D2.3 - Server and visualization systems for digital health care and a remote patient monitoring system

Work package	WP2
Task	T2.3
Due date	15th July 2023
Submission date	9th January 2024
Deliverable lead	ULL
Version	0.1
Authors	All partners
Reviewers	NUM, NUST

Table of Contents

Executive Summary

The deliverable D2.3 is a deliverable of work package 2 “Development” and is related to the task T2.3 “Data collection, storage, and visualisation”. The main objective of this deliverable is to formalize the needs of each pilot case for server and visualization architecture and to propose an architecture that can be used either for all the pilot cases or for each separate pilot case.

1. Introduction

This deliverable deals with server and visualization systems for digital health, with a primary focus on our three key pilot cases: cardiovascular monitoring in Pakistan (run by NUST and CUST), mobility disorder assessment for elderly patients in Thailand (CMU and MFU), and remote patient consultations and pre-diagnosis assisted with AI in Mongolia (MNUMS and NUM). We break down the specific needs and stakeholders for each case, outlining the actors – health practitioners, researchers, patients or their families –, data collected, and specifications for the visualization platform. Then, we give a prospective architecture for the data analytics and visualization platform and an associated web application. Importantly, the guidelines provided are flexible, recognizing that the dynamic nature of digital health may lead to adaptations in the final products of the pilot cases.

2. Pilot case 1: remote monitoring of cardiovascular patients (NUST)

This pilot case is aimed at developing an ECG data acquisition device that is capable of visualizing, recording, and uploading the signals over the cloud. The raw data is first acquired by the device, then saved locally for further AI-based analysis.

Actors

The actors of this pilot case include the patients diagnosed with cardiovascular chronic disease, and as well as normal individuals. The patients can be from the cardiac care unit or intensive care unit. The electro cardio signals of these actors would be used as raw data for further analysis.

The actors also include the medical staff that monitor the data, as well as the researchers.

Data collected

For this project, the data to be collected is the ECG signals of the patient's using chest leads. These signals contain continuous heart beats at a constant sampling rate that have PQRST intervals. This raw data is then filtered from noise using noise reduction algorithms.

Figure 1: ECG Annotation Tool GUI

Visualization needs

In the visualization needs, there are three types: Realtime Visualization, Annotation Tool, Web Server. For real-time visualization, the raw ECG signals are visualized for continuous stream of data. It also allows for the inference of the signals, once the model is trained. Once the data is acquired and saved, it is then visualized inside the annotation tool in standard ECG format, which is then interpreted by the professionals as shown in Figure 1. For the web part, an admin-controlled portal with access to the data uploaded on the cloud is to be available for visualization. For the doctors, patients, and researchers, the portal will be accessible with secured credentials.

To structure the systems used to create and display visual representations of data and information from the remote monitoring of cardiovascular patients, we propose an architecture as shown in Figure 2. It involves several components, namely:

1. Data source: In this part, real-time ECG signals are collected from sensors. To do the collection, streaming data platform Apache Kafka is used.
2. Data preprocessing: After collecting data, this data needs to be cleaned, transformed, and pre-processed to make it suitable for visualization. To make this preprocessing real-time and meet the need, stream processing framework, Apache Kafka Streams, is used.
3. Data storage: Before storing the data that will be displayed to the user, real time analytics need to be processed to derive insights from the data and provide recommendations. Then, this data will be stored in a fast-access data store like Redis or Apache Casandra for quick retrieval.
4. Data visualization: To create dynamic visualizations of data in graphs, different libraries can be used, like D3.js or Plotly. The visualizations will be deployed in a web interface using various technologies, like Node.js for back-end and React for the front-end where doctors and patients can access and interact with the real-time visualizations.

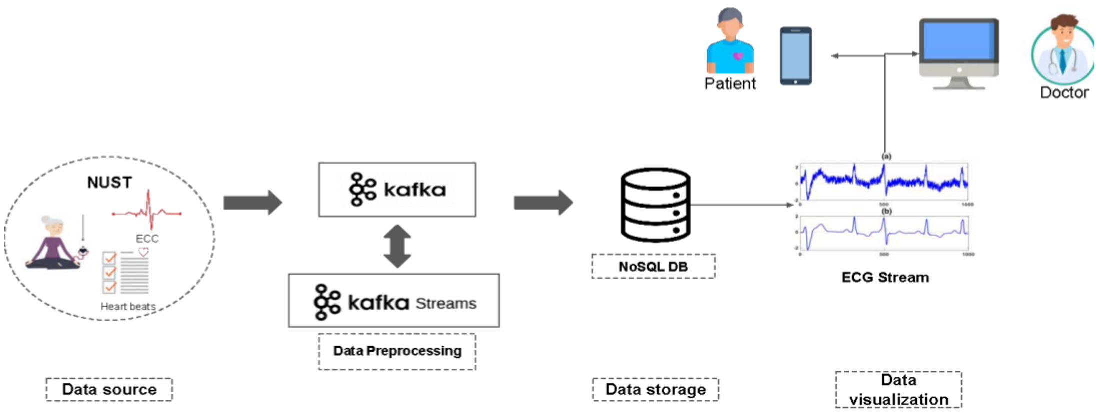


Figure 2: Proposed Architecture Diagram

3. Pilot case 1: remote monitoring of cardiovascular patients (CUST)

This pilot initiative revolves around harnessing the capabilities of Internet of Things (IoT) devices in conjunction with artificial intelligence (AI) for remote patient cardiovascular monitoring. The primary aim is to develop into the potential of IoT equipment and AI algorithms to collect and analyze cardiovascular data from a distance through sensor technology. The ultimate objective is to elevate patient care by promptly identifying potential cardiovascular concerns and enabling well-informed medical interventions.

