Software Architecture Report

Team: FT_3
Version: 1.6
Authors:

- Bart de Man
- Felix Barten
- Joel James Bartholomew

- Spiros Tzavaras

Table of Contents

1. Introduction

The purpose of this software architecture document is to instruct the reader with the information needed in order to build a Quality Care Robot. The target audience are primarily less-abled users.
In the following chapters we cover the details of our created architecture, starting with the stakeholders of the system and those involved with their own interests, followed by the requirements which have been separated into non- and functional requirements. We cover the main functionalities and have designed several diagrams which can be found in the 4+1 chapter. Having selected several architecture patterns to use and made the right decisions for storage and monitoring. Having covered the components of the accompanied wristband for clear and direct specifications which will be used to further instruct other parties.

2. Stakeholders

2.1 Current Stakeholders

In order to find the necessary requirements and functionalities we have composed a list of stakeholders in our project. We have decided to split the stakeholders into small groups for better traceability of requirements. All the internal company stakeholders have the prefix QC. Our stakeholders for this project are the following:

(Less-abled) Product users

The end users of our product are people who are physically or mentally less able to perform household tasks. The goal of our product is to improve the Quality of Life of our customers. People who can still live at home instead of a full care nursing facility but have difficulties with performing certain (household) tasks. Our product should be able to alleviate the amount of work a resident has to perform or even reduce the amount of visits required per week from outside help. The concerns of the product user for this product are:

- Usability
- Reliability
- It should be easy for them to learn robot's functionalities
- Availability
- The robot should be secured from outside influences.
- The robot should provide security for sensitive data stored on it
- The robot must not hurt the product user.
- The user would like the robot to be a durable product. The following attributes are expected:
    - Water resistant
    - Shock resistant (falling)

Relative or family member

A relative or family member is a stakeholder in our project because they are likely responsible for configuring the robot (when the user receives one most likely). These relatives should be able to instruct the robot (See Logical view services) without much prior knowledge and should be able to customize the robot's settings, such as how the user wants their coffee.

- Usability
- Modifiability of behaviour

QC: Quality Assurance Team

Quality assurance will be involved with the whole development of the product to ensure the product adheres to the requirements, to test and improve the entire development process.

- Testability
- Well-defined requirements
- Making a better product

QC: Software Testers

This group of stakeholders might have insights to make sure the software of the product is and remains testable.

- Clear details about QC-Robot's functionality and design
- Testability (for the whole product and for the separate modules)

QC: Hardware Testers

This group of stakeholders might have insights to make sure the product is and remains testable.

.

- Clear details about QC-Robot's functionality and design should be provided
- Testability

QC: Developers

The developers will be responsible for creating the software behind the robot. The developers are a stakeholder for this project because they would like a well-maintainable software architecture which can be maintained and modified in the future without major overhauls.

- Modifiability
- Maintainability

QC: Researchers

The researchers will be conducting research that will clarify uncertainties surrounding the product.

- Clearly specified requirements
- Subjects for research

QC: Hardware Specialists

The hardware specialists will be responsible for designing the hardware for the robot.

- Clear specifications about the functions hardware should be able to perform

QC: Managers

Management wants a profitable product and to preserve the continuity of the company.

- Profitable product
- Continuity of the company
- Preserve company's status

QC: Investors

Investors are the reason our company is able to conduct research and build products. As Quality care is a small company we need a lot of outside funds to get our company off the ground. Investors have a large stake in our first commercial product as it will indicate if they will be rewarded for their investment in our company. The main concerns of the investors are:

- Schedule and budget estimation
- The continuity and profits of the company

QC: Software Architects (FT_3)

The architects want the ability to design a well-functioning system that properly addresses the different concerns and interests of each stakeholders.

- Clear instructions and requirements
- Satisfaction of stakeholders

Doctor

The doctor will administer the medicine required by the patient and track their health. They will be able to administer medicine and adjusting or adding prescriptions from their office using a web interface. Less need for the less-abled to visit, thus saving time and money.

- Usability

QC: Software maintainers

The software maintainers will be responsible for maintaining the robot’s software after development. While the people that perform this task may or may not be the same people that developed the software in the first place their concerns are different.

