E-Health Service
Requirements Definition

Version 2.0

Revision History

Date	Version	Description	Author
2012-10-29	0.1	Partially added Use Cases and System Requirements.	Petar Kekez, Bojan Kosanović, Vedran Šikić
2012-11-1	0.2	Changes made to Use Cases and System Requirements.	Petar Kekez, Bojan Kosanović Vedran Šikić
2012-11-2	1.0	Revised document format and added Use Case Diagram and Future Development part.	Gregorio Perego Stefania Pezzetti Bojan Kosanović
2012-11-3	1.1	Edited Use Case Diagram and some Use Case.	Gregorio Perego Stefania Pezzetti
2012-11-9	1.2	Add Emergency Levels Table	Stefania Pezzetti Marko Kelava
2012-11-19	1.3	Change some Requirements Definition	Stefania Pezzetti
2012-12-10	1.4	Change some Requirements Definition. Added a use case. Changed Use Case Diagram	Stefania Pezzetti Gregorio Perego
2012-12-23	1.5	Removed unusual alarm activities from Use Case Diagram	Gregorio Perego
2012-12-27	1.6	Added some Use Cases and updated requirements. Revised the entire document.	Stefania Pezzetti Gregorio Perego
2012-12-28	1.7	Updated Future Development	Stefania Pezzetti
2013-01-08	1.8	Revised the entire document.	Marko Kelava
2013-01-20	2.0	Revised the entire document.	Stefania Pezzetti

Contents

1.   Introduction

The problem with current health system is that the patient has to go to the doctor for a live check-up to determine how his treatment is going. All checks for vital functions can be done at home, but patient still needs to be physically there for the doctor to measure the vital parameters.

E- Health Service is an idea aimed at improving the quality of the health service system and making the service more flexible for both patients and doctors. This will be done by creating a web and a mobile application. The web application is intended primarily for home use, while the mobile application is intended for outside use.

The system will allow patients to input vital parameters via mobile application or web from their home and then the doctor can check it in his office. This will give some feedback from the doctor if they are in a critical situation. Doctors will be able to see the patient's vital functions directly in the system without the patient coming for a live check-up.

In case of an emergency the doctor will be notified, without checking the patients profile, that the patient's vital functions are critical and he will be able to alarm the first aid. The first aid will get all the necessary information for their intervention.

The system will also include a list of all doctors, so the patients will be able to see the doctor's professional information.

1.1   Purpose of this document

The purpose of this document is a formal, approved document used to guide the team by the requirements of the system. The primary use of the Requirements Definition is to list in details all requirements of the system that is to be developed. Requirements are defined and categorized by priorities, the essential and the optional requirements. For each requirement there will be a use-case that defines it.

1.2   Intended Audience

The intended audience is:
- The team (Project manager, team leader, developers, etc.)
- Customer
- Supervisor
- Anyone else involved in the project

1.3   Scope

The scope of the requirements document is the definition of all requirements and related use cases.

1.4   References

Main references to this project can be found on the DSD course official webpage:
http://www.fer.unizg.hr/rasip/dsd/projects/e-health_service

2.   Overall Description

E-Health Service is a system for medical care, useful for patients and doctors. It is intended for all the patients and doctors of one city and for the first aid of that city (the designed system will have only one first aid). The system will be composed of a web application and a mobile application.

2.1   Product Perspective

There is no plan for the system to connect to outside sources or to allow outside sources to connect to the system. Within the system there will be:
- Database: stores all data.
- Service layer: encapsulates the business logic, including data storage. The mobile and web application are strictly going to use this layer, and are not going to communicate directly with the database.
- Web application: displays the data and sends user input to the service layer.
- Mobile application: displays the data and sends user input to the service layer.

2.2   Product Functions

The system will include patient's personal information and doctor's professional information, which will be consulted by interested patients. The goal of our application is to help patients make the necessary vital parameters measurements via mobile or web application and then send them to their doctor. For the doctors it will be easier to keep track of their patients and the first aid will be more accessible in emergencies. On this latter feature, because we are not experts in medical science, we wrote just an example of how the system can work using emergency levels, which are known by doctors and first aid with the aim of making more immediate communication between the two subjects and improving the service to patients.

Since vital parameters which can be measured by patients themselves are Glycaemia, Blood Pressure, Respiratory Rate, Oxygen Saturation, Heart Rate and Temperature, emergency levels are based on these parameters and they are basically codes where 1 means light emergency and 2 means strong emergency.

In addition:
- G means that the problem refers to Glycaemia.
- B means that the problem refers to Blood pressure
- R means that the problem refers to Respiratory rate
- S means that the problem refers to Saturation
- H means that the problem refers to Heart rate
- T means that the problem refers to Temperature

There is also a code ALL which can be used when the patient has more than one problem and he needs the entire equipment available to the ambulance, and a rescue team as soon as possible.

