Chapter 3

System Requirements and Architecture

3.1 System Requirements

This section presents the system requirements specification, as a result of the first phase of the prototype development. The following subsections begin with a description on the requirements elicitation process, followed by a context description, actors classification, use-case diagrams, non-functional requirements and the system's assumptions and dependencies.

3.1.1 Requirements Elicitation

This specification process can be divided in three main parts. A first one, where the system's main goals were evaluated, helping to define its scope, that is, the system boundaries. The next part was an investigation on the state-of-the-art, looking for projects, studies and technologies similar to the one being developed in this dissertation. This research improved the system requirements firstly, by discovering the latest and most innovative works around this area; and secondly, by understanding what went right and wrong in those works, hopefully this system will gather the best of them.

At last, several informal interviews were made with medical specialists, including doctors and nurses. This gave insights on what the system should do (or not do) and also some directions regarding the health monitoring. During these interviews and based on the opinion of those specialists, important decisions were made on the features to be developed. It should be noticed that although the interviewed clinicians had a different technical background, the sample was not as diverse as it could be, given the available time for these phase was short. An overview of the points-of-view appointed by the specialists are presented next:

- Given the available devices [1], the most easily detected disease would be hypertension. Regarding hypertension, specialists gave us a few insights, namely that the main causes of its appearance are lack of physical exercise, excess of salt and stress. Also, for a correct diagnosis, one blood pressure reading alone is not conclusive, it must always be compared with

[1] At this phase of the thesis, the available devices on Fraunhofer AICOS were a weight balance, a blood pressure monitor, an heart-rate monitor and a foot-pod.

the patient's measurements history and medical background. Besides, for the blood pressure reading to be accredited, it should follow a set of guidelines [NIC11], otherwise it could be meaningless. For instance, if a patient measures himself while in stress or after a workout, the values will always be higher than a regular measurement.

- Each sensor has more or less utility, depending on its target patients or diseases. Both medication and diagnosis processes vary significantly depending on the device and desired monitoring. Also on the sensor devices, a glucose meter would be a very interesting complement the system, given that it monitors a disease – diabetes – that requires constant monitoring of its main values.

- A better knowledge of the disease by the patient himself is always beneficial. As stated by Seto et al. in [SLC+12], appropriate self-care has been found to improve health outcomes. Integrating some kind of hint advisor before and/or after each measurements, would be interesting.

- Regarding the user interface, it is very important to keep things simple and not complicate processes and operations for both patients and physicians.

- From a patient point-of-view, giving constant feedback from the patient's clinician is an important feature, although it may be hard for him to handle extra workload. All doctors indicated nurses, as the ones with the most appropriate profile for managing the patients. They would filter most measurements and the doctor in charge would be contacted in emergency situations, only. In order to optimize the feedback sent by each clinician, the patient readings could have a configurable schedule, so each clinician would be able to manage his own schedule with the moment when patient readings are most abundant. The system should also be able to filter between normal values and the ones exceeding certain threshold values, for easier reading.

- Similar systems being currently used are focused on real-time monitoring, with the patient having to walk with some devices at all time. When the patient goes back to the clinic, all recorded data is exported and only later, analysed. The flexibility and mobility provided by the system being developed here would be a big advantage. One of the clinicians is confident that, despite all medical applications out there, there is still room for a mobile health monitoring system like this, not only because of the clear advantages it may bring, but mostly, because of the focus being given to both patients and physicians.

- One doctor referred the possibility of manually entering the measurements, if the patient has no ANT+ device. Although it does not assure he is not sending fake values – which frequently occurs –, it does not limit the system to these kind of devices and allows the system to handle any sensor that can be used by the patient, at home.

- All interviewed clinicians referred the importance of separating the information presented to patients and physicians. For instance, if an anomaly is detected perhaps it is not wise to show it to the patient right away, or even show it at all. This type of information may worry the patient in a way that is not beneficial for either patient and physician.

Given all these insights and after weighting the pros and cons, the following decisions were made. The devices to be integrated in the system would be the weight scale and the blood pressure monitor, as these are the ones with most medical relevance, easing the clinician task of monitoring hypertension. Because a blood pressure reading must follow certain guidelines to be validated, the mobile application would include a simple wizard for the patient to know how and when to take that reading.

In order to access the system, it was decided to build a web application so that physicians could access it, at any time. Patients would also be able to do so, but their main platform would be a mobile application, installed in his smart-phone, able to communicate with the sensor devices, send and receive information from the web application.

To remember the patient of each assigned measurement and also to give some flexibility to the clinician, the web application would allow the scheduling of alarms for each sensor device. Regarding the separation of information to be shown to patients and physicians, it was decided to not present any kind of warnings to the patient. The decision of when and how to warn the patient, is transferred to the physician, which in deed, is a more natural process. If the anomaly is detected, the physician will be immediately warned and then, if he wants to, he can send a message to his patient saying whatever he feels more appropriate.

Since more than one clinician can be monitoring a patient, the system should be able to associate several clinicians to the same patient. This way, a nurse could be monitoring that patient, for instance, every two days, while his doctor would only access the web application once per month.