- Platform compatibility
- Maintainable system
- Modifiability
- Testability

QC: Service engineers

Service engineers are a stakeholder in this project as they will be responsible for servicing the robots after launch of the project. The service engineers would like the robot to have easy to replace parts so the replacement of certain components is as easy as possible. The service engineer can use his/her experience to influence the development to make hardware that is easy to service. The service engineers’ main interests are:

- Remote analytics of the status of QC robots
- Modular Hardware which can be replaced easily
- Easy (dis)assembly of the product
- Clear maintenance instructions

QC: Customer Support

It’s the role of customer support to help any users of the product in case they have trouble with the product. The customer representatives will provide telephone and email support for customers who are having difficulties with the product.

- Well-documented instructions for troubleshooting
- Clear documentation of QC-Robot

Health Insurance Providers

Health insurance is a main stakeholder for our product. Health insurance companies may be able to save money if the product is cheaper (and thus saves them money) than care provided by humans. This will allow medical personnel to perform less mundane tasks and focus on other aspects of care.

Health insurance companies might subsidize robots if they want people to transition to other types of care. This method would incentivize end users to buy the product if a part is refunded. Or alternatively the insurance company may offer the robot as a substitution of human care.

Because of the position of the health insurance companies these stakeholders will be viewed as customers in this case.

- Profit. A robot is cost effective alternative to real personnel and they would like to make more accessible for users in need of help
- Security
- Robot should not harm patients
- Licensing of robots to be able to provide it to many users

External Manufacturer for the Hardware

The manufacturer has to make sure the product can be produced once the designs are finalized.

- Clear specifications on the hardware needed for the product
- The design provided by the designers at QC should be realistic and be able to be manufactured by the designated hardware manufacturer

Regulatory Authority (Nederlandse Zorgautoriteit(NZa))

The NZa is a stakeholder in this project because the product will be used in a medical fashion and can have effects on public health if it is not deemed safe. To make sure the product will comply with the latest rules and regulations the NZa is a definite stakeholder in this project.

- Product should be safe for use and should comply with all health care regulations
- Security of medical data

Distributor

The distributor will deliver the product to retail stores or directly to the customers

- Distributors want the product to be easily deliverable

Third party developer

Third party software developers will add additional behaviour to the robot's default behaviour. With the help of third party developers the robot could provide extra methods of care or more social interaction with main users of the system (See logical view). Third party development is also a good way of gaining additional exposure for the product if a lot of interesting new functionalities are developed. However, these additions need to be curated otherwise the product will not be safe for the public anymore as external developers might not adhere to the health regulations which the product's programming is bound by.

- Modifiability of the code or rather good interfaces/API’s to interact and extend the existing system.
- Interoperability with other systems

2.2 Future Stakeholders

Third party addition curator

The third party curator is a future stakeholder of our project. The 3rd party curator’s job will entail checking packages to ensure the public’s safety. This stakeholder is not listed in the current stakeholders because it is dependent on the adoption rate of the project. If the project is widely adopted after being completed it is likely more third party development will occur which will lead to more packages that need curation.

- Modifiability
- Testability
- Supportability

3. Requirements

After deciding on the list of stakeholders we have created a list of requirements in order to give our architecture more meaning. These requirements are split in functional and nonfunctional requirements. These requirements were used to formulate the main functionalities.

3.1 Functional requirements

1. Users should be able to customize how they like their meals
2. Users should be able to customize the reminder settings
3. It must be able to open doors and receive guests
4. Unit must be able to notify the user that it is in need of servicing
5. Unit must be able to analyze faults
6. Unit must be able to send diagnostics back to service engineers for analysis
7. Unit must have modular hardware
8. Unit must be easily disassembled
9. Unit must have a (maintenance) instruction manual
10. Unit must have facial recognition
11. Unit must be able to recognize voice commands
12. Unit must be able to recognize hand gestures (sign language) commands
13. Unit must be able to be controlled remotely via a remote.

