that during the software development lifecycle, the implementation of new functionalities can cause bugs in already developed functionalities, which must be fixed while developing these new functionalities so that the application continues to develop optimally.

The source code for this project can be found in this [GitHub repository](#) under the GNU GPLv3 license.

CONCLUSIONES

En conclusión, gracias al conocimiento que hemos conseguido desarrollar durante estas semanas podemos decir que hemos cumplido con los objetivos que nos marcamos antes de iniciar el diseño de la aplicación, permitiendo que los tres tipos de usuarios puedan realizar las funciones que nos marcamos como principales, consiguiendo además una interfaz funcional, accesible e intuitiva siendo visible en dispositivos de pantalla pequeña, mediana y grande.

Obviamente al ser un desarrollo de pocas semanas el potencial de la aplicación, así como su margen de mejora es enorme. Por ejemplo, se podrían añadir nuevas funcionalidades como un chat en vivo con tu médico, la posibilidad de realizar video-consultas, un sistema de solicitud de pruebas médicas como pruebas radiológicas, analíticas, etc.

Además, se podrían añadir nuevos perfiles de profesiones sanitarias como enfermeros, fisioterapeutas, técnicos de laboratorio... lo que posibilitaría a su vez crear nuevas funcionalidades como crear tablas de ejercicios personalizados para cada paciente, generación de informes de resultados de analíticas, etc.

Por último, el código actual podría ser mejorado en cuanto a optimización, rendimiento y apariencia con un mayor conocimiento de las tecnologías utilizadas que han sido explicadas a lo largo de este documento.