

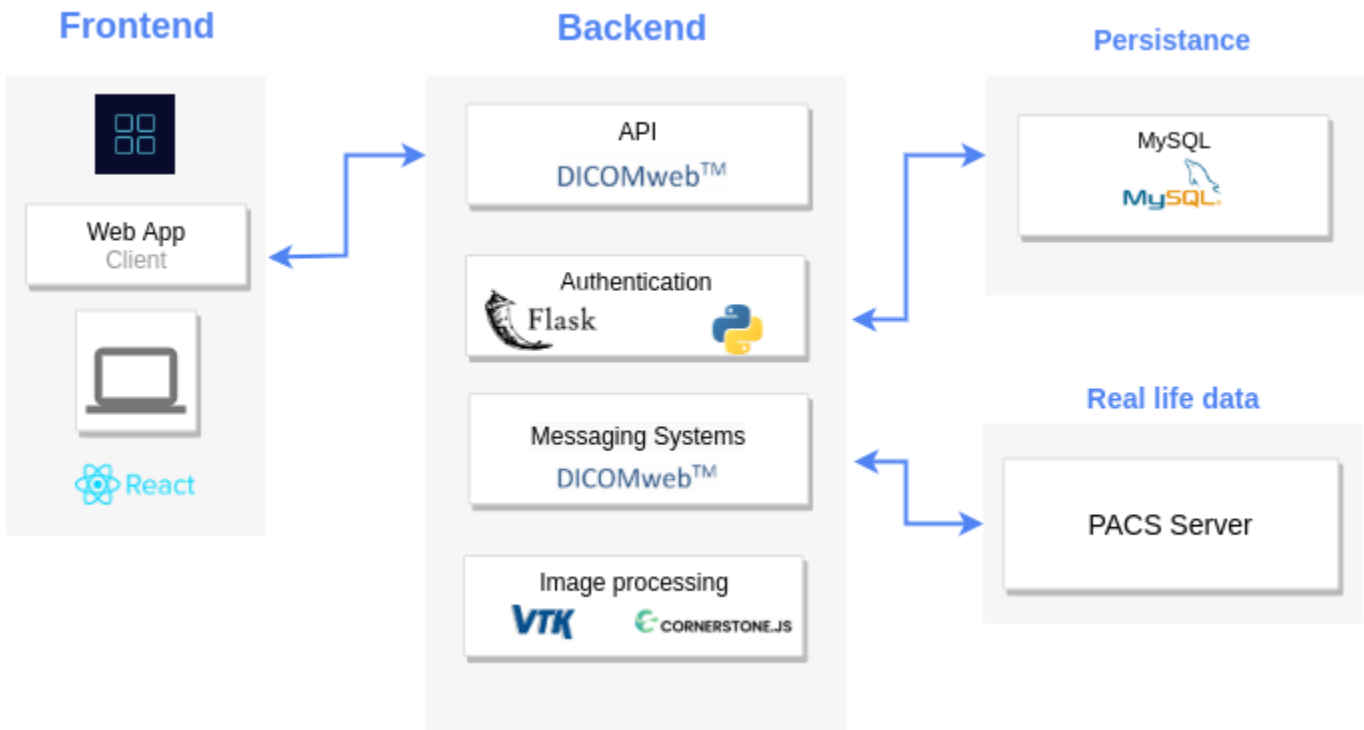
## Architecture Decisions

Impact	HIGH
Contributors	@ Pedro Sobral
Resources	<a href="https://reactjs.org">https://reactjs.org</a> <a href="https://ohif.org">https://ohif.org</a> <a href="https://www.dicomstandard.org/using/dicomweb">https://www.dicomstandard.org/using/dicomweb</a> <a href="https://vtk.org/about">https://vtk.org/about</a> <a href="https://www.cornerstonejs.org/docs/getting-started/overview">https://www.cornerstonejs.org/docs/getting-started/overview</a> <a href="https://devops.com/8-advantages-using-mysql/">https://devops.com/8-advantages-using-mysql/</a> <a href="https://www.fivetran.com/blog/postgresql-vs-mysql">https://www.fivetran.com/blog/postgresql-vs-mysql</a> <a href="https://www.orthanc-server.com/static.php?page=about">https://www.orthanc-server.com/static.php?page=about</a>

### Architecture

In this page will be explore the details to have a good implementation of the system, as well the pros and cons to choice a technology instead of other. So all the architecture will be deeply and extensively analyzed.