LEVEL	PROBLEM	SEND RESCUE TEAM IN	SPECIFIC EQUIPMENT
G1	Glycaemia	Max 1 day	Nurse - glucometer- insulin
			Nurse – glucometer –sugar
G2	Glycaemia	30 – 60 mins	Doctor - nurse - insulin - glucometer
			Doctor – nurse – glucometer - glucagon - glucose solution
B1	Blood pressure	Max 3 days	Nurse - sphygmomanometer – beta 2 agonist
			Nurse – sphygmomanometer – pharmaceutical therapy
B2	Blood pressure	60 – 120 mins	Doctor - Nurse - sphygmomanometer – diuretic
			Doctor – Nurse - sphygmomanometer – drips – blood transfusion
R1	Respiratory rate	Max 3 days	Nurse – stethoscope
R2	Respiratory rate	4 – 5 hours	Doctor – nurse - stethoscope – Oxygen -

			bronchodilator
S1	Saturation	Max 2 days	Nurse - Oximeter - stethoscope
S2	Saturation	Max 3 hours	Doctor - Nurse - Oxygen – Oximeter - stethoscope
H1	Heart rate	Max 3 days	Nurse – pulse – sphygmomanometer pharmaceutical therapy
H2	Heart rate	Max 60 mins	Doctor - Nurse - Pulse – defibrillator - sphygmomanometer
T1	Temperature	Max 6 hours	Nurse – thermometer – acetaminophen
T2	Temperature	Max 60 mins	Doctor- nurse - thermometer – acetaminophen + aspirin – ice - drips
ALL	ALL	ASAP	Entire equipment

Doctors can set vital parameters' range for each patient, so they can decide different ranges based on their observation and taking into account patient's personal situation, but below we report an example of possible vital parameters ranges:

- **Glycaemia**
  70 mg/dl < GOOD < 100 mg/dl
  BAD < 70 mg/dl || BAD > 100 mg/dl
- **Blood pressure**
  It is composed by 2 values and it is expressed in this way: max / min
  GOOD → max <= 140 mmHg && min >= 60 mmHg
  BAD → max > 140 mmHg || min < 60mmHg
- **Respiratory rate**
  10 breaths/min < GOOD < 14 breaths/min
  BAD < 10 breaths/min || BAD > 14 breaths/min
- **Saturation**
  GOOD > = 90 % O2
  BAD < 90 % O2
- **Heart rate**
  60 bpm <= GOOD <= 100 bpm
  BAD < 60 bpm || BAD > 100 bpm
- **Temperature**
  GOOD < 38 Celsius
  BAD > 38 Celsius

Doctors can observe vital parameters and if they need to, they can inform the first aid about an emergency by sending the patient's personal information (name, surname, position, medical history), diagnosis and an emergency level in the form of an alarm which can then be acted upon by first aid.

2.3   User Characteristics

The main user is the patient, which must be older than 18 and have the necessary computing skills to handle the web or mobile application. As for doctors and members of first aid, they must have a medical degree and should also have the necessary computing skills.

2.4   Constraints

- Limitation of the VM provided to us
- Security protocols can be implemented to connect to the server
- Privacy of the single user is defined so that only his doctor can see his personal information
- Limitations of the mobile platform

3. Requirements Description

3.1 Introduction

The purpose of this project is to create a system with a mobile and a web client that helps patients, doctors and first aid staff exchange information, patient history and send alerts for a faster, easier and more efficient medical service.

3.2 General requirements

The system must incorporate sharing of patient's information between patients, doctors and first aid team. System must allow access to basic information about doctors to users who are not registered in the system. System must allow both patients and doctors to send alerts to first aid team; it must be able to automatically send alerts to first aid team depending on defined rules.

3.3 Functional requirements

Administrator should be able to create and delete accounts for doctors, change first aid's login password and change his personal login password.

Outside users should be able to access public information written by doctors, such as work experience and CV, and download it as a pdf file.

Doctors should be able to:
- Edit their public information.
- Create patient accounts and delete those accounts.
- Add, modify and delete patient medical notes.
- Add and delete patient's measured vital parameters.
- Define and edit mandatory parameters for patients.
- Define and edit parameters ranges for patients.
- Send alarms to first aid team with patient's name, location, alarm level and diagnosis.
- Access patient's medical notes, vital parameters and images or files (on web).

Patients should be able to:
- Add a new measurement of vital parameters.
- Update their location.
- Send alarms using a mobile application to first aid team without providing any information. The alarm would contain the current GPS location of the patient.
- Change their personal login password.

First aid should be able to:
- Access the personal information of patient related to alarm received.
- Get a list of alarms in web application.
- Delete alarms when they are acted upon, in web application.