14. Unit must be able to be configured
15. Unit must have a display to be able to accommodate deaf or near deaf people
16. Unit must be able to respond to events within 0-10 seconds
17. Unit must be able to move around at a reasonable pace
18. Unit must have a simple user interface
19. Unit’s behaviour can be modified from extra software defined by third parties
20. Unit must be able to choose what the next task is that it will perform (scheduler)
21. Unit must be able to show its status (either by voice or a graphical representation)
22. Unit must be able to send and receive secure video from and to QC headquarters
23. Unit must be able to store medical information about the user.
24. Unit must be able to detect heart rate of the user
25. Unit must be able to measure blood pressure of the user
26. Unit must be able to take the user’s temperature
27. Unit must have a bluetooth connection
28. Unit must have a WiFi connection
29. Unit must have a microphone
30. Unit must have motion detection sensors
31. Unit must have an inventory of available medicine
32. Unit must be able to order additional medicine if the medicine runs out.
33. QC Control Centre employees must be able to view a secure video from the robot
34. The product must be able to monitor health information such as heartbeat and blood pressure

3.2 Non-Functional requirements

1. Unit must comply with health and safety regulations
2. A detailed description of unit’s functionalities
3. Unit must be water resistant
4. Unit must be shock resistant
5. Unit must be assembled from components that are able to be tested individually
6. Unit must have replaceable parts
7. Parts of the unit must be able to be recycled
8. There must be documentation to give information about the robot
9. User must be able to get acquainted with the basic robot functionalities in less than 15 minutes
10. Unit’s power consumption must be minimized where possible

4. Main Functionalities

After documenting the stakeholders of this project and its requirements, it was decided to create a list of functionalities for the robot. These functionalities are focussed on the end user and should all raise the quality of life for the user of this product.

1. Must be able to provide the main stakeholder with aid in case they have fallen down
2. Must be able to remind users of (important) subjects/appointments
3. Must be able to respond to a doorbell if the main stakeholder may not be physically able to respond to the door bell
4. Must be able to make meals and/or drinks available. The resident may not be able to cook for him/herself
5. Must be able to clean the house using a vacuum cleaner, thus removing the need for (extra) personnel to do so.
6. Must be able to administer medicine to the resident and keep the medicine stocked via prescriptions.
7. The QC Robot will show (secureline) video in the QC control centre located in the (nearby) area.

5. 4+1 Architectural View Model

5.1 Logical view

Below a class diagram is provided, that represents the functionalities that QC-Robot provides to the system. The logical view has been designed in such a way that the end user can understand the workings of the robot.