Actors

This pilot involves patients, doctors, an AI system, and an administrator. Patients share data through IoT devices, doctors use remote data and AI predictions, the AI system processes data, and an administrator manages user registrations. The goal is an efficient remote monitoring system to enhance patient care.

Data collected

In this project, the collected data comprises of sex, body temperature, age and ECG signals that will be obtained from patients using chest leads. These signals encompass a continuous sequence of heartbeats, each with PQRST intervals, all sampled at a consistent rate. The raw data undergoes noise reduction through specialized algorithms, rendering it suitable for use by machine learning algorithms for predictive analysis.

Visualization needs

For visualization purposes, the patient has access to their profile and can review the doctor's recommendations. The doctor, on the other hand, can comprehensively examine patient data, including monitored vital signs such as electrocardiogram (ECG) readings, body temperature trends, and Arrhythmia predictions categorized as normal and abnormal. These predictions offer detailed distinctions between normal and abnormal results, aiding the doctor in making informed assessments and providing personalized care for the patient. This comprehensive information equips the doctor to make well-informed decisions regarding patient care. Figure 3 shows some visualizations of the data.

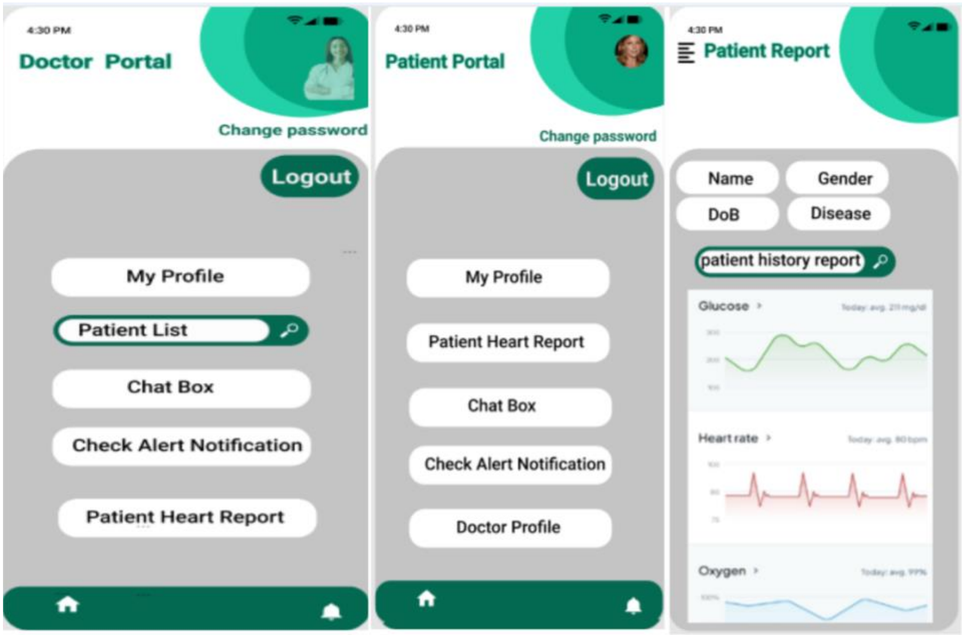


Figure 3: Patient and Doctor Dashboard visualization

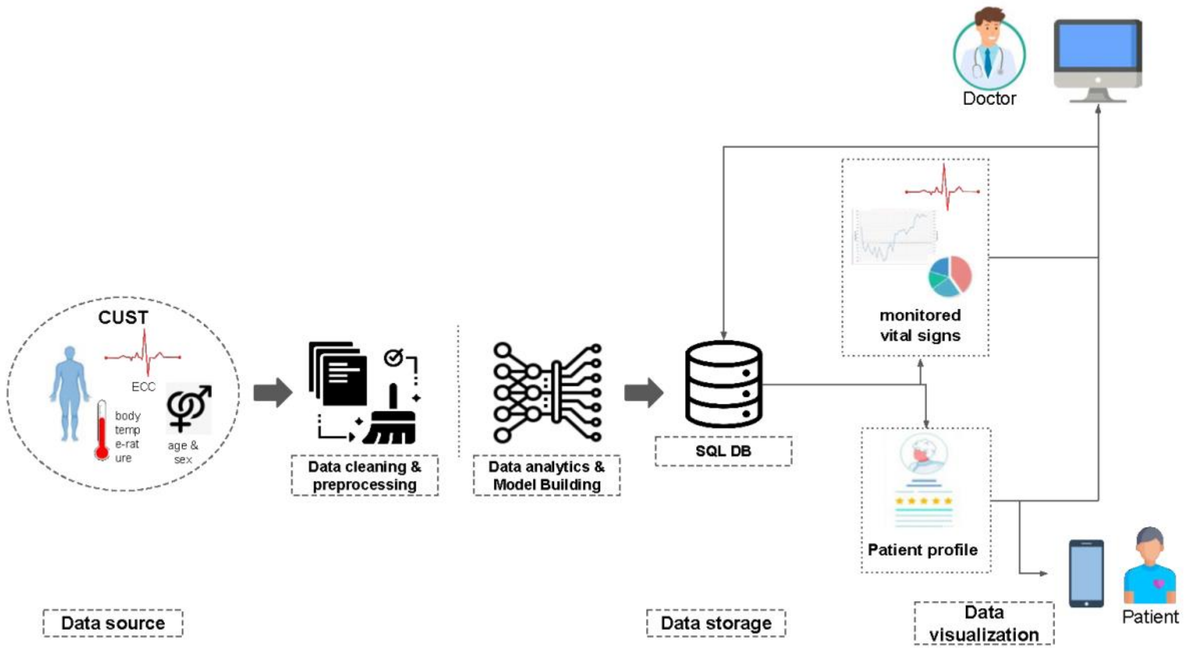


Figure 4: Architecture design.

To structure the systems used to create and display visual representations of data and information from the remote monitoring of cardiovascular patients, we propose this architecture. It involves several components, namely:

1. Data source: Patient data, including sex, body temperature, age, and ECG signals, are collected from monitoring devices.
