

# Software Design Documentation

*EducatiOnline v0.1*

DevIT & Co.

# Software Design Documentation

*EducatiOnline v0.1*

Student	Mail
BEN HAMOUDA, Hibatallah	<a href="mailto:hbenhamouda@et.esiea.fr">hbenhamouda@et.esiea.fr</a>
CHARTON, LEO	<a href="mailto:lcharton@et.esiea.fr">lcharton@et.esiea.fr</a>
DARAMUS RAICA, Raúl Cosmin	<a href="mailto:daramusraica@et.esiea.fr">daramusraica@et.esiea.fr</a>
HADJADJ, Paul	<a href="mailto:hadjadj@et.esiea.fr">hadjadj@et.esiea.fr</a>
MOY, Zacharie	<a href="mailto:moy@et.esiea.fr">moy@et.esiea.fr</a>
YERRO, Pierre	<a href="mailto:yerro@et.esiea.fr">yerro@et.esiea.fr</a>

Task: Documentation  
Subject: CAP Project  
Class: 4°A International Class  
Degree: Master's Degree in Engineering  
Academic Year: 2023 - 2024  
Faculty: Paris



# Index

<b>1 Requirements Specification</b>	<b>1</b>
1.1. Functional Requirements . . . . .	1
1.2. Non Functional Requirements . . . . .	1
1.3. Use Cases . . . . .	2
<b>2 Analysis</b>	<b>3</b>
2.1. Classes Diagram . . . . .	4
2.2. Components Diagram . . . . .	4
2.3. Packages Diagram . . . . .	5
2.4. Deployment Diagram . . . . .	5
<b>3 Implementation</b>	<b>6</b>
3.1. Technologies . . . . .	6
3.2. System Architecture . . . . .	6
<b>Figures Index</b>	<b>6</b>

# 1

# Requirements Specification

## 1.1. Functional Requirements

- **FR-1:** The platform must have functionalities to efficiently dematerialize information file on diseases.
- **FR-2:** The platform should facilitate effective communication between teachers and the medical team.
- **FR-3:** The platform must offer a comprehensive portal with detailed information on symptoms and possible procedures of mental disabilities.
- **FR-4:** The platform must foster effective communication among all involved parts, including teachers, medical staff, and other educational entities.
- **FR-5:** A system is required to collect and analyze user feedback, ensuring continuous improvement of the application.
- **FR-6:** The platform must sign in and sign up functionalities.

## 1.2. Non Functional Requirements

- **NFR-1:** The platform must be efficient in terms of response time and overall performance.
- **NFR-2:** Ensuring the security of stored data is required to protect the privacy of the users.

- **NFR-3:** The user interface must be intuitive and easy to use for teachers and medical staff.
- **NFR-4:** The platform must be compatible with common web browsers to ensure broad accessibility.
- **NFR-5:** The platform should be designed to be scalable, anticipating a potential increase in the number of users and records.