Mobile application should:
- Display patient's medical notes (without images or others attachment files), all measurements of patient's vital parameters to the doctor in charge.
- Allow patient to input a new measurement of his vital parameters, and force him to input mandatory vital parameters (if defined).

Web application should:
- Display patient's medical notes (with images or others attachment files), all measurements of vital parameters to the doctor in charge of the patient.
- Allow patient to input a new measurement of his vital parameters, and force him to input mandatory vital parameters (if defined).
- Allow patients to input their address and update location.

- In alarms for the first aid: display patient's personal data, GPS location (if the alarm has been sent from the patient's mobile phone), alarm level and diagnosis (if alarm was sent by the doctor).

The service should:
- Encapsulate storage (database).
- Communicate with mobile and web application, exchanging data.
- Calculate possible notifications from patient inputs based on defined alarm levels and alert doctor (via e-mail) when it detects that one of the patient's vital parameters is outside of normal range.
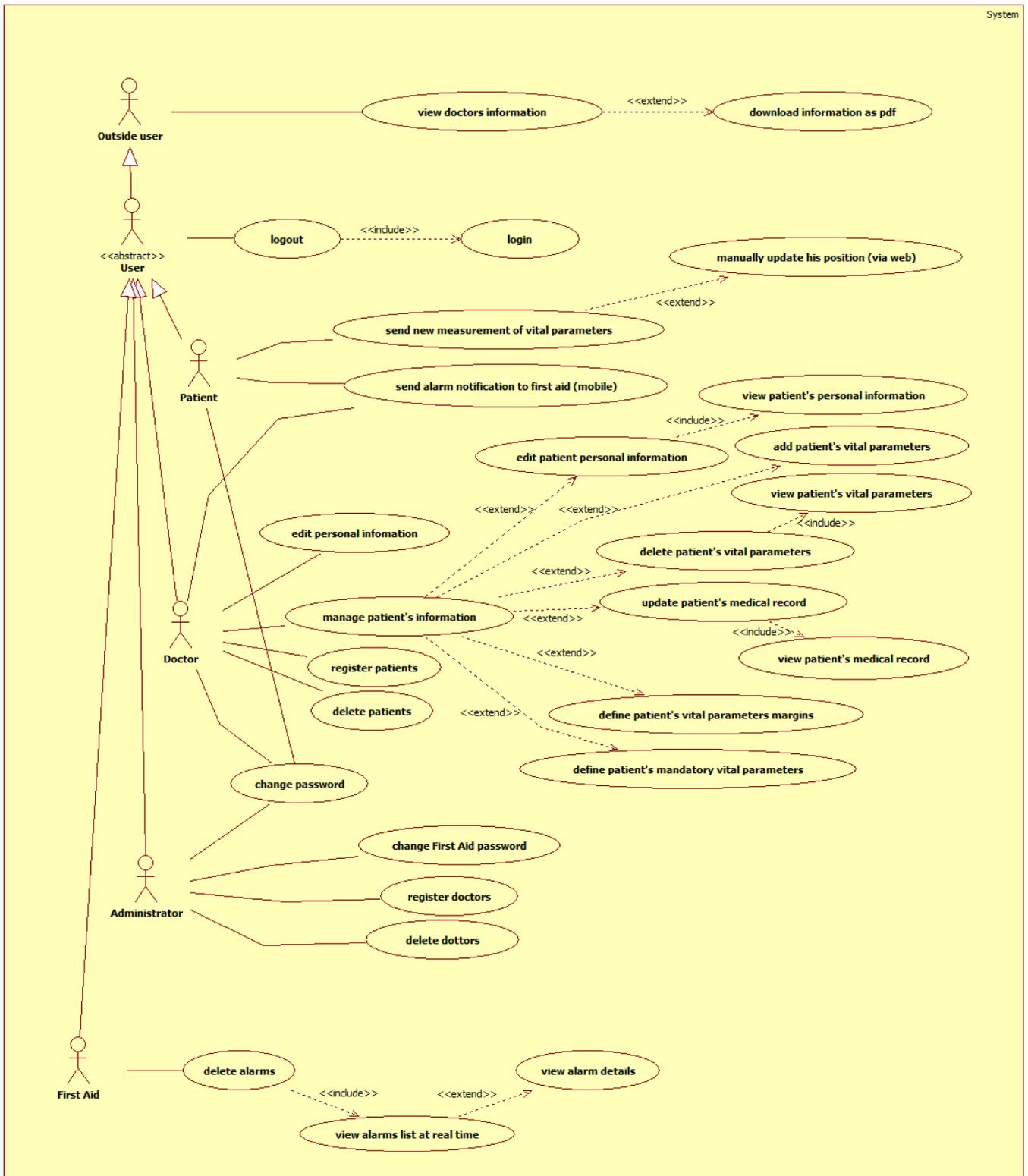
3.4   Non-functional requirements