At last, it was decided not to implement the automatic anomaly detection system as it was initially thought, at least for now. From what we were told, it is a difficult task even for clinicians and given its complexity, it would take time that is not available during this thesis. Instead, the web application would allow the clinician to set threshold values for each variable of each device, for each of his patients. Once a reading exceeds that threshold value, then the system would present that reading as an irregular measurement, alerting the clinician for that situation.

3.1.2 Context Description

This subsection presents a description on how the system is expected to be used by its different stakeholders. To begin with, a chief physician — someone in charge of the system administration inside the hospital or clinic — should register physicians (doctors and nurses) in the system. This will be an occasional operation taking no longer than a couple minutes. This chief medic should also give instructions to physicians on how to use the system, since they will be the main users of the web application and the ones who will present both web and mobile application to the patients.

Once the system is well understood and a doctor has an appointment with a patient — who could benefit from the monitoring system — he presents it to the patient. If the patient is willing to use the system, he should first register himself on the web application (this can also be done by his physician). It is then necessary to configure the mobile application with the user's credentials, so it is able to communicate with the web server. This operation can be done during the appointment or at a later moment by the patient, if he feels comfortable to do so. The last required step, is for all interested clinicians in watching over this patient to say it on the web application. They can do so with a click of a button on each patient's public page.

By the time the patient leaves the clinic, he can start interacting with the system by taking measurements from his sensors, sending them and receiving feedback from his physician, that will then access the web application whenever he receives an alarm or chooses to analyse the historical data from his patient.

This procedure assumes the patient already has all required devices: one smart-phone and one (or more) sensor devices. Acquiring the sensors will depend on the business model adopted by the medical entity. For instance, the patient could buy the devices to that medical entity or some kind of renting system could be used.

3.1.3 Actors

The target user for the mobile application is a senior person, whose knowledge on technology and smart-phones might be fairly different between each user. However, the interface design was designed to mitigate the low level of the user's expertise, in order to provide a seamlessness interaction with the system. The user of the web application will commonly be a physician, expected to have, at least some experience with computers and browsing the Internet. It is also expected the physicians are capable of making informed decisions on each patient's condition.

The levels of expertise required to seamlessly use the system are little: for the patients, although familiarity with the Android platform is an advantage, the expected knowledge required to use the mobile application is practically none, as the emphasis given on usability and interface design was great; for the physicians, as the web application experience is similar to using any regular website, anyone familiar with web browsing is skilled enough.

Given the fact this system was a developed as a proof of concept, little emphasis was given on the management of the web application. Features such as creating, editing and removing physicians, patients and health records were kept simple, although in future development these would require a clear specification, which would be different whether the system is used within a hospital or a private clinic.

The main actors are described in the following list:

- **Patient**: Represents the patient, having access to all of the features available on the mobile application. He can also access the web application, having mainly read permissions. This allows him to view his historical data, his physician's general information, his assigned sensors and respective measurements frequency. He can also view all received and sent feedback messages, as well as send new messages to the physician.

- **Physician**: User authenticated in the web server, having access to most of the web application features. These features include registering new patients, managing them and setting up general system settings. This actor is not supposed to interact with the mobile application.
- **Chief medic**: Entity with most authority on the system, allowed to do everything a physician is able to do and, also capable of registering and removing new physicians in the system.

3.1.4 Use Cases

Figure 3.1 presents the use case model of the whole system, having two main packages: the mobile application package and the web application package. They both represent the usage of the mobile and web application, respectively.

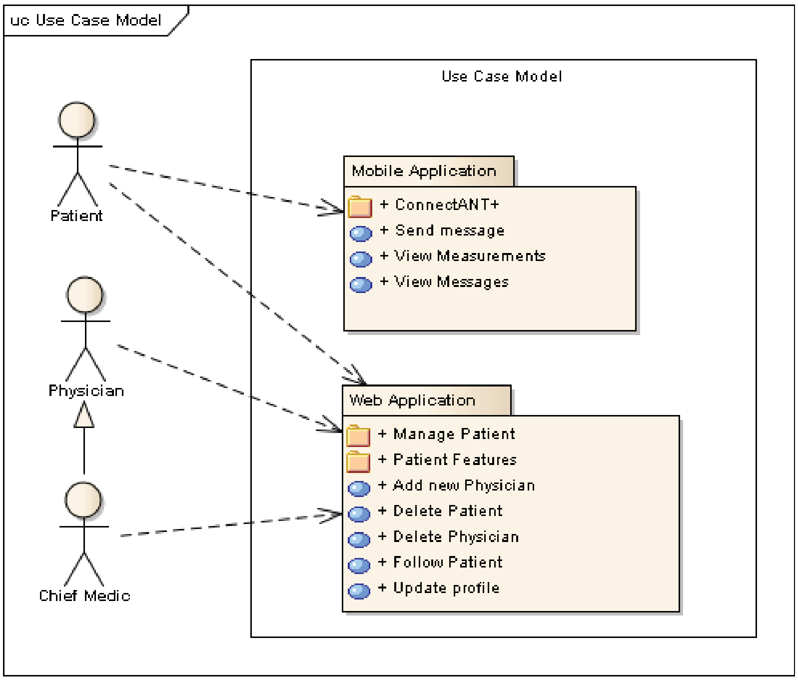

Figure 3.1: Use Case Model

As it can be seen, a patient will use both mobile and web application, while the physician and chief medic solely interact with web application. The following subsections provide a description for each of the use cases in each package. Although there is a connection from the patient to the web application, not all use cases are available to him. These details are further detailed on the relevant use cases.