2. Data preprocessing: After collecting data, this data needs to be cleaned, transformed, and pre-processed to make it suitable for visualization.
3. Data analytics: Before storing the data that will be displayed to the user, data analytics and AI models need to be integrated to derive insights from the data and provide recommendations.
4. Data storage: The result of data analytics will be stored in database like SQL or PostgreSQL.
5. Data visualization: Dynamic visualizations of patient data are generated using visualization libraries like D3.js or Plotly. These visualizations are designed to be informative and interactive.

A user-friendly web interface is developed using technologies like Node.js for the back-end and React for the front-end. This interface allows doctors and patients to access and interact with the visualizations.

4. Pilot case 2: remote monitoring of mobility disorder patients (CMU and MFU)

This pilot case focuses on risk assessment for fall and monitoring of the elderly through smart watches and other IoT equipment. CMU focuses on physical training, while MFU focuses on risk assessment through physical performance evaluation.

Actors

The actors for this pilot case are the elderly, their caregivers, the health practitioners, the care unit, and the researcher that will use the data.

Data collected

Both risk assessment and physical monitoring require a general profile of the elderly (age, sex, weight, height, physical activity readiness PAR-Q+, BMI). For risk assessment through a physical evaluation, specific data is collected (testing times for the chair stand test, balance test, gait speed test and the time up and go test). For the physical training, the specific data is exercise data (frequency, intensity, time and duration, type of exercise) and the heart rate of the elderly during the effort.

Visualization needs

The visualisation need depends on the actor. In CMU pilot case, there are 3 main actors i.e., elderly, care giver, and care unit. The system architecture is depicted in Figure 5.