Non-functional requirements are concerned with the quality of service, not functionalities.
- Mobile application will be developed for Android.
- Mobile application can, but does not have to be developed for Windows Phone.
- The UI for all applications should be intuitive and easy to use.
- Performance: timing of finding and entering the information in the database must be acceptable.
- Reliability and stability: the system must always notify the user about the result of each operation requested by him, especially if managed in the wrong way, maintaining a good level of reliability. Therefore, the system must be able to cope with a possible loss of connection between the client and the central server, ensuring atomicity of all transactions.
- Availability of service: the system must be accessible to each user at any time of the day. So we assumes that servers are equipped with continuity group which, in cases of absence of electric current, prevents the blocking of the service.
- Concurrency: the system will ensure data consistency after concurrent access to the same resource in the database by multiple users authenticated to the system.
- Security: each user will have access only to the functionalities of his role. Therefore, it is necessary to provide a mechanism for authentication with username and password.

4. Use Cases

4.1 Use Case diagram

4.2 Outside user use cases

Use case ID	OUTSIDE USER 1
Name	Check doctor's public information.
Brief description	Displays public information of doctors to anyone who wants to check it.
Participating actors	Outside user.
Precondition	
Main scenario	1. User requests a list of all doctors. 2. User selects a doctor whose details he wants to look at. 3. The doctor's public profile is displayed to the user. The user can see doctor's name, surname, description and e-mail address.
Alternative scenarios	

Use case ID	OUTSIDE USER 2
Name	Downloading doctor's public information as pdf.
Brief description	After selecting a doctor, an outside user downloads the doctor's public profile.
Participating actors	Outside user.
Precondition	An outside user is viewing a doctor's public profile.
Main scenario	1. User requests the doctor's profile in PDF. 2. The system allows the user to download the file.
Alternative scenarios	

4.3   Administrator use cases

Use case ID	ADMIN1
Name	Add a doctor.
Brief description	The administrator adds a new doctor to the system.
Participating actors	Administrator
Precondition	User logged in as administrator
Main scenario	1. User requests adding a new doctor. 2. User inputs name, surname and e-mail, which will serve as username. 3. User submits request. 4. The doctor account is created. 5. An e-mail is sent by the system to the doctor's e-mail address with account information.
Alternative scenarios	*In step 2 if an e-mail address which is already in use is entered* 3. Error message is shown. 4. System returns to step 2 of main scenario.

Use case ID	ADMIN2
Name	Delete a doctor.
Brief description	The administrator deletes a doctor, and all his patients and related data get deleted.
Participating actors	Administrator.
Precondition	User logged in as administrator.
Main scenario	1. User requests a list of all doctors. 2. A list of all doctors is displayed. 3. User chooses the doctor he wishes to delete. 4. The system asks for confirmation. 5. User confirms deletion. 6. The doctor's account is removed from the system. 7. All patient accounts from that doctor are deleted.
Alternative scenarios	*In step 5: User cancels.* 6. The system takes no action.

Use case ID	ADMIN3
Name	Change the login password of the first aid.
Brief description	Administration changes the password of the first aid.
Participating actors	Administrator.
Precondition	User logged in as administrator.
Main scenario	1. User requests changing the password of the first aid. 2. User inputs the new password. 3. The system changes the password of the first aid.
Alternative scenarios	*In step 2: User inputs the same password* 4. The system displays a warning that passwords must differ.

Use case ID	ADMIN4
Name	Change his login password.
Brief description	Administration changes his login password.
Participating actors	Administrator.
Precondition	User logged in as administrator.
Main scenario	1. User requests changing his login password. 2. User inputs the new password. 3. The system changes the password of administrator.
Alternative scenarios	*In step 2: User inputs the same password* 4. The system displays a warning that passwords must differ.

4.4   Doctor use cases

Use case ID	DOCTOR1
Name	Edit personal information.
Brief description	A doctor changes his public profile visible to outside users.
Participating actors	Doctor.
Precondition	User logged in as doctor.
Main scenario	1. User requests editing his public profile. 2. The system shows user's name, surname, description and e-mail address. 3. Edits desired parameters. 4. Requests storing his new profile. 5. The system stores user's new profile.
Alternative scenarios	

Use case ID	DOCTOR2
Name	Add patient.
Brief description	Add a patient assigned to that doctor.
Participating actors	Doctor.
Precondition	User logged in as doctor.
Main scenario	1. User requests adding a new patient. 2. User inputs name, surname, date of birth, ID number, e-mail address, which will also serve as username, phone number and optionally mandatory vital parameters. 3. User submits patient's profile for creation. 4. Patient's account is created. 5. An e-mail is sent to the e-mail address with account information.
Alternative scenarios	*In step 3 if an e-mail address which is entered is already used* 4. An error message is shown. 5. System returns to step 2 of main scenario.