3.1.4.1 Mobile Application

The mobile application use case diagram 3.2 shows the several activities the patient can perform on his smart-phone. The *Connect Ant+* package (Figure 3.3) encompasses all operations involving the connection between sensors and smart-phone. The following list describes each use-case:

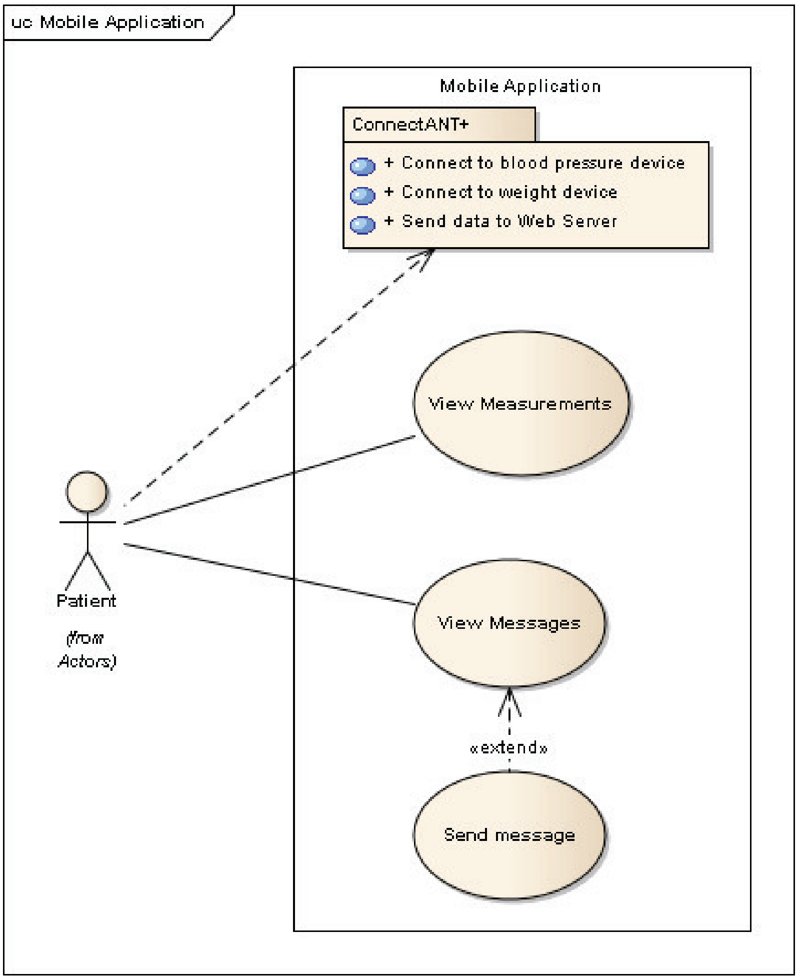

Figure 3.2: Mobile Application

- **View Measurements**: Allows the patient to view his past measurements, grouped by sensors. It is supposed for the user to have already taken some measurements and sent them to the web server.
  A measurements list will appear upon opening this view. It is possible to group this list by different sensors, selecting them from a check list.
  Priority: High
- **View Messages**: Shows all sent and received messages, assuming there is at least, one message sent from the physician. This message list will not be updated until the patient opens the mobile application being connected to the Internet.
  The list of messages appears when this view is opened. Each list item can be clicked in order to view the full content of the message.
  Priority: High
- **Send Message**: Allows the patient to send a text message to the web server, assuming there is an Internet connection available.

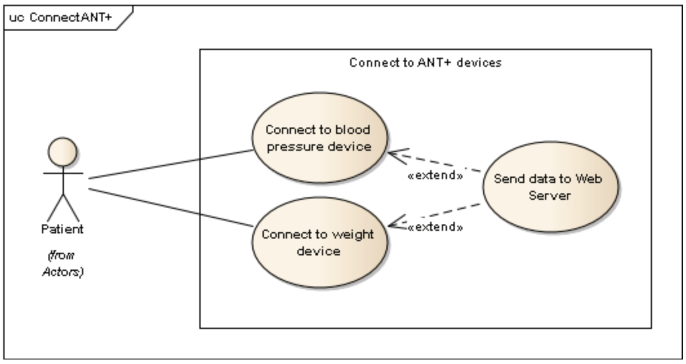

Figure 3.3: Connect to ANT+

This use case is available through the previous one – View Messages – having an input form to write and send the message.
Priority: High

- **Connect to blood pressure device**: Establishes a connection with the blood pressure device and receives the latest measured data, assuming the user has an ANT+ blood pressure device near him. Once he chooses this device, the patient's procedure is to follow the carefully designed instructions and wait until the connection is established and the values are read.
  Priority: High

- **Connect to weight device**: Connects with the weight device and receives the latest measured data, assuming the user has an ANT+ weight scale near him. Just like the previous use case, once the patient selects this device, he simply follows the given instructions on his smart-phone and in time, the values should appear in his smart-phone.
  Priority: High