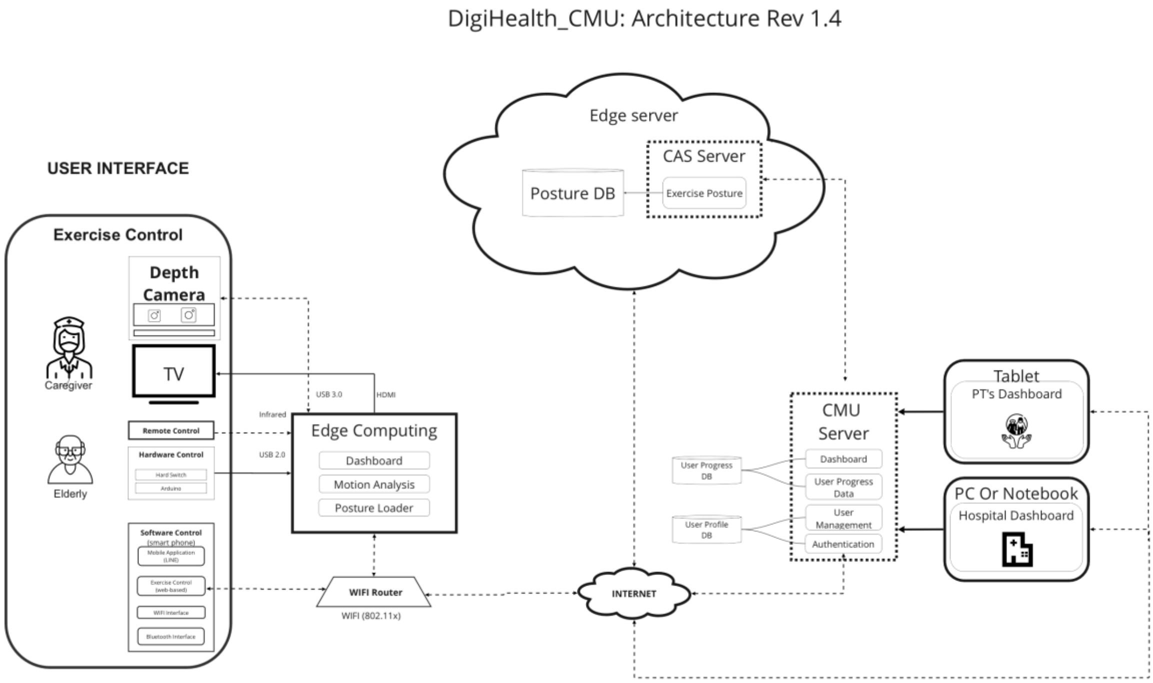


Figure 5: CMU Architecture.

For the elderly population, it is imperative to establish a system that grants them seamless access to their personal health data, encompassing crucial information such as exercise scores, historical records, comprehensive statistics, and real-time heart rate data.

In the case of caregivers, it becomes even more vital to provide them with comprehensive access to the elderly individuals' health profiles, exercise history, and ongoing heart rate monitoring. Additionally, caregivers should have the capability to review and analyze testing data and assessment scores, thereby facilitating informed decisions in caregiving. (See Figure 6)

Meanwhile, the care unit, as the central hub of management, must be equipped with a sophisticated dashboard. This dashboard should not only offer a succinct summary of each elderly individual's profile and exercise data but should also serve as a gateway to in-depth, granular health metrics. This comprehensive access enables the care unit to provide top-tier healthcare services efficiently. (See Figure 7)