Use case ID	DOCTOR3
Name	Delete patient.
Brief description	A doctor deletes a patient from the system.
Participating actors	Doctor.
Precondition	User logged in as doctor.
Main scenario	1. User requests a list of all his patients. 2. All patients are shown to the user. 3. User finds the patient he wants to delete. 4. User is asked for confirmation. 5. User confirms patient deletion. 6. Patient's account is deleted.
Alternative scenarios	*In step 5: User cancels.* 6. The system takes no action.

Use case ID	DOCTOR4
Name	Check patient's details and medical record.
Brief	A doctor gets insight into patient's details and medical record.

description	
Participating actors	Doctor.
Precondition	User logged in as doctor.
Main scenario	1. User is in his personal area. 2. All the user's patients are shown to him. 3. User selects a patient. 4. The patient's personal data is shown to the user (see Use Case DOCTOR2) 5. A set of medical entries which can also contain images is shown to the user. If the user is currently using a mobile application, no images are shown. Each medical entry contains the date and time of creation.
Alternative scenarios	*In step 2 the user does not have any patients to be displayed* 2. An empty list is shown. Steps 3-4 are not available.

Use case ID	DOCTOR5
Name	Add a note about medical history.
Brief description	Doctor adds a new note to the patient's medical history.
Participating actors	Doctor.
Precondition	User logged in as doctor. User has chosen desired patient.
Main scenario	1. User requests adding a new note for the patient. 2. User types in an entry. 3. User attaches an image (optional). 4. User submits data. 5. The note is stored in the patient's medical history.
Alternative scenarios	

Use case ID	DOCTOR6
Name	Edit a note about medical history.
Brief description	Doctor edits patient's note.
Participating actors	Doctor.
Precondition	User logged in as doctor. User has chosen desired patient.
Main scenario	1. List of all notes is displayed. 2. User selects patient's note. 3. User edits note details. 4. User submits changes. 5. Patient's note is changed.
Alternative scenarios	

Use case ID	DOCTOR7
Name	Delete a note about medical history.
Brief description	Doctor deletes patients note.
Participating actors	Doctor.
Precondition	User logged in as doctor. User has chosen desired patient.
Main scenario	1. List of all notes is displayed. 2. User selects patients note. 3. User requests patient's note deletion.

	4. The note is deleted.
Alternative scenarios	

Use case ID	DOCTOR8
Name	Delete a measurement of a patient's vital parameters.
Brief description	Doctor deletes an entry of measurements of patient's vital parameters.
Participating actors	Doctor.
Precondition	User logged in as doctor. User has chosen desired patient.
Main scenario	1. User requests the list of all measurements of patient's vital parameters. 2. List of all measurements is displayed. 3. User selects patient's measurement. 4. User requests deletion. 5. The measurement entry is deleted.
Alternative scenarios	

Use case ID	DOCTOR9
Name	Define and edit patient's mandatory vital parameters.
Brief description	Doctor changes which vital parameters are mandatory.
Participating actors	Doctor.
Precondition	User logged in as doctor. User has chosen desired patient.
Main scenario	1. User requests changing his patient's mandatory vital parameters. 2. List of all vital parameters is displayed. 3. User changes mandatory vital parameters. 4. User confirms input. 5. The system stores the change.
Alternative scenarios	*In step 4: User cancels.* 5. The system takes no action.

Use case ID	DOCTOR10
Name	Add new measurement of vital parameters.
Brief description	Doctor adds new vital parameters measurement.
Participating actors	Doctor.
Precondition	User logged in as doctor. User has chosen desired patient.
Main scenario	1. User requests adding a new measurement of vital parameters. 2. User inputs values of desired parameters. 3. User confirms input. 4. The system stores measurement in patient's medical history (notes).
Alternative scenarios	*In step 2: User does not input mandatory vital parameters.* 4. An error message is shown to the user. 5. User is returned to step 2 of the main scenario. *In step 3: User cancels.* 4. The system takes no action.

Use case ID	DOCTOR11
Name	Set patient's vital parameters margins.
Brief	Set patient's span of the patient's vital parameter.

description	
Participating actors	Doctor.
Precondition	User logged in as doctor. User has chosen desired patient.
Main scenario	1. User requests setting patient's vital parameters margins. 2. User inputs desired interval. 3. User confirms input. 4. Patient's vital parameters margins are saved.
Alternative scenarios	*In step 3: User cancels.* 4. The system takes no action.