- **Send data to Web Server**: Sends the measured values to the web application through the network. It is expected the user has already performed one of the previous use cases – connect to blood pressure device or connect to weight device – and has an Internet connection available, otherwise it will not send those values. If the measurement is successfully sent, it is stored in a local database, so the patient can review his measurements whenever he wants to.
  Priority: High

It is important to notice, that all ANT+ devices connected to the mobile application must operate according to their corresponding protocol or profile, as defined by the ANT+ Alliance. These profiles assure the interoperability between ANT+ devices. However, different protocols sometimes require different use cases for each device. These questions are better described later, on chapter 4, but the fact that one of the main goals in this thesis was to keep the user interface easy and concise, efforts were made in order to keep the same user interaction between each device.

3.1.4.2 Web Application

Figure 3.4 presents the use case diagram related with the web application. As it was introduced in subsection 3.1.3, several actors can access it: patients, physicians and a chief-medic. The patient only has access to reading operations; the chief medic is allowed to do everything the physician does, plus adding new physicians into the system. A detailed description on each of the use cases is presented below.

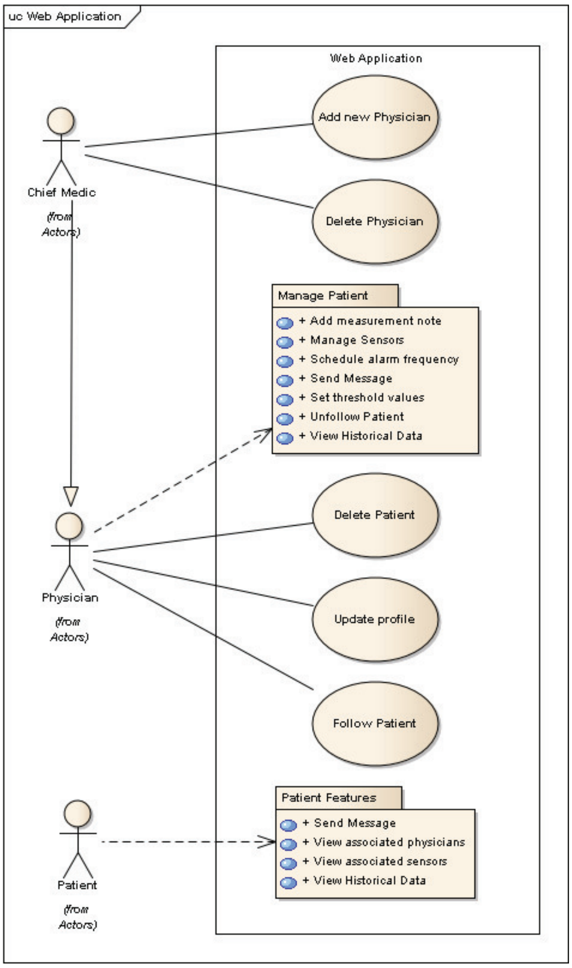

Figure 3.4: Web Application

- **Add New Physician**: Allows the chief medic — and no one else — to register new physicians, a doctor or a nurse, into the system. This will be a seldom operation, as the number of physicians in the clinic does not change very often.
  The chief medic introduces the physician basic data, such as name, speciality, email contact, among others and after submitting the form, an email is sent to the newly physician with his system credentials.
  Priority: Low

- **Delete Physician**: Opposite use case to the previous one, enabling the chief medic to remove physicians from the system.
  Priority: Low

- **Delete Patient**: Similarly to the previous use case, allows the removal a patient from the system. It can be done by either the chief medic or a physician.
  Priority: Low

- **Update Profile**: Use case where the user is presented with a simple form allowing him to change his personal data, such as name, contact, password, among others.
  Priority: Low

- **Follow Patient**: By following a patient, the physician is allowed to do all operations regarding the management of a patient, as described in Figure 3.5, allowing the physician to keep an eye on as many patients as he wants to.
  Priority: High

Regarding the management of each physician's patient, Figure 3.5 presents the use case diagram that encompasses all operations available to interact with them. All the use cases on this package assume the user already selected a patient to interact with.