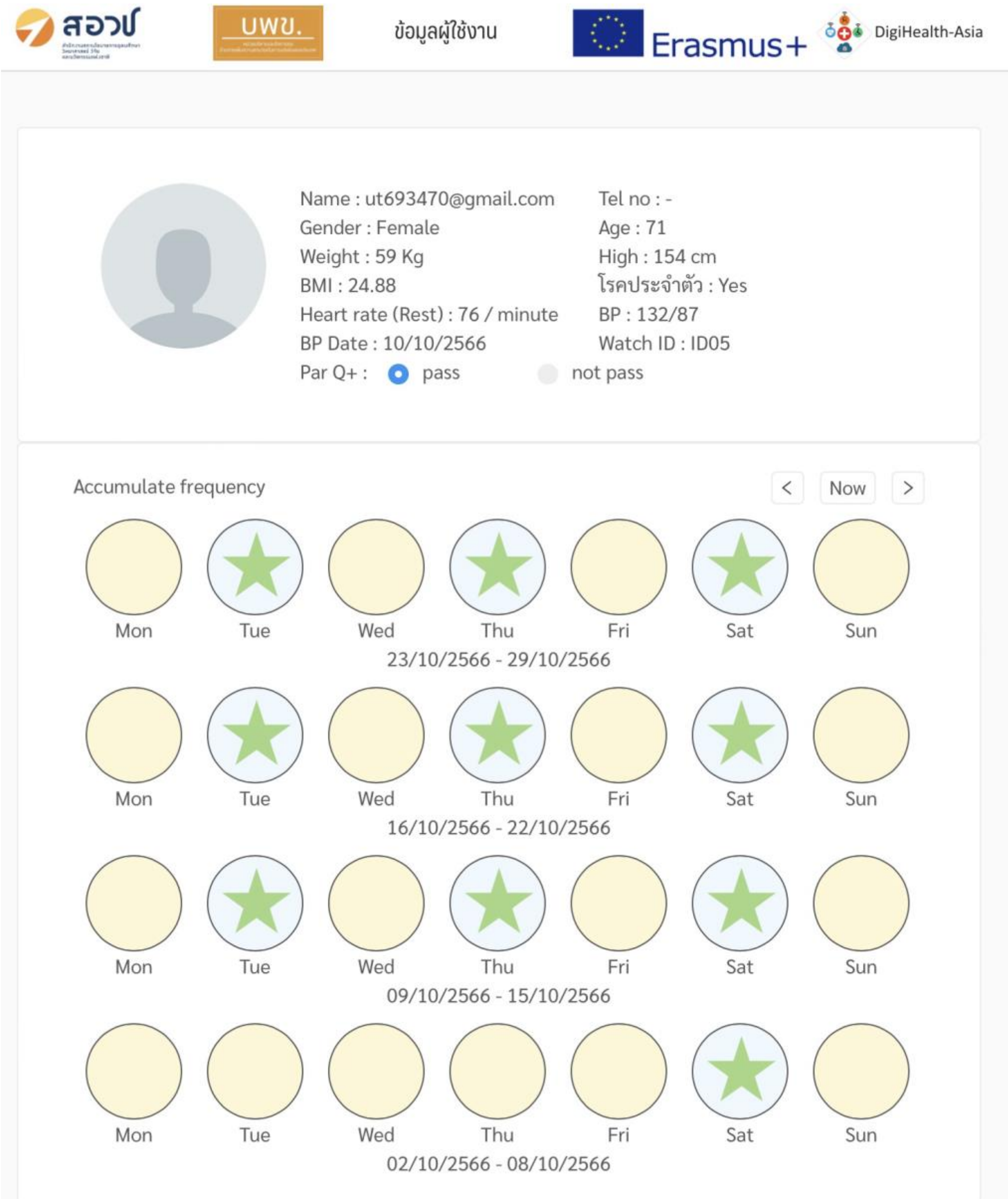


Figure 6: Caregiver Dashboard

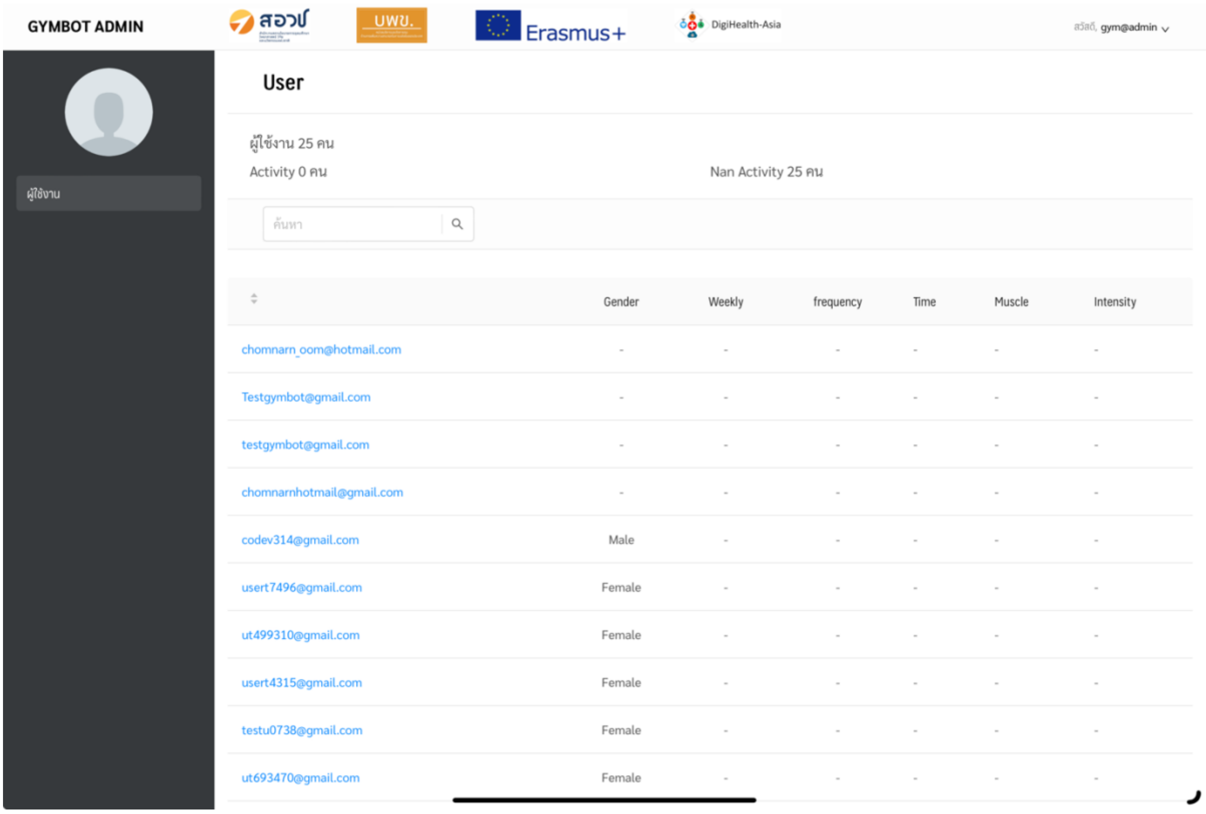


Figure 7: Care Unit Dashboard

Finally, researchers must have access to a dashboard with summary statistics, and anonymous data for profiles, testing and exercise data.

For MFU, the visualization responds to each actor's need presented in the system architecture as shown in Figure 8.

