Department of Economics

Democritus University of Thrace

Interreg
Greece-Bulgaria
eHealth Monitoring
European Regional Development Fund

EUROPEAN UNION

D4.3.2: APPLICATION/ SOFTWARE DEVELOPMENT

(Application/Software Development)

«Healthcare Monitoring System Development»

Reporting period: 20/09/2018 - 30/08/2020

WP 4: Joint Monitoring System

project

IMPROVING HEALTHCARE ACCESS THROUGH A PERSONAL HEALTH MONITORING SYSTEM

Vasileios Mitsios

August 2020

The project is implemented in the framework of INTERREG V-A "Greece-Bulgaria 2014-2020" Cooperation Programme and is co-funded by the European Regional Development Fund (ERDF) and by national funds of the countries participating in the Programme

http://www.ehealthmonitoring.eu/

Summary for the deliverable

This deliverable «Healthcare Monitoring System Development» is part of the project APPLICATION/ SOFTWARE DEVELOPMENT within the frame of WP 4 Joint Monitoring System of the project IMPROVING HEALTHCARE ACCESS TROUGH A PERSONAL HEALTH MONITORING SYSTEM, according to our contract (14/09/2018 – Ref. No: 44955) and is being implemented within the framework of the Programme INTERREG V-A Greece – Boulgaria 2014-2020.

This deliverable aims to highlight the activities performed within the period 20/09/2018 - 30/08/2020 concerning the design, development and pilot operation of the Integrated Health Monitoring System.

The deliverable describes the system as a whole and focuses on issues that affect its subsystems horizontally. Special reference is made to the platform subsystem that is installed and operates on cloud computing infrastructure and supports the specific applications of the system as described in the following chapters:

- Web App
- Mobile App (Android)
- Cloud Infrastructure

Specifically, the user types and supported usage scenarios are recorded. In addition, technical and security features of the application are recorded.

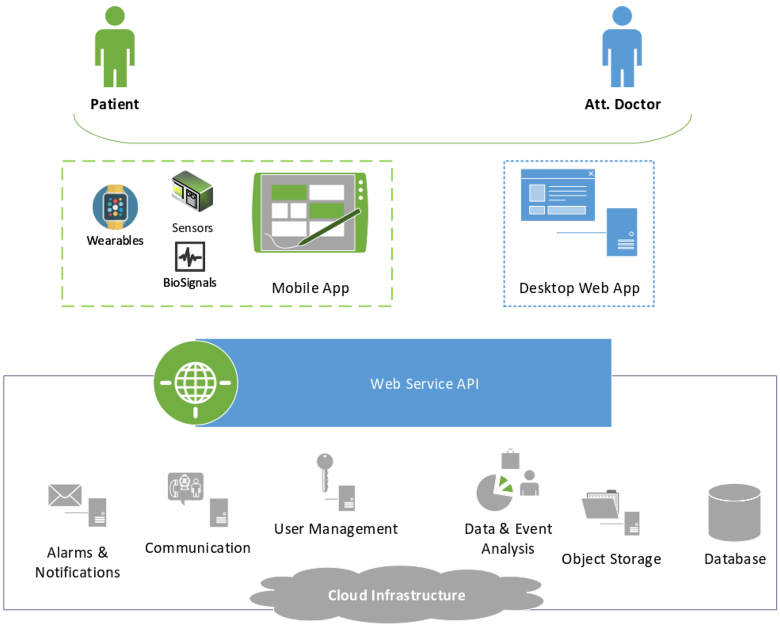
Table of Contents

1 Introduction

The Integrated Health Monitoring System, which will be designed within the frame of the **“EHEALTH Monitoring”** project**,** consists of four main sub-systems that realize the required functionalities, as well as a cloud back-end platform, which supports all other subsystems. These subsystems have been designed and developed utilizing a rich set of state-of-the-art technologies and tools, in order to secure optimal levels of robustness, security and extendibility and the finest user-experience.

The system follows a service-oriented architectural design, exploiting the advancements and flexibility of cloud offerings, and implements modern UIs for all types of users. Cloud Computing allows for ubiquitous access to shared resources and common infrastructure, offering services on-demand, serving the constantly changing needs of the health-centric digital services. To that end, is being developed an integrated system for patient monitoring at home, utilizing Cloud Computing concepts and tools for data managements and analysis. The proposed solution focuses on the system decision support functionality, which is utilized within the smartphone app for initial assessment, as well as in the Cloud.