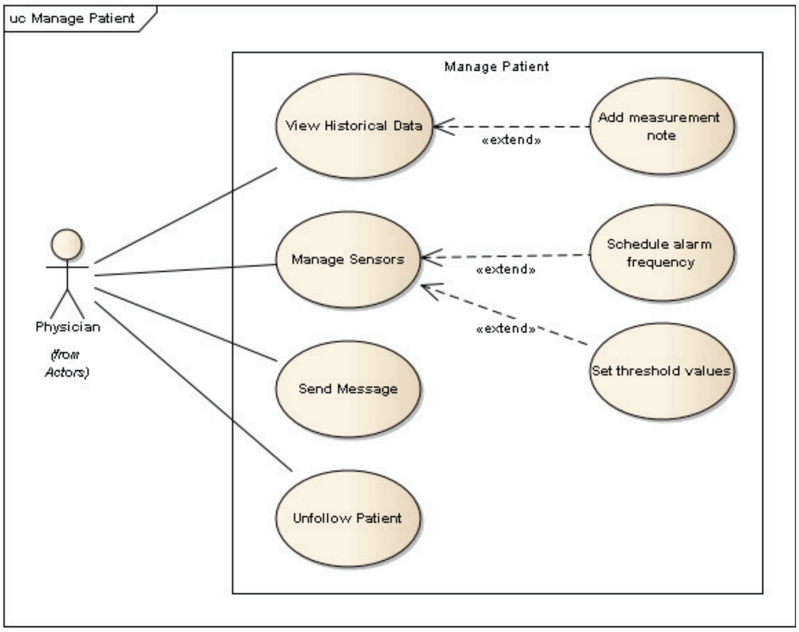

Figure 3.5: Manage Patient

- **View Historical Data**: Shows both measurements taken by this patient and feedback messages sent to him, or received by him, ordered by date. The measurements list can grouped by the different sensors associated with the patient. For each measurement it is also possible

to do the following operation.
Priority: High

- **Add Measurement's Note**: Allows the physician to add a specific note for a measurement on this patient, visible for all doctors and nurses associated with this patient. Each measurement has an appended button that pops up a simple form to write the note. Once created, the note can be edited or removed at any time.
  Priority: Low

- **Send Message**: Allows the physician to send feedback messages to his patient. The patient will then be able to see those messages in his smart-phone or in the web application.
  Priority: High

- **Unfollow Patient**: Allows the user to stop monitoring a patient. By choosing this option, the physician stops having access to all operations inside this use case package (Figure 3.5)
  Priority: High

- **Manage Sensors**: Allows the physician to associate or dissociate sensors from this patient. The list of sensors of a patient, defined in this use case, has a direct impact on what measurements are taken by that patient, since the physician will need to specify both an alarm frequency for the user to take this measurement and the threshold values associated with this device. These two use cases are described below. **Priority**: High

  - **Schedule Alarm Frequency**: Allows the physician to specify how frequently should this patient take a measurement with this sensor. This frequency is then reminded to the patient on his smart-phone with custom-built alarms, remembering him to measure this value.
    As it was previously described, the physician has to specify how many times per day, week or month, he wants his patient to perform the measurements and when the patient's smart-phone connects with the web server, the system synchronizes itself.
    Priority: High
  - **Set Threshold Values**: Lets the physician specific threshold values, regarding this sensor. The physician can specify a lower bound and/or upper bound for each value received with this sensor.
    Whenever a patient's measurement exceeds a threshold value, that measurement appears with an alarm icon, so the physician will know something wrong happened. This alarm icon will only be visible for the associated physicians.
    Priority: High

The *Patient Views* package, shown on Figure 3.6, presents all operations that a patient is allowed to do on the web application. These operations are described below.

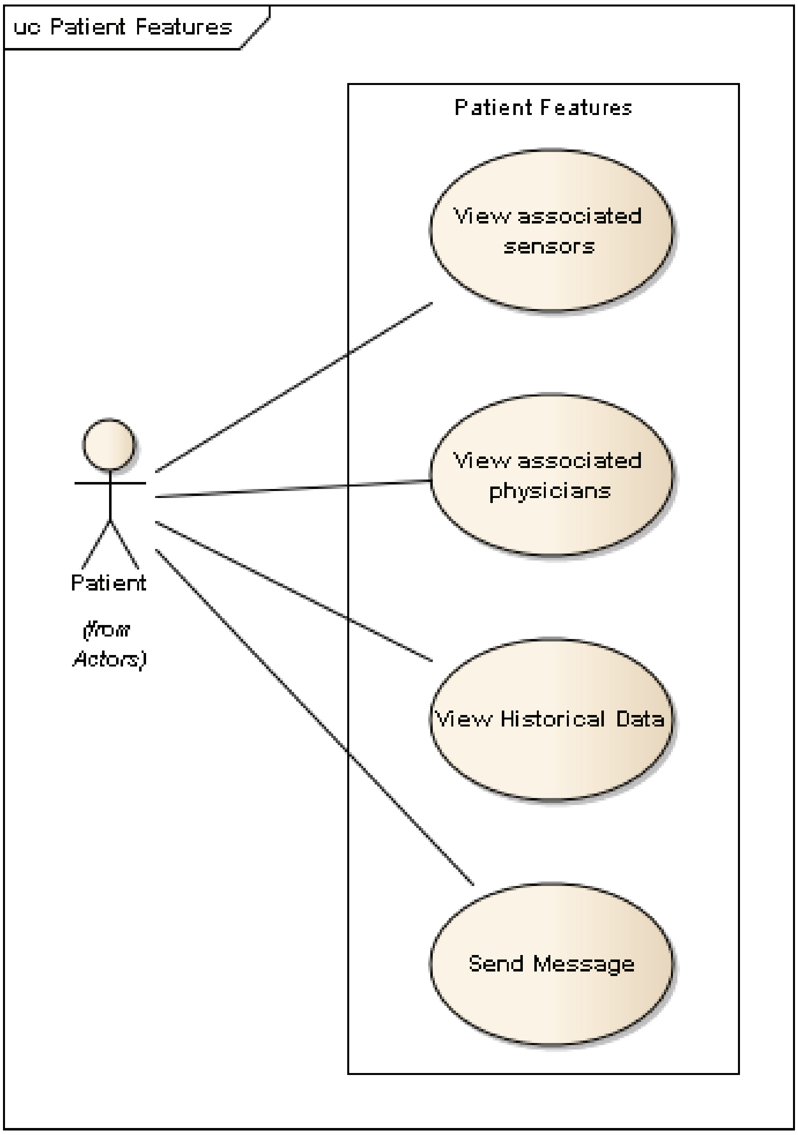

Figure 3.6: Patient Views

- **View Historical Data**: The patient can view his measurement's history and feedback messages, ordered by date.
  Priority: Low

