**EVAREWIEW**

**Project Analysis Report**

****

**Batuhan İşcan**

**Yiğit Özarslan**

**Ozan Özkök**

**Project Repository Link:** [**https://github.com/PelinalWS/CMPE\_491**](https://github.com/PelinalWS/CMPE_491)

[**Document Revision History 2**](#_axh8yu2r6k5k)

[**1. Introduction 3**](#_4nh1qfkebydq)

[**2. Current System 3**](#_goq0avxuphe4)

[**3. Proposed System 3**](#_r4j0ahbzbxjh)

[**3.1. Overview 3**](#_b3i26iworfls)

[**3.2. Functional Requirements 3**](#_f732kvi1f7fq)

[**3.3. Nonfunctional Requirements 3**](#_ylxkqm9cvual)

[**3.4. System models 3**](#_629pd15znnlq)

[**3.4.1. Scenarios 3**](#_esvo8scjkn8x)

[**3.4.2. Use case model 3**](#_cgdzhmoujwyl)

[**3.4.3. Object and class model 3**](#_11trwdt0p73g)

[**3.4.4. Dynamic models 3**](#_dqamod2nimyx)

[**3.4.5. User interface - navigational paths and screen mock-ups 4**](#_xryc69cxozkq)

[**4. Glossary 4**](#_de5vu3b7gq0r)

[**5. References 4**](#_3435bbxyknrj)

## **Document Revision History**

| **Version** | **Date** | **Ammendment** |
| --- | --- | --- |
| 1.0 | 18.11.2024 | Initial Document |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## **Introduction**

This document was devised to thoroughly explain the details of the EVAReview project. The project’s aim is to streamline surgical planning for a repair method for aneurysms and possibly other operations in the future by providing an easy to use web application, boasting a patient record system with a customisable data overlay. Using deep learning, the application will be able to make suggestions and provide comprehensive support throughout the planning processes.

## **Current System**

Currently, the feasibility of a surgery is discerned by using software that reconstructs the vessels’ arrangement from the cross-sections and the known frame intervals from the CT scan. This is then inspected by the respective person and involves the collaboration of looking at the data, which increases the number of dependencies to multiple separate applications. The varying levels of expertise between the user base also falls onto the users to solve.

## **Proposed System**

## **Overview**

The proposed system aims to alleviate the standard workload of some procedures by performing them in their stead. Allowing users to customize the patient information’s layout and filtering them, saving the setting as a template to change between different views of data. The users will also be able to draw and take notes if they want to show a specific point to their colleagues, being able to quickly toggle between the adjusted and unadjusted view to view the system with less clutter.

There will also be an in-built messaging system along with the ability to be able to share specific views with the other users through their account handles. The user that creates the page for a patient will be able to share their access to their team by adding them to the specific planned operation, either one by one or by setting a team preset previously.

The AI decision support tool will give numbers for a predicted success percentage and give brief explanations of the ways leading to that outcome.

## **Functional Requirements**

| **Requirement ID** | **Requirement Description** |
| --- | --- |
| FR1 | User signup and login |
| FR2 | User authentication |
| FR3 | Add contacts by user handles |
| FR4 | Customization for info display |
| FR5 | Saves for customized information layouts |
| FR6 | Note taking and drawing on page with togglable visibility |
| FR7 | Note sharing |
| FR8 | Allow levels of access to contacts to view a patient |
| FR9 | Communication panel |
| FR10 | Message sending between contacts |
| FR11 | AI suggestions |
| FR12 | Secure patient medical history |
| FR13 | Automatic changelog |
| FR14 | Version reversal from changelog |

## **Nonfunctional Requirements**

| **Requirement ID** | **Requirement Description** |
| --- | --- |
| NR1 | High availability uptime |
| NR2 | Reliable database connection |
| NR3 | Encrypted sensitive data |
|  |  |

## **System models**

## **Scenarios**

## **Use case model**

The only planned actor for the application will be the users and since the patients are not designed to be the users, the use cases are intended to focus on the requirements of healthcare employees.

UC-1: Signup

UC-2: Login

UC-3: Add contact

UC-4: View profile

UC-5: Customize information view

UC-6: Save information view/layout as a template

UC-7: Note taking

UC-8: Drawing

UC-9: Note/drawing deletion

UC-10: Note/drawing visibility toggling

UC-11: Note/drawing sharing

UC-12: Access sharing for the specific surgery or for the whole patient page

UC-13: Message sending

UC-14: Patient creation

UC-15: Patient removal

UC-16: Patient info entry

UC-17: Patient info edit

UC-18: Changelog viewing

UC-19: Revert/recovery to an older version

## **Object and class model**

Briefly:

These info will be kept in a relational database

Doctor -> has id and user handle, password etc.

-> has patients

-> has contacts

-> has pages for their patients

->

Patient-> has medical history -> allergies etc

-> has planned surgery

-> has past surgeries

Surgery-> doctors who partake in the surgery

-> patient

-> description

## **4. Glossary**

CT: Computer Tomography

## **5. References**