A combination of Java and JavaScript technologies and frameworks are used for implementation and communication of the various application components and services.



The main functionality of the platform is recording and analysis of biosignals. Java technologies have been utilized for development and the functionality is offered via Web Services. Similarly, synchronization and central administration functionalities of the application subsystems are provided. The following features are supported:

- ✓ User account management
- ✓ Data analysis
- ✓ Data storage
- ✓ Reminders and notifications to patients and other end-users in cases of measurements out of normal range, detection of errors, measurement and medication schedule, etc.
- ✓ Videoconferencing functionality based on the WebRTC protocol, which is utilized both for the Web app and the mobile app.

As mentioned above, the proposed integrated health monitoring system consists of four main sub-systems that realize the required functionalities. These subsystems have been designed and developed utilizing a rich set of state-of-the-art technologies and tools, in order to secure optimal levels of robustness, security and extendibility and the finest user-experience. To that end, the proposed solution follows a service-oriented architectural design, exploiting the advancements and flexibility of cloud offerings, and implements modern UIs for all types of users.

A combination of Java and JavaScript technologies and frameworks are used for implementation and communication of the various application components and services. A cross-layer technology that has a key role on the realization of the communication and videoconferencing functionality is WebRTC[1], which is used both for the desktop and mobile applications.

1.1 Web App

The web app includes the core functionality for interacting with all user types. In contrast to the mobile application, which is lightweight and simplified so as to ease the interactivity with the elderly users and patients, the web application has rich user interface for configuring the user and application parameters, and also for visualizing the biosignals and health records. WebRTC based communication is also provided through the web application.

The following technologies were utilized for the development of the application:

- ✓ **Java Programming Language** for the development of the Business Logiv of the Web app and communication with the database.
- ✓ **Apache Tomcat Servlet Container** for the deployment and operation of the application based on Java technologies.
- ✓ **Apache Web Server - Load balancer** for for seamless operation of the application even in cases of high load and number of simultaneously connected users.

From the end-user's point, the only requirement is the use of a web browser that supports the WebRTC protocol, which can be any of the following:

---

[1] http://www.webrtc.org/

- Google Chrome >= 65
- Mozilla Firefox >= 60
- Opera >= 54

1.2 Mobile App (Android)

For a simplified, flawless and ubiquitous access to the system, a mobile app for Android devices has been developed. Besides the core functionality for communicating with the backend, the mobile applications allow for the integration of Bluetooth devices, sensors and wearables. The native android application targets (API) Android 21 / Lollipop. The application incorporates the required functionality for communicating with the cloud-based platform services and also acts as a platform gateway to the various Bluetooth devices (biosensors και wearables).

1.3 Cloud Infrastructure

The cloud infrastructure of okeanos has been configured to host the platform components and services.

2 Methodology

The application development methodology follows the Design-Development-Control model, also known as the Prototype Method. According to this model, a prototype of the software system is initially developed that contains only the basic features of the final system. Using this version gives you experience that is used to improve the version. This method is mainly used when the requirements are difficult to determine or when the latter may change significantly during implementation. The software development process using the prototype method consists of four stages:

- Clarification of the basic requirements of the user
- Development of the original system prototype
- Use of the original system to reshape the user requirements
- Review and improve the original system

The work related to this deliverable is part of the prototype development stage, but also of the requirements redefinition stage.

3 Roles and Supported Use Cases

3.1 Roles, Requirements and Functionalities

The user roles are described in the following table:

Type	Role	Description
User	Patient (χρήστης)	The main user of the system (recipient of the service)
User	Contact	A user that can communicate with Patients. Every Patient may have more than one Contacts.
User	Attending doctor	Doctors
Admin	System Administrator	Technical and administrative role. Has access to all reports/MiS.

The system supports the following user roles:

1. Ordinary user (relative): This user has the ability to add any type of user to his contacts, view photos or biosignals and make calls.
2. Doctor: This user, together with the functions of the ordinary user, has the ability to monitor his patients-contacts, to take notes, to see their biomarkers and to make measurements for them.
3. Patient: This user, along with the functions of the ordinary user, has the ability to take measurements from sensors and choose which of his contacts will have access to his measurements and history.