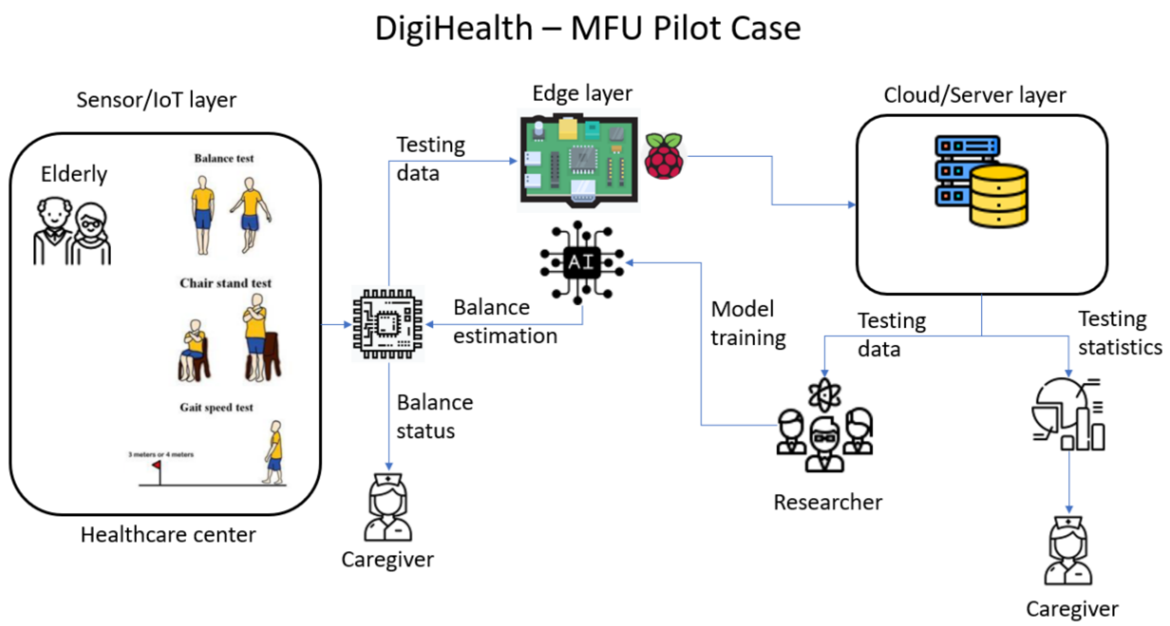


Figure 8: MFU architecture.

Three principal actors i.e., the elderly, caregiver and researcher interact differently with the system. The elderly participate in the physical test and have no need to understand or access to the raw testing data. Meanwhile the raw data will be collected and pre-processed at the edge layer and in case of the machine learning model is ready, the inference or the balance estimation will be returned to the caregiver to evaluate, inform and treat the elderly subsequently. In addition, the edge layer will pass the pre-processed data to the server layer for storage and visualization. The caregiver can visualize the testing statistics belong to his/her patients through a web interface. The researchers would be also permitted to query the server's data anonymously for retrieving testing data for training machine learning model and updating it to the edge layer.

MFU provides a web system for visualization. Some examples are presented in Figure 9 and 10.

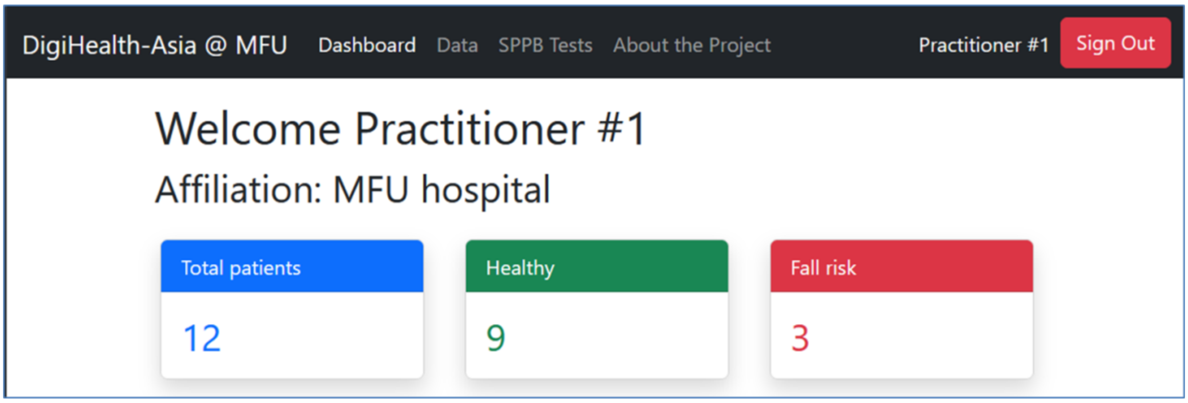
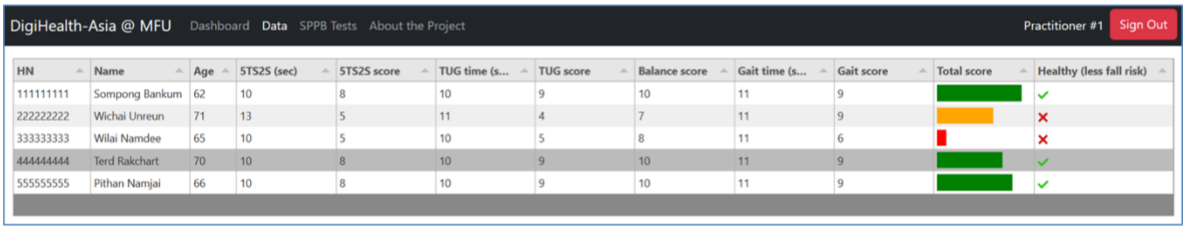