- **View Associated Sensors**: Presents the list of sensors currently associated with this patient and for each sensor, presents the alarm frequency set for this device.
  Priority: Low

- **View Associated Physicians**: Shows the list of physicians who are associated with the patient, displaying their public details, such as name and speciality.
  Priority: Low

3.1.5 Non-functional Requirements

The following list presents requirements specifying non-functional requirements, i.e. a description of a property or characteristic that a software system must exhibit or a constraint that it must respect [Wie03].

- **Usability**: Given the characteristics of the mobile application users, it is mandatory for the application to be easy to use and learn. Usability tests and validation were made in order to comply with this property. Regarding the web application, although usability is not a major goal, its users do not have a technological background; therefore its processes should still be of easy comprehension. **Priority:** High

- **Reliability**: The simple fact that the system deals with health-related data, has a direct impact on the importance of the system to perform its required functions for a certain period of time without failing. Error-handling must be treated carefully and, in case of system error's, both the mobile and web applications should be responsive enough so the user is not led into misunderstandings and abandoning of the system. **Priority:** High

- **Security**: Also related with the fact of dealing with health-related data, comes security concerns. With a system as the one being developed, there are several points where security can be breached. The first one is inside the BAN (Body Area Network): between the sensor device and the user's mobile phone, a third-party could be listening to the measurements made by the users and send them to an unauthorized web server. Although this is plausible situation, as described in 2.1.6, at the moment of taking his measurements the user is supposed to be in a safe environment, usually at home; therefore little study is required regarding this situation. Another rising issue is about controlling the access to the web application. Patient's private data will be available on the web application and it should only be accessed by authorized entities; therefore, an authentication module should be implemented. At last and probably the biggest issue on security, is the transmission of data between the mobile application and the web server. Information will be sent through wireless networks over the TCP/IP protocol, where security can not be ensured, unless security measures are taken. The system must implement security protocols, in order to provide encrypted communication and secure identification of entities. For instance, the use of an HTTP-Secure protocol is a potential solution. **Priority:** High

- **Interoperability**: In the scope of this project, the mobile application is required to connect with pre-defined ANT+ enabled devices. However, it should implement the device protocols and profiles associated with each device. These are provided by the ANT+ Alliance [2], in order to give the system the interoperability to work with any device whose manufacturer is part of the ANT+ Alliance. **Priority:** Medium

[2] http://www.thisisant.com/ant/ant-alliance

- **Efficiency**: The mobile application will operate on a mobile device, with limited resources, such as power, storage and battery life. These limitations are a strong motivation for focusing on saving resources. Android recommends[3] a set of best practices for optimizing the system and during development these should be followed. **Priority:** Medium

- **Portability**: The physicians will access the web application through a network connection. As such, it should be accessible by different web browsers and operate in the same way for each of them. Regarding the mobile application, although it could be ported to a different platform, it will not be required within this thesis. **Priority:** Low

3.1.6 Assumptions and Dependencies

In order for the application to work as expected the following assumptions are made. To get started, the patient's mobile phone must operate on the Android operating system and support the ANT+ wireless communication technology. If this last requirement is not met, the application will not be able to connect to any device. It is also expected that each and every device supports the ANT+ technology and complies with the protocols and profiles established by the ANT+ Alliance, in order for the mobile application to set up a valid connection with them. These protocols are the link between ANT+ devices providing their interoperability. At last, it is expected the patient has an Internet connection to send his measurements to the web server. However, it is not required to have a full-time Internet connection, as the system will store the minimum necessary information until a connection is found. On the physician's side, an Internet connection will always be required so he can access the web application using a common web browser.

3.2 System Architecture

This section presents an overview of the system architecture, describing its domain model and underlying relationships, its physical and technological model.

3.2.1 Domain Model

The domain model presented on Figure 3.7 describes the various entities, roles and relationships of the system. More specifically, this was the model used for the web application. Because the mobile application deals with only one patient, the smart-phone user, its domain model is a subset of the model now being described.

To begin with, there is the user entity, which then specializes in the three actors described in 3.7: patient, physician and chief medic. Then there is a record entity which can be a message record, a blood pressure record or a weight record. The message record can be a message either sent from the patient to his physician or from his physician to him. A blood pressure record and a weight record represent patient readings taken from either one of the devices.