The application, depending on the type of each user, offers different functionalities.

3.1.1 General Requirements

The general application requirements are the following:

Application	**Role**	**Functionalities**
Web app	Patient	• Contact management • Video conferencing with contacts • Account and personal info management • Reminders management • View PHR and add content • View recorded biosignals • Manage medications and relevant settings
	Attending doctor	• Account management • View patients

Application	Role	Functionalities
		• Video conferencing with patients • Manage medications and relevant settings (for patient) • View patients' biosignals • View patients' PHRs and add content
	Contact	• Contact management • Video conferencing with contacts
Mobile app (Android based)	Patient	• Video conferencing with contacts • Reminders management • View PHR • Record, view and upload biosignal measurements
	Contact	• Contact management • Video conferencing with contacts

3.1.2 Non-functional Requirements

Cloud infrastructure will be used for platform hosting and support of the various sub-systems. Exploiting **Cloud offerings** allows for:

- ✓ **Improved resources management**: Preparing services which may later not cover user requirements can be avoided. Cloud computing reduces costs and maximize utility, since resources are provided only when needed.
- ✓ **Flexibility**: All systems and software remain constantly available.
- ✓ **Accessibility**: Cloud computing allows ubiquitous access to applications and data, with encryption security, via the Internet. When connection to the Internet is not available, bandwidth requirements can be covered even with a 3G connection.
- ✓ **Cooperation**: Accessibility of applications and data over the cloud facilitates cooperation between different components, since applications can have simultaneous access to data.
- ✓ **Disaster recovery & Business continuity**: Recent research indicates that approximately 90% of businesses do not have adequate plans for managing total or partial loss of their computer infrastructure, in order to ensure uninterrupted operation. Cloud computing allows for automation of disaster recover processes through the use of backups, as well as maintaining servers that can operate as others' images.

3.1.3 General Functionalities

- Login & Register: The application, for the identification or registration of the user provides separate screens, which achieve this purpose. On the Login screen, the user has the ability to either enter his details (e-mail and password) and log in, or reset his password via a dialog, or be led to the Register screen where by filling in his details he can to register on the platform.

- Dashboard: Dashboard is the initial and main menu of the application. Through this, the user can go both to the screens that provide the functions of the application, and (via the left slide menu - drawer) and to the settings of his profile.
- Photos & Videos: The Photos screen provides the user with a complete gallery, in which he can, by choosing between his camera or his library, upload and view his photos and videos, as well as the photos or videos of his contacts. The photo upload service also provides image editing tools (crop etc).
- Contacts: The Contacts function provides the user with a contacts management system. The user has the ability to request, remove or accept a contact. Also, from this screen the user has access to the profile of each contact, as well as to functions such as video calling.
- Profile: The application allows each user to maintain a profile. The profile is accessible from the contacts screen and includes options such as video calling or chat but also access to information such as biomarkers, notes (only for doctors), phone, email, etc. To edit the profile the user must go to the settings menu slide menu-drawer or click on his profile picture in the same menu.
- Video calling: The application supports video calling between its users. By selecting the appropriate option, the user is led to the video call screen, which supports all the necessary functions for a video call: mute the microphone, turn off and rotate the camera, free slide the personal camera in the screen, function in the background, receive a call even with the application is closed (after the appropriate licenses have been accepted).
- Notifications: Notification management function (for any missed calls or for new contact requests etc).
- Biosignals: The biosignal projection function is provided for a patient. This screen allows the (authorized) user to see the measurement history (pressure, pulse, etc.) of all supported measurement types with tables and diagrams for a period of time that he can specify.

3.1.4 Criteria for Infrastructure Design & Selection of Development Tools

The main criteria for the Design of the Infrastructure are the following:

- ✓ **Functionality:** The main focus of the design process is to satisfy all functional and technical requirements (communication between components, communication with end-user devices, etc.).
- ✓ **Costs**: Operations and initial deployment cost. Open-source software bears no intial deployment cost (software licences for server web apps, database, etc.), as well as maintenance costs. Distributions are provided for free even for commercial use, while the

open-source community offers continuous support through new versions (often with the capability for automated installation).