Figure 9: Caregiver Dashboard



HN	Name	Age	5TS2S (sec)	5TS2S score	TUG time (s...	TUG score	Balance score	Gait time (s...	Gait score	Total score	Healthy (less fall risk)
111111111	Sompong Bankum	62	10	8	10	9	10	11	9		
222222222	Wichai Unreun	71	13	5	11	4	7	11	9		
333333333	Wilai Namdee	65	10	5	10	5	8	11	6		
444444444	Terd Rakchart	70	10	8	10	9	10	11	9		
555555555	Pithan Namjai	66	10	8	10	9	10	11	9		

Figure 10: Testing Statistics

To structure the systems used to create and display visual representations of data and information from the remote monitoring of mobility disorder patients, we propose this architecture. It involves several components, namely:

1. Data source: Various data from both CMU and MFU are collected including a general profile of the elderly, exercise data, and the heart rate of the elderly during the effort.
2. Data preprocessing: After collecting data, this data needs to be cleaned, transformed, and pre-processed to make it suitable for visualization.
3. Data storage: Before storing the data that will be displayed to the user, data analytics need to be processed in order to derive insights from the data and provide recommendations. Then, this data will be stored in database like SQL or PostgreSQL.
4. Data visualization: To create dynamic visualizations of data in graphs, different libraries can be used, like D3.js or Plotly. The visualizations will be deployed in a web interface using various technologies, like Node.js for back-end and React for the front-end where doctors and patients can access and interact with the real-time visualizations.

5. Pilot case 3: Remote consultation of patients (NUM and MNUMS)

This pilot case focuses on remote consultation of children for oral hygiene and dental health issues. Regional medical centers must be able to collect data (questionnaires and intraoral camera images for teeth) for patients, that will first be processed by the machine learning model for a general evaluation and score. Depending on this score, the child will then be provided with directions to the dentist or personalized advice.

Actors

In this pilot case, the participants include a child undergoing an oral examination and their parents or caregivers, who assist in completing a survey. Additionally, there is a dentist or researcher (possibly a teacher) responsible for taking intraoral pictures, a health practitioner receiving pre-diagnosed data, and a researcher with access to the collected data.

Data collected

In this pilot study, the gathered data includes the child's profile (including age, gender, and city) and their parents' information (age, income, education). It also encompasses details about any reported teeth-related issues, self-assessment of dental hygiene, evaluation of parents' or caregivers' oral health knowledge and attitude, the specific oral condition of each child, and images of their teeth.

Visualization needs

For visualization, the child and their parents need to have access to the profile, the DMFT (Decayed, Missing and Filled Primary Teeth) score of the child, the questionnaire score (output of the first model), and the diagnosis of the practitioner.

The dentists need to have a summary of the child profile and DMFT score, together with the full profile, questionnaire answers and images.

Researchers should be able to access anonymous children's profiles, questionnaire answers and images.

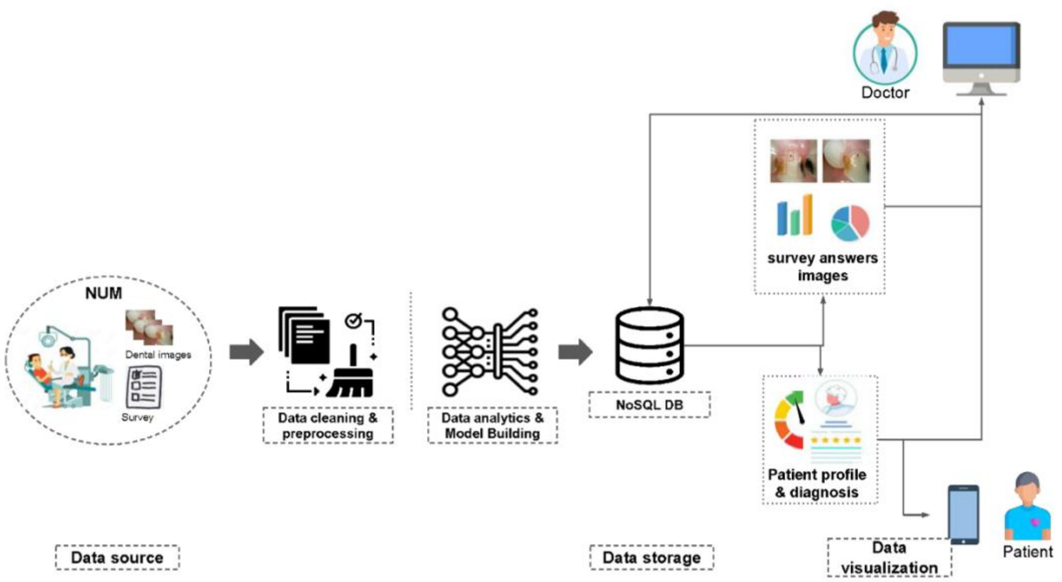


Figure 11. Data flow for the Pilot study.