The next figure presents the architecture defined for our system. In a first view we can separate the architecture in 4 main modules: Frontend, Backend, Persistence, PACS Server. In the frontend module, it is suppose to use the OHIF viewer, so because of that, the technology selected were React. In the backend module, we have to deal with many aspects, a API is necessary to communicates with frontend and the server, to generating and processing the 3D images it is important to have an Image Processing component. It is extremely important to have a secure system, so that an Authentication protocol will be implemented so granted that. In the Persistence Module, we look forward to a simple database, MySQL, to store user profiles mainly. In the PACS Server module, will be used a Orthanc Server that allows us to handle DICOM images.



### Frontend

As referred before this module is built using the OHIF viewer, and as this viewer work with React, it will be used as well in our project.

- React: Is a JavaScript framework that is very popular nowadays, with React it is possible to creates complex pages in a facilitated way since some componentes need for it were already develop. Another advantage of using React is the speed of writing code, is much more

faster developing a Web Platform. The previous factors and the factor that some elements of the group already had a contact with the technology make us using React. It is important to highlight System Manager platform will also be created using the React library.

- OHIF: The Open Health Imaging Foundation is a medical imaging viewer that is open-sourced and web-based which can be customized in order to connect to image archives that support DicomWeb, helping to map proprietary API formats. It has so many advantages, such as:
  - It enables the deployment of imaging applications on premise or in the cloud without requiring installation of custom software on the user's computer.
  - It decreased costs and information technology support requirements, as well as improved accessibility across sites.
  - It can also be modified in order to support site-specific workflows and accommodate evolving research requirements.

A disadvantage that we see is that can be a little bit complicated, at the first to understand all the viewer code, although reading the documentation (<https://docs.ohif.org>) we will pass this difficulty.

## Backend

The backend is divided in 4 components: API, Authentication, and Image Processing.

- API: The API is a RESTful DICOM service called DicomWeb, it is able to sending, retrieving and querying for medical images and related information. It is a light-weight way to access the images, and it was built to be easy to implement by developers who have minimal familiarity with the DICOM standard.
- Authentication: The security is a very important factor in our project, we are dealing with medical images, and the security of the data is and the privacy of the patient are very relevant. In order to make our web application secure, will be used a authentication system based on a E-CHAP protocol, so granted only the authorised have access to the their information.
- Image Processing: One of the main features of the project is to develop ways and tools to see medical images in different ways (views), and to make 3D images in the different images and views orientation. In order to implement those features will be use a JavaScript Library called VTK.js that is a software system for 3D computer graphics, modelling, image processing, volume rendering, and 2D plotting. So there are many VTK.js's features that are relevant for us, so that we will use this library.

We also want to have a synchronisation between multiple viewports, to do this we will use another JavaScript Library, Cornerstone.js. This library have a bunch of relevant features like: Rendering stacks of volumes viewports; Manipulate Images (zoom, scroll); Make annotations in the images (measure distances, write importante things, supports multiple viewports); Segmentation of 3D images in the Volume viewports; and Synchronisation between multiple viewports.

## Persistence

In this module a database will be implemented, with the objective of storing the user informations of the system, mainly logins and tokens to identify users. As we already work with MySQL in previous project, we decide to use this database. It is light, it is easy to scale, and offers a good protection in terms of security.

## PACS Server

We need a server to communicate and store our images, so we need a PACS server, we will use the Orthanc one. Orthanc is a open-source, lightweight DICOM server for healthcare and medical research. It is designed to have the following objectives:

- To ease DICOM scripting for clinical routine;
- To ease data management for clinical routine and medical research;
- To bring DICOM images to the Computer Vision systems;

It hides the complexity of DICOM format and DICOM protocol and lets the users give more attention to the content of the DICOM files. A very good thing about this server is that it supplies a RESTful API, which allows to driver Orthanc from any computer language, becoming a very good server to implement on our project.