- ✓ **Application Support**: Support of applications – development tools, in terms of updates, as well as on operational level, by official guides or the user-developer community was and important consideration in the choice of development tools.
- ✓ **Scalability**: Both the architectural design and the development tools should allow the addition of new modules, as well as new functionalities, without the need for redesign or radical changes in implementation.
- ✓ **Security:** The hosting and operational environment should be secure, incorporate strict user and access-rights management, and support on the application level state-of-the-art mechanisms for user authentication and secure data transfer. The following aspects have also been considered:
    - securing integrity and availability of information
    - protection of stored and processed personal data, by methodically selecting and applying technical measures and organizational processes.
    - composition of the required documents for terms of use and privacy policy
- ✓ **Electronic Health Record Specifications**
    - Cloud technologies support and web-based access.
    - Incident based structure and visualization.
    - Patient authentications using their SSN.
    - Classified access/recording, depending on each role’s access rights
    - Stored data
        - Demographics.
        - Medical History: diseases, medications, allergies, etc.
        - Biosignals
        - Other exams if available (in case of examination at home or if provided by a user)
        - ICD-10 diagnosis, if available, enriched with free text.
- **Relevant Legislation**
    - Institutional and legal framework in force (directive 95/46/EK, protection of personal data Law 2472/97, protection of personal data in the telecommunications sector Law 2774/99).
    - The General Data Protection Regulation (EU) 2016/679 (GDPR).
- **Usability Requirements**
    - Convenience and user-friendliness. Common theme for all applications, unified and consistent functionality.
    - Special care for elderly user needs.

3.1.5 Functionalities for Doctors

- Notes: From the patient profile, the user has access to the submission and editing functions. The user selects a note type and then enters the note text. The notes are personalized, ie each patient has his own. They are not accessible to the patient.
- Questionnaires: The doctor has the ability, through the patient profile screen, to run a questionnaire and record the patient's answers in the application. This questionnaire appears in the application, the doctor reads the question aloud to the patient, the patient answers and the doctor fills in the answer.
- Manual Measurements: It is possible for the doctor to manually add measurements (from the supported ones) for a patient, through the Biosignals screen. The measurement is made with a sensor or instrument and then the value is entered manually (eg 99 for SpO2) in a suitable field of the screen.

3.1.6 Functionalities for Patients

- Measurements: The basic function of patient type users. Users have the ability to compose their mobile or tablet with a variety of devices, the list of which is constantly updated. With these devices they can perform both real-time measurements and synchronization of measurements - for devices that support this feature. These measurements are then stored in the cloud, allowing both doctors or relatives as well as themselves to be able to monitor them at a later time. The measurement screen has the same shape as the screen seen by the doctor / relative.
- Questionnaires: Similar to the function of the doctor's questionnaires, with the difference that now the patient chooses, answers and completes the test. Located on the left slide menu-drawer.

4 Information Security

Data security and the prevention of malicious and illegal use is essential for an application. The application uses the following mechanisms and technologies, while incorporating relevant procedures.

4.1 Technologies

- Sembast: With the Sembast library the local data of the application is saved. This library supports encryption through a codec which is generated by a code given by the developer.
- Shared Preferences: The Shared Preferences library uses NSUserDefaults on iOS and SharedPreferences on Android. It stores more sensitive data (such as access token, login token) etc.

4.2 User Authentication

The registered user in order to be identified must enter his e-mail and password during Login. The following is the identification process:

- Login Token: After entering the above, and in case of a valid combination, the server generates a unique login token key, which will be used to connect the user to the application. This key is an alphanumeric character sequence. It is stored in a safe place of the application as it is needed for the future times when the user will log in (auto login).
- Access Token: Having the login token, in order to be able to make calls to the rest api of the application, an access token must be given. This token is generated through a specific call of the api and lasts one (1) hour, giving as an argument the login token. After one hour, an access token must be requested again. If a rest call returns an appropriate error message, the user is automatically logged in, and the invalidate tokens. The whole system with tokens has the logic that the login token remains constant throughout the user login (even when the application calls and reopens), while the access tokens are renewed when the application is opened.