### 1.3. Use Cases

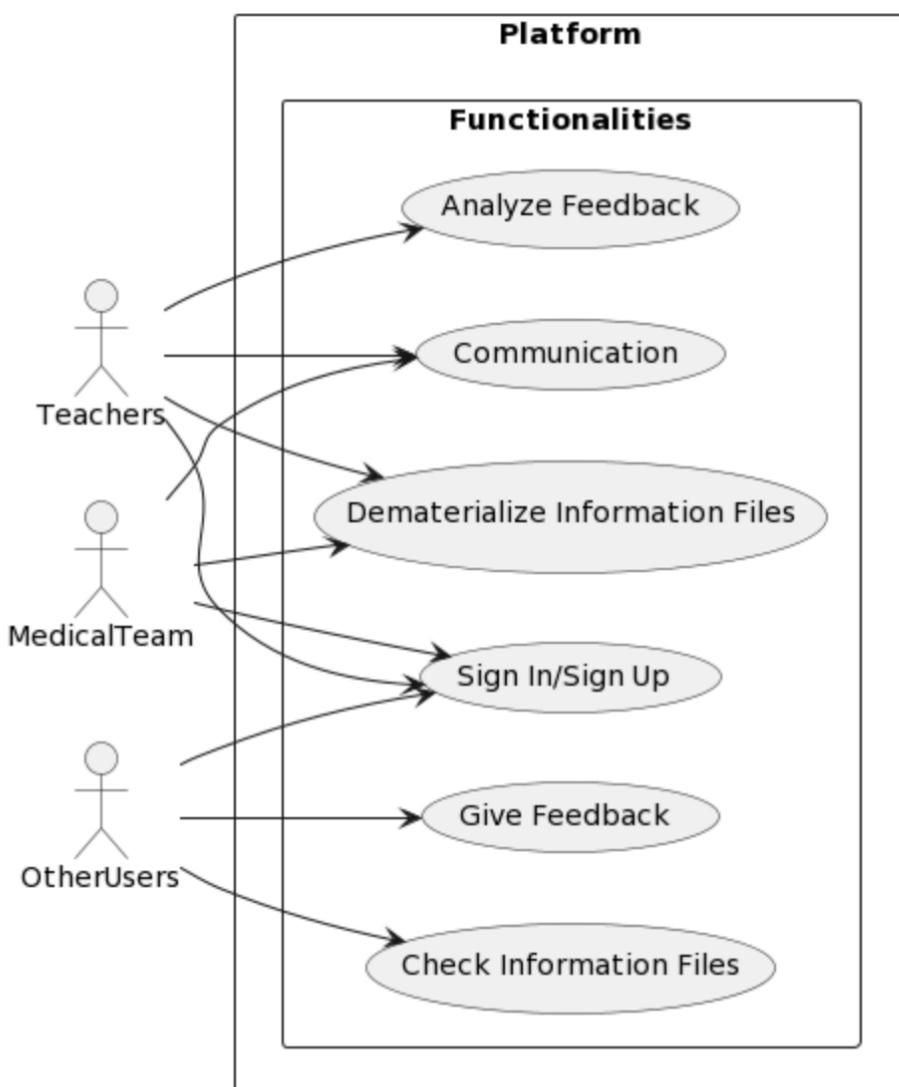


Figura 1.1: General Use Case

# 2

## Analysis

In this section, we will be modeling the software using an abstracted implementation, ensuring implementation independence for future technological changes.

NOTE: In the next section we will address the specific implementation.