[3] http://developer.android.com/guide/practices/design/performance.html

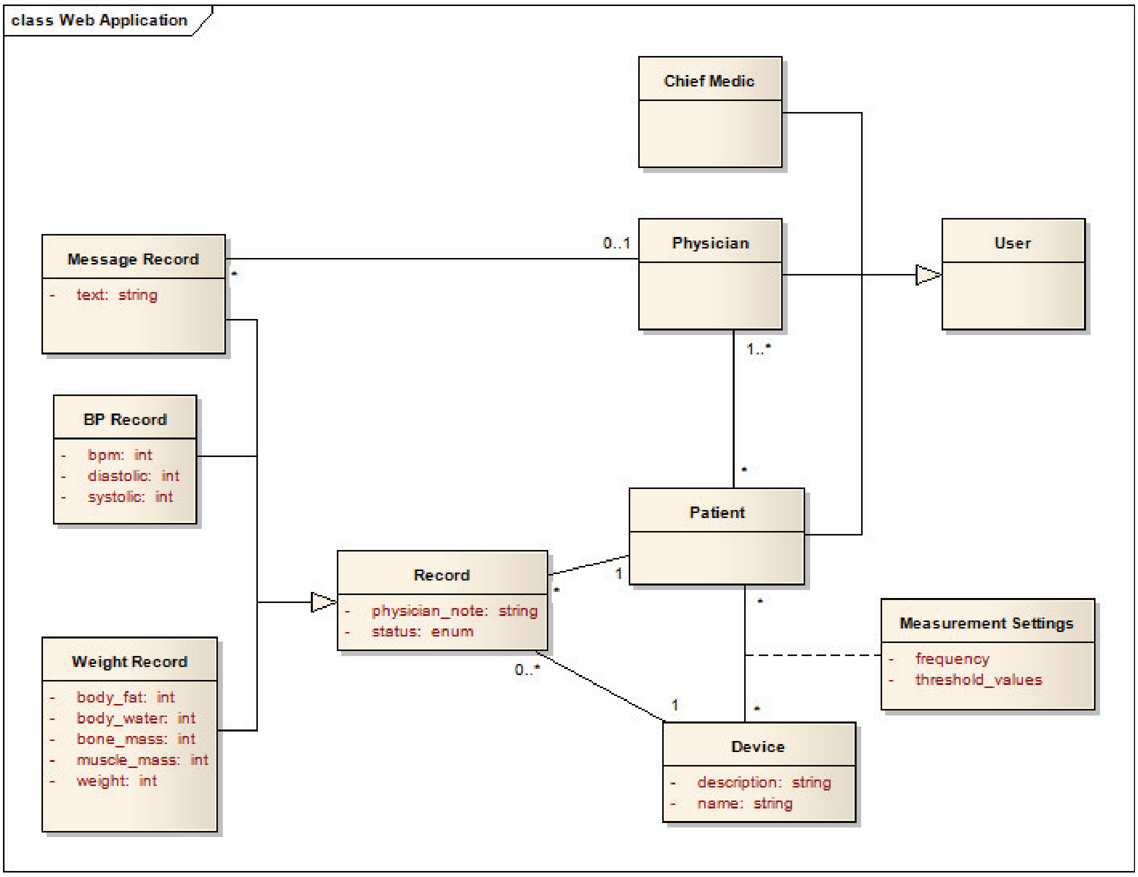

Figure 3.7: Domain Model

In order to represent all relationships between patient and physician, there is a many-to-many relationship with those entities, indicating that a patient may have several physicians and vice versa. Since each patient can have many records and every record belongs to a single patient, there is a one-to-many relationship with patient and record.

Regarding record messages, it was need to assure the following: when a patient sends a message, it is sent to all his associated physicians; when a physician sends a message, it is sent to a particular patient. Therefore the one-to-many connection between physician and message record. This way it is possible to find out which messages were sent by the patient's physician. If the message was sent by the patient, a physician will reach it through all messages of his associated patients. If it was sent by the physician, the patient knows who sent it by this physician/message record connection.

Each patient has a set of associated sensors and associated settings. These settings include the frequency with which the patient should measure with that device and also the threshold values assigned by his physician. To accomplish this feature, the patient entity has a many-to-many relationship with the sensor entity, together with the settings association class. Given that it was decided to use only two devices – blood pressure monitor and weight scale – the relationship between patient and sensor could have been simplified. However, for the sake of future improvements it was kept this way.