Use case ID	DOCTOR12
Name	Send alarm with alarm level to the first aid team.
Brief description	Doctor sends alarm to the first aid team with an optional diagnosis and an alarm level.
Participating actors	Doctor.
Precondition	User logged in as doctor. User has chosen desired patient.
Main scenario	1. User requests the vital parameters measurement list. 2. User requests sending an alarm to the first aid. 3. User selects the alarm type. 4. User optionally inputs the diagnosis. 5. User confirms send alarm. 6. The alarm is saved in the system.
Alternative scenarios	*In step 5: User cancels.* 6. The system takes no action.

Use case ID	DOCTOR13
Name	Change login password.
Brief description	Doctor changes his own password.
Participating actors	Doctor.
Precondition	User logged in as doctor.
Main scenario	1. User requests changing his password. 2. User enters the current password. 3. User enters the new password. 4. User submits request. 5. User's password is changed.
Alternative scenarios	*In step 2 user enters a password which does not match the existing one* 5. An error message is shown to the user.

Use case ID	DOCTOR14
Name	Edit patient personal information.
Brief description	A doctor changes his patient's personal information.
Participating actors	Doctor.
Precondition	User logged in as doctor.
Main scenario	1. User views a list of all his patients 2. User selects a patient 3. User sees patient's personal information and chooses to edit this information 4. Save changes. 5. Returns on the patient's details page.

Alternative scenarios	*Step 4: User cancels.* 5. The system takes no actions.

4.5 Patient use cases

Use case ID	PATIENT1
Name	Input a new measurement of vital parameters.
Brief description	A patient wants to input a new measurement of his vital parameters.
Participating actors	Patient.
Precondition	User logged in as patient.
Main scenario	1. User requests adding a new measurement of vital parameters. 2. User inputs mandatory vital parameter values and optionally inputs non-mandatory values. 3. User requests submitting the input values. 4. The system stores the input values.
Alternative scenarios	*In step 2 user does not enter a mandatory vital parameter field* 4. The system displays an error message until all vital parameters have been entered or until user cancels the request. *In step 2 user enters a measurement not in valid range* 4. The system displays an error message

Use case ID	PATIENT2
Name	Update location using a web browser.
Brief description	The patient updates his location by manually inputting his current address.
Participating actors	Patient.
Precondition	User logged in as patient.
Main scenario	1. User requests location update from the system. 2. User inputs his location by writing his address. 3. User requests submitting the input values. 4. The system stores the address.
Alternative scenarios	

Use case ID	PATIENT3
Name	Send alarm via mobile device.
Brief description	Patient sends an alarm to the first aid via mobile device.
Participating actors	Patient.
Precondition	User logged in as patient
Main scenario	1. User requests sending an alarm to the first aid. 2. The system sends the alarm to the first aid containing the current GPS coordinates of the user, extracted from the mobile phone.
Alternative scenarios	

Use case ID	PATIENT4
Name	Change login password.
Brief description	Patient changes his login password.
Participating actors	Patient.
Precondition	User logged in as patient

Main scenario	1. User requests changing is login password. 2. User inserts his old password. 3. User inserts his new password. 4. User confirms. 5. The system saved the new password.
Alternative scenarios	*In step 2: User inserts an incorrect password.* 5. The system displays an error message. *In step 4: User cancels:* 5. The system takes no actions.

4.6 First aid use cases

Use case ID	FIRSTAID 1
Name	Display list of all alarms sent by patients or doctors.
Brief description	The first aid team wishes to see all received alarms and receives that information from the system.
Participating actors	First aid.
Precondition	User logged in as First aid.
Main scenario	1. User requests the list of all received alarms sent either by the patient or the doctor. 2. The system displays all received alarms.
Alternative scenarios	

Use case ID	FIRST AID 2
Name	View alarm and patient details of particular alarm sent by a patient or a doctor.
Brief description	The first aid team wants to get more information regarding a received alarm.
Participating actors	First aid.
Precondition	User logged in as First aid.
Main scenario	1. User requests the list of all received alarms sent either by the patient or the doctor. 2. The system displays all received alarms. 3. User requests more information about one particular alarm to the system. 4. The system shows the following information of the patient: 4.1. Received time 4.2. Last known GPS location of the patient 4.3. Basic personal data of the patient (see Use Case DOCTOR2) 4.4. Diagnosis, provided that the alarm has been sent by the doctor. 4.5. Alarm level, provided that the alarm has been sent by the doctor.
Alternative scenarios	

Use case ID	FIRST AID 3
Name	Delete an alarm notification sent by a patient or a doctor.
Brief description	The first aid team wants to delete an alarm notification received by a patient or a doctor.
Participating actors	First aid.
Precondition	User logged in as First aid.
Main scenario	1. User requests the list of all received alarms sent either by the patient or the doctor. 2. User finds the alarm he wants to delete. 3. User requests the deletion of the alarm. 4. The system deletes the alarm.
Alternative scenarios	

5. Requirements Definition

This section defines all requirements for the developed system.