To structure the systems for remote consultation of patients and create visual representations of the data, we propose this architecture as shown in Figure 11. It involves several components, namely:

1. Data source: Patient data, including various questions about oral health and images, is collected from monitoring devices.
2. Data preprocessing: After collecting data, this data needs to be cleaned, transformed, and pre-processed to make it suitable for visualization.
3. Data analytics: Before storing the data that will be displayed to the user, data analytics and AI models need to be integrated to derive insights from the data and provide recommendations.
4. Data storage: The results of data analytics will be stored in databases like SQL or PostgreSQL and Amazon RDS for Images.
5. Data visualization: Dynamic visualizations of patient data are generated using visualization libraries like D3.js or Plotly. These visualizations are designed to be informative and interactive.

A user-friendly web interface is developed using technologies like Node.js for the back-end and React for the front-end. This interface allows doctors and patients to access and interact with the visualizations.
To organize the components of a web-based application, a web architecture has been designed and is shown in Figure 11. This architecture consists of multiple layers and components that collaborate to deliver web services and content to users. Below is an overview of the key elements within this web architecture:

5. Client-Side Components:
- Web Browser: The user's web browser acts as the client, rendering HTML, CSS, and JavaScript to display web content
- User Interface (UI): The UI layer handles user interactions by displaying web pages and providing a means for users to interact with the application.
6. Server-Side Components:
- Web Server: The web server receives and responds to HTTP requests from clients.
- Application Server: The application server executes the application's logic, processes requests, and communicates with databases or external services.
- Database Server: The database server stores and manages application data.
7. Communication Protocols:
- HTTP/HTTPS: Hypertext Transfer Protocol (HTTP) is the foundation of web communication. Secure HTTP (HTTPS) adds encryption for secure data transmission.
- WebSocket: WebSocket allows bidirectional, full-duplex communication between clients and servers, suitable for real-time applications.

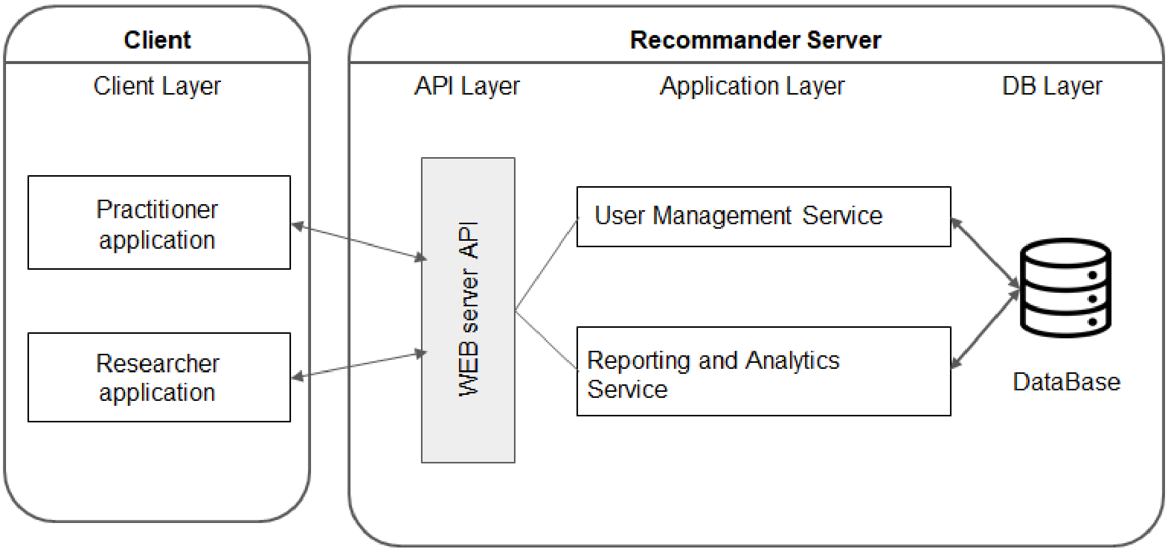


Figure 12: Web architecture.

6. Conclusion

This deliverable, focused on addressing critical server and system requirements for effective visualization and data collection in the specified pilot cases, takes a unique approach for each case. Since each pilot case has very distinct actors, specific needs, and various types of AI models and functions, they have followed different development trajectories. It was thus deemed more effective to have three separate platforms for visualizing and managing the data associated with each pilot. Practical technical solutions have been tailored to the nuances of individual scenarios, emphasizing the importance of a comprehensive strategy.

Furthermore, proof of concept has materialized into tangible developments. For pilot cases 2 and 3, we've successfully implemented web applications, demonstrating their potential to enhance accessibility and user interaction. Taking a step further and recognizing the diverse preferences and needs of the health practitioners, a dedicated mobile application has been crafted for pilot case 1. These applications not only showcase our contributions within the pilot cases, but they also exemplify our dedication to creating user-friendly, accessible tools that integrate into the dynamic landscape of healthcare practices.

The deliverables associated with each pilot case – D2.4, D2.5, D2.6 – will include all the details necessary for their specific platforms.