3.2.2 Physical Model

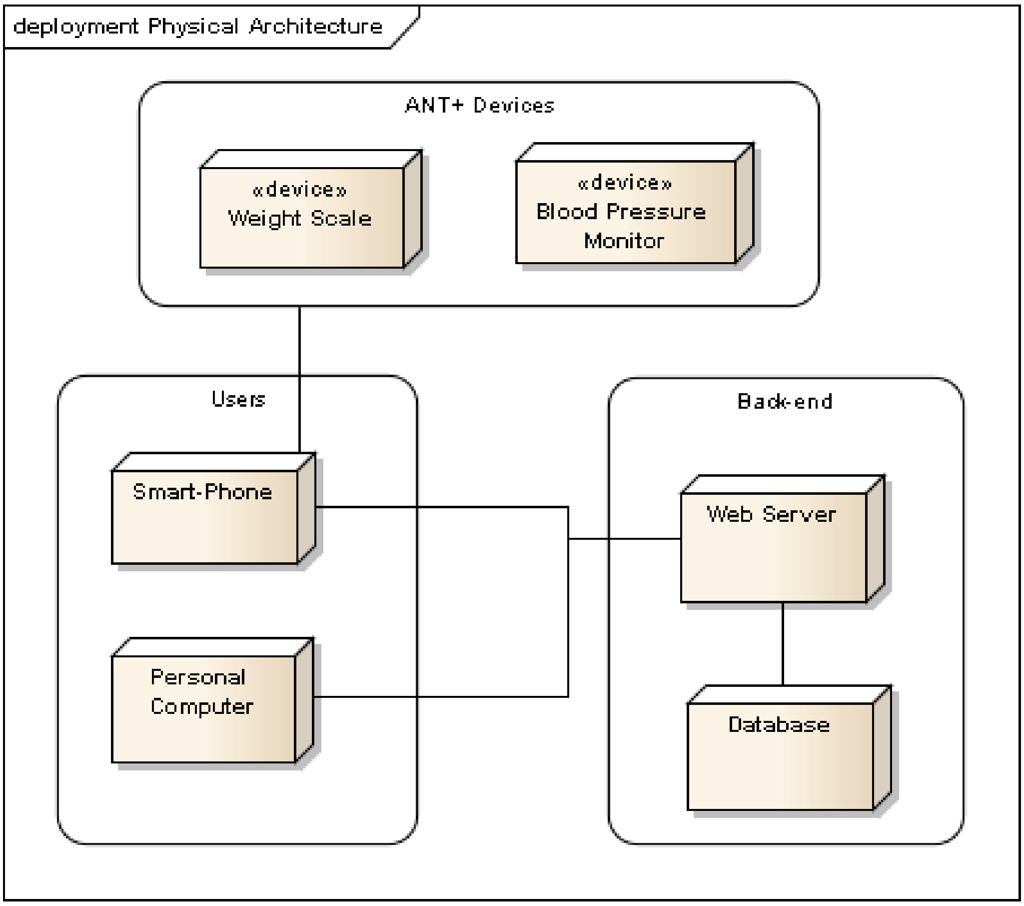

Figure 3.8: Physical Model

Figure 3.8 presents the physical architecture of the system, providing a description on how physical components will interact between each other. The diagram was grouped by: devices which solely interact with ANT+ technology, devices which remain on the back-end and devices used by patients and physicians.

On the users devices there is the smart-phone, which is used by the elderly and interacts with all available ANT+ devices – weight scale and blood pressure monitor. The other device on this group is the personal computer, used by either patients or physicians. Both users devices interact with the web server on the back end group, serving as the connection point between the system users. At last, the web server connects with its database in order to store and retrieve the system's persistence data.

3.2.3 Technological Model

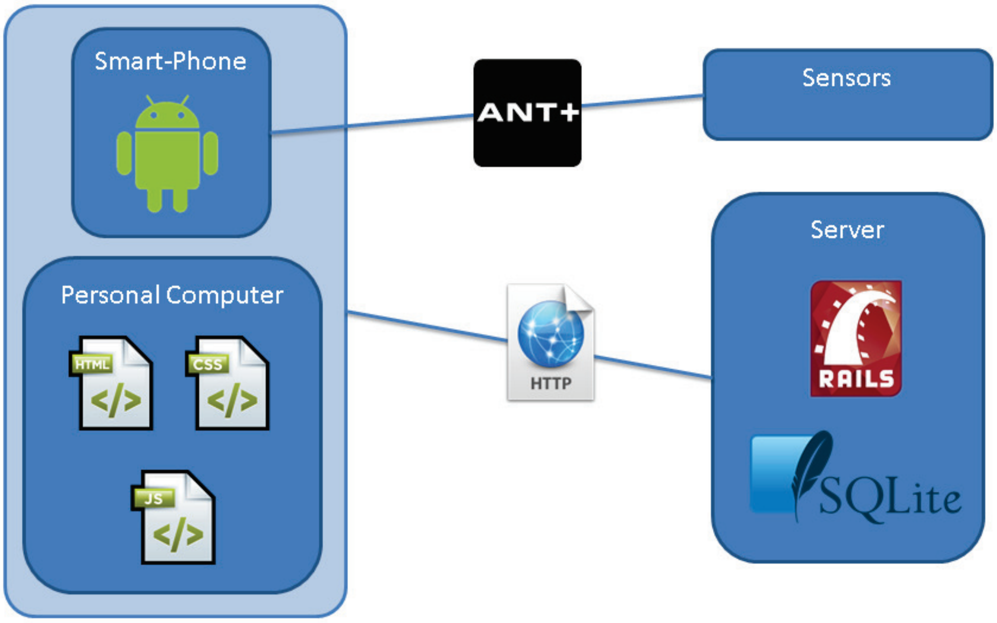

Figure 3.9: Technological Model

The technological model can be seen on Figure 3.9, giving an overview of all technologies used within the system. The smart-phone, used by each patient, operates under the Android operating system and connects with the weight scale and blood pressure monitor using ANT+ technology. Users accessing the system through the web application need a common web browser, interpreting HTML and CSS markup languages on the front-end and Javascript on the back-end. The web server operates under Ruby on Rails technology, connecting with an SQLite3 database. All users connect with the web server through the HTTP protocol – GET and POST commands were used.