## 2.1. Classes Diagram

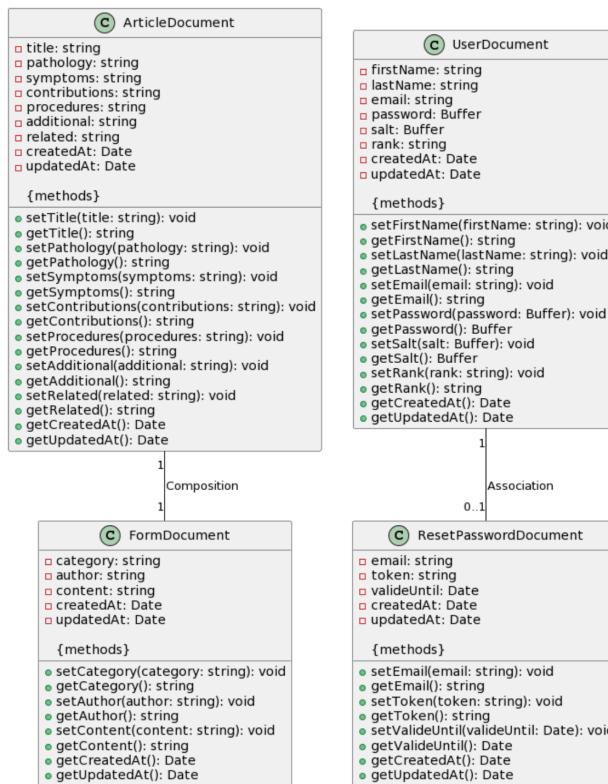


Figura 2.1: UML Class Diagram

## 2.2. Components Diagram

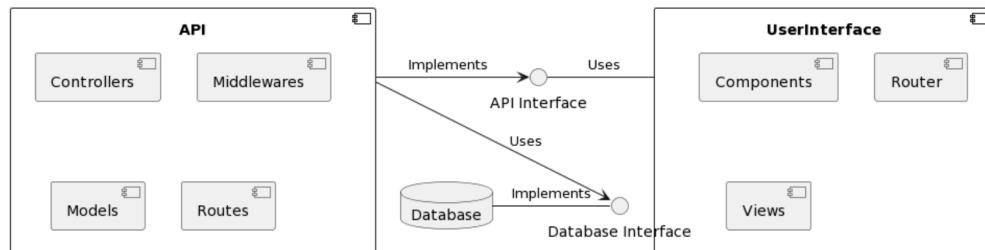


Figura 2.2: UML Components Diagram

## 2.3. Packages Diagram

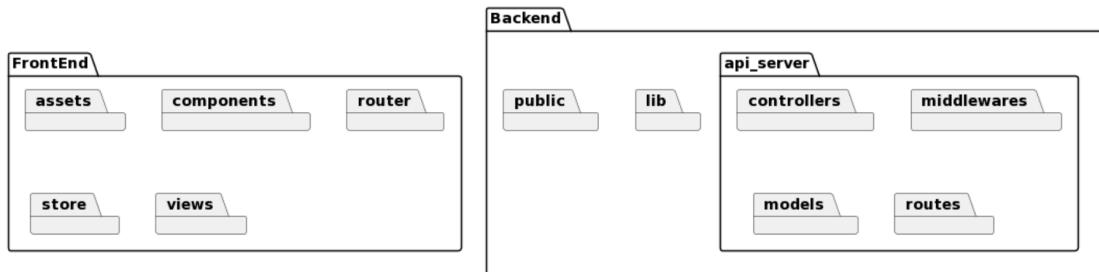


Figura 2.3: UML Packages Diagram

## 2.4. Deployment Diagram

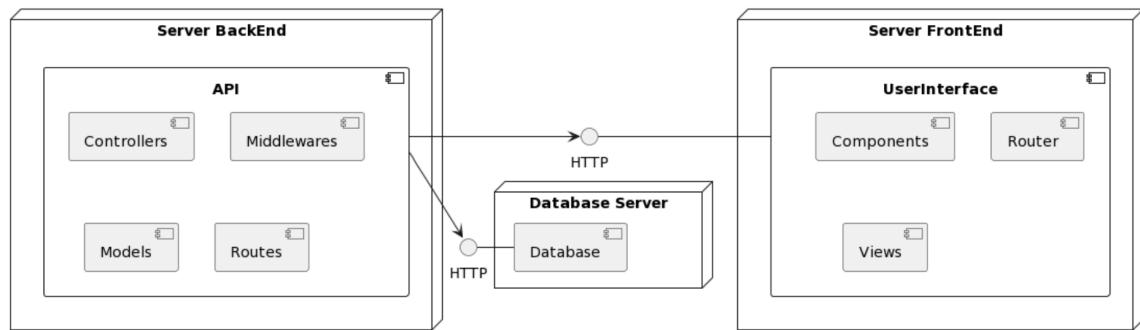


Figura 2.4: UML Deployment Diagram

# 3

## Implementation

### 3.1. Technologies

We decided to choose this Stack of Technologies:

<b>Backend</b>	Node.js, Express.js, MongoDB
<b>Frontend</b>	Vue.js
<b>Deployment</b>	Google Cloud, Docker

Cuadro 3.1: Technologies Stack

### 3.2. System Architecture

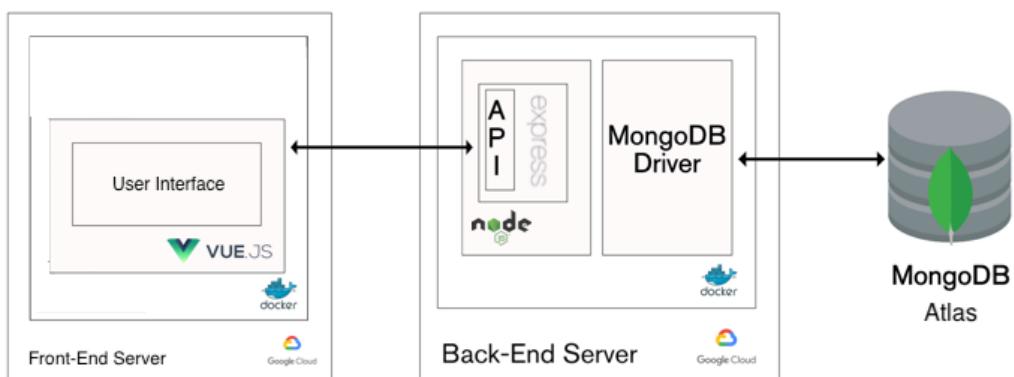


Figura 3.1: General View

# Figures Index

1.1. General Use Case . . . . .	2
2.1. UML Class Diagram . . . . .	4
2.2. UML Components Diagram . . . . .	4
2.3. UML Packages Diagram . . . . .	5
2.4. UML Deployment Diagram . . . . .	5
3.1. General View . . . . .	6