5.1 Requirement Group Definitions

Each requirement belongs to one requirement group: in the table below, are listed all requirement groups, their identification code and longer requirement group name.

Identification	Requirement Group	Rem.
ADM	System Administration	
OU	Outside user	
DR	Doctor	
FA	First Aid	
PA	Patient	
SER	Server	

5.2 Requirement Sources

Each requirement is defined by its source: in the table below are defined all sources of requirements in the system.

Source	Description	Rem.
CTM	Customer	
SYS	Required as a consequence of system design	

5.3   Requirement definitions

Identity	Status	Priority	Description	Source
			General Actions	
GEN-1	I	1	Four user roles: administrator, doctor, first aid, patient.	CTM
GEN-2	I	1	Registration of first aid users not necessary: only a single first aid user is enough.	CTM
GEN-3	I	1	The system implements access control by mandatory logins with a user name and password. The interface and the functionality depend on the role of the user that has logged into the system.	CTM
GEN-4	I	1	Outside users have limited access to the system.	CTM
GEN-5	I	1	Access to the system using a web browser or a client application for smartphones.	CTM
GEN-6	I	1	Mobile application will be developed for Android.	CTM
GEN-7	I	3	Mobile application might be developed for Windows Phone.	CTM
GEN-8	I	1	Minimum supported browsers: Google Chrome.	CTM
GEN-9	I	2	A new measurement of vital parameters input cannot be accepted by the system unless all mandatory vital parameters values have been entered.	CTM
GEN-10	I	1	The administrator can only use the web application.	CTM
GEN-11	I	1	Doctors can use the web application or the mobile application.	CTM
GEN-12	I	1	Patients can use the web application or the mobile application.	CTM
GEN-13	I	1	First aid can only use the web application.	CTM
GEN-14	D	3	System should use http basic for authorization	SYS
GEN-15	D	3	System should use https for encryption	SYS
GEN-16	A	3	System should take no more than 2 minutes using GPS to get location	SYS
			Administrator	
ADM-1	I	2	Adding and deleting doctors.	CTM
ADM-2	I	2	Deleting a doctor deletes all his patients.	SYS
ADM-3	I	2	Change the login password for the first aid.	CTM
ADM-4	A	3	Change his personal login password.	CTM
			Outside users	CTM
OU-1	I	2	Check doctor's public information.	CTM
OU-2	I	3	Downloading doctor's public information as pdf.	CTM
			Doctors	
DR-1	I	1	Edit his public information displayed to outside users.	CTM
DR-2	I	1	Register a new patient to the system.	CTM
DR-3	I	1	Delete a patient from the system. Deleting a patient from the system removes all the data concerning the patient.	SYS
DR-4	I	1	Add, edit and delete notes from patient's medical history.	CTM
DR-5	I	2	Adding a note to a patient's medical history needs to support adding images.	CTM
DR-6	I	2	Define patient's mandatory vital parameters during or after registration.	CTM
DR-7	I	2	For each vital parameter of a patient, set a normal interval, which doesn't trigger an alarm. This option should be available during or after registration.	CTM
DR-8	I	2	Input and delete measurements of vital parameters of the patient.	CTM
DR-9	I	1	Send alarm with patient's data to the first aid.	CTM
DR-10	I	2	Change his login password.	CTM
DR-11	A	2	Edit patient's personal data	CTM

				Patients	
PA-1	I	1		Each patient has exactly one assigned doctor.	CTM
PA-2	D	-		View and edit his personal data, including location.	CTM
PA-3	I	2		Input a new measurement of vital parameters.	CTM
PA-4	I	1		Update location from web by inputting the address.	CTM
PA-5	I	1		Update location using GPS (mobile application).	CTM
PA-6	I	1		Send an alarm from a mobile device without specifying any information.	CTM
PA-7	A	2		After registration receives an email with his login password. Then he can change his login password from his personal home page both with web application and mobile applications.	CTM
				System	
SER-1	I	2		Alert doctor about patient's abnormal vital parameters via e-mail.	CTM
SER-2	D	3		Notify unusual alarm activities to the first aid. An unusual alarm activity is triggered when the system notices that 15 alarms have been raised within 15 minutes inside a 100 meter radius circle.	CTM
				First Aid	
FA-1	I	1		Display list of alarms sent by patients or doctors. The alarm contains patient's data.	CTM
FA-2	I	2		Get alerted when a patient or a doctor sends an alarm.	CTM
FA-3	I	1		Delete received alarm from patients or doctors.	CTM
FA-4	D	3		Get alerted when the system recognizes an unusual alarm activity.	CTM
FA-5	D	3		Display unusual alarm activities. A raised unusual alarm activity has to contain the GPS location of the unusual activity and the number of raised alarms.	CTM
FA-6	D	3		Delete received alarm for unusual activities.	CTM
FA-7	A	3		Display a map that explains where the patient is when he needs help.	SYS

Requirement status:

     I = initial(this requirement has been identified at the beginning of the project),

     D = dropped (this requirement has been deleted from the requirement definitions),

     H = on hold (decision to be implemented or dropped will be made later),

     A = additional (this requirement was introduced during the project course).

5.3.1 Change Log

Table contains the list of changes to requirements from the main table, in chronological order.

Identity	Action	Date	Comments
GEN-14	A	19/11/2012	Added after the Design Description
GEN-15	A	19/11/2012	Added after the Design Description
GEN-16	A	19/11/2012	Added after the Design Description
PA-2	D	19/11/2012	Dropped because the doctor will be in charge to change patient's personal data
DR-11	A	19/11/2012	Added because the doctor has to edit patient's data
PA-7	A	10/12/2012	Added because only doctor can edit patient personal information, but patient has to change his login password.
GEN-15	D	27/12/2012	Dropped because the customer does not require it.
SER-2	D	27/12/2012	Dropped because it seems inconsistent with other requirements.
FA-4	D	27/12/2012	Dropped because the requirement SER-2 was deleted.
FA-5	D	27/12/2012	Dropped because the requirement SER-2 was deleted.
FA-6	D	27/12/2012	Dropped because the requirement SER-2 was deleted.
FA-7	A	27/12/2012	Added because it seems useful.
GEN-14	D	27/12/2012	Dropped because the customer does not require it.

Requirement status:

D = dropped (this requirement has been deleted from the requirement definitions),

H = on hold (decision to be implemented or dropped will be made later),

A = added (this requirement was introduced during the project course).

R = *resurrected* (dropped or on hold requirement was reactivated)

6. Future Development

The E-Health Service future developments can be divided into two categories: the first one regarding the technologies and the second one regarding data processing.

About *technologies*, the most important future development is the integration of sensors to the system. The sensors would be programmed to input data directly to the system. This way, older people who are not very good with personal computers or smartphones would minimize the work with such devices. Another innovation about technologies can be development for other mobile platforms, such as BlackBerry and iOS.
Always in this field the system can be improved by adding a new functionality which detects unusual alarms. As unusual alarms we define which alarms that are detected by system, monitoring alarms from patients. If these alarms are sent from the same building and at the same time, the system will create a "system alarm".
A possible algorithm for this function is reported in chapter 6.1.1.

Referring to *data processing*, the relevant development regards the fact that the system can use data sent by patients and stored as anonymous data for making reports. This can be done using a data warehouse that pulls data from system database and then elaborates data to give the possibility to understand how vital parameters vary, varying areas where they are collected. Then associations, which aim to prevent diseases and pathologies, can intervene with specific prevention campaigns in areas that really need this kind of interventions.

Finally, the system can be expanded to include multiple hospitals and first aid teams. For now, as said before, we are going to develop this system only for one hospital, so it could be upgraded to a regional system or even a national system.

6.1.1   Possible algorithm for System Alarms

```
class Alarm {
      gpsCoordinates;
      timeOfCreation;
}


class UnusualAlarmGroup {
      gpsCentre;
      dateTimeOfUpdate;
      numberOfAlarms;
}


class UnusualEventRecognizer {
      List<Alarm> ungroupedAlarms;
      List<UnusualAlarmGroup> unusualAlarmGroups;
      // needs to be called every time the system receives a new alarm
      void registerNewAlarm(alarm) {
            // groups that have not been active for more than 15 minutes
            deleteOldUnusualGroups( unusualAlarmGroups )
            // alarms older than 15 minutes
            deleteOldUngroupedAlarms( ungroupedAlarms )
            existingGroup=tryFindExistingGroupForAlarm(alarm,unusualAlarmGroups);
            if (existingGroup != null) {
                  existingGroup.numberOfAlarms++;
                  existingGroup.dateTimeOfUpdate = alarm.timeOfCreation;
                  return;
            }
            ungroupedAlarms.add( alarm );
            // find an alarm which is surrounded by 14 other alarms
            // within a 100 meter radius circle
            centralAlarm = tryFindUnusualAlarmGroupCentre(ungroupedAlarms);
            if (centralAlarm == null) return;
```

```
            unusualAlarms = getAlarmsWithinRange(
                            centralAlarm.gpsCoordinates,
                            ungroupedAlarms
            );
            newUnusualGroup = new UnusualAlarmGroup(
                            centralAlarm.gpsCoordinates,
                            getMaximumCreationTime( unusualAlarms ),
                            unusualAlarms.size()
            );
            unusualAlarmGroups.add( newUnusualGroup );
            ungroupedAlarms.removeAll( unusualAlarms );
        }
}
```