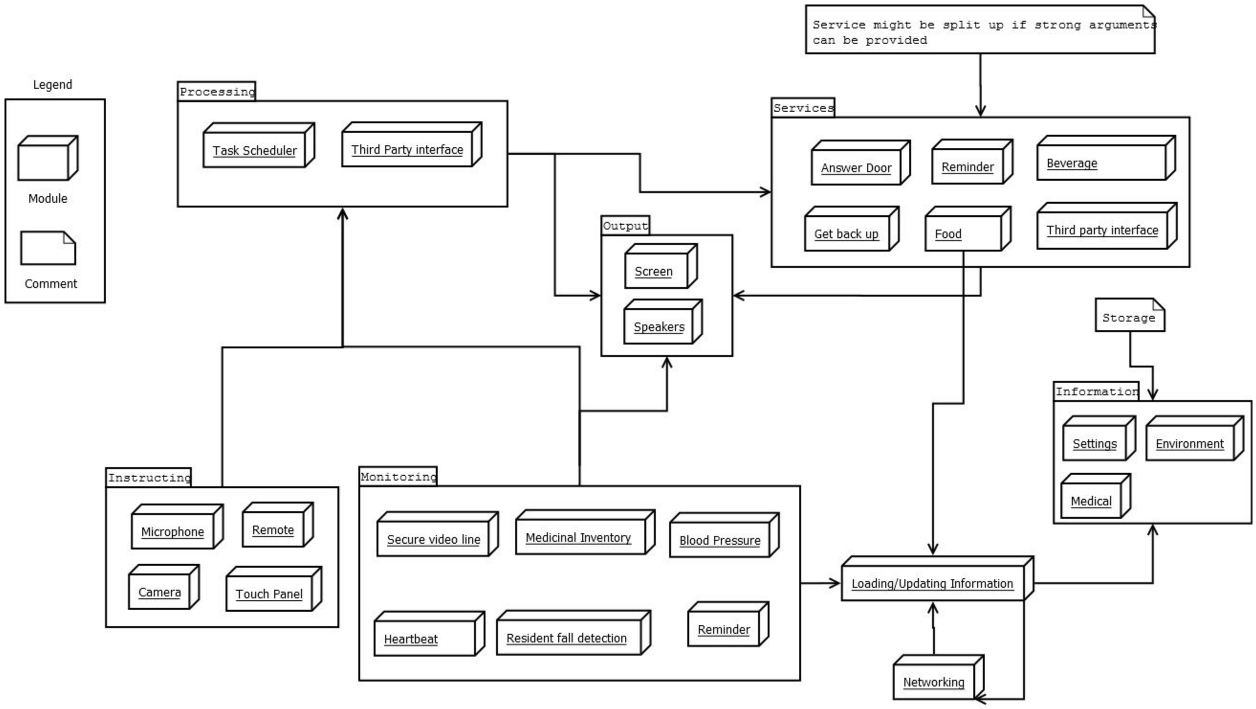

Diagram 1: Logical view

[View image on github]

5.2 Process view

As for the process view, a sequence diagram has been created, in order to show how QC-Robot handles events, and how each of the components acts in a process.In case of a critical failure, which might occur in any time while invoking one of the components, a relevant message is shown to the user.

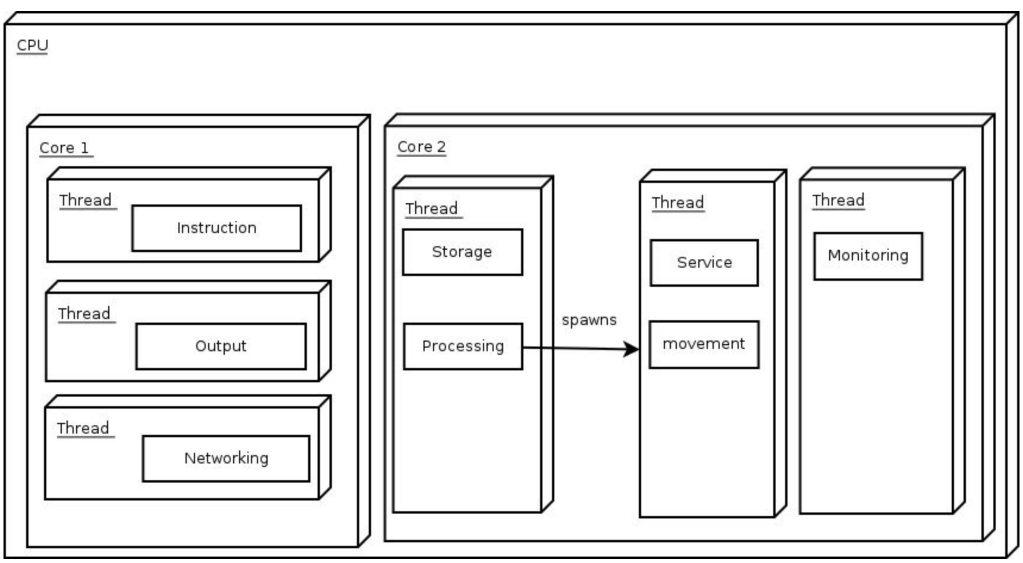

Diagram 2: Process view

[View image on github]

5.3 Deployment view

The diagram describes the software components of the system, as well as the way these are connected to each other.

The deployment has the following main elements:

1. A Quality Care sensor for monitoring heartbeat and blood pressure
2. A remote control for the Quality Care robot
3. The Quality Care robot itself
4. A web-service which allows the robot to make its video feed available to the Quality Care control centre

There is also a service which provides access to the health information of the primary user of the robot.

The robot has the following software components:

1. Instruction Module - The user interface which has several sensors which the user can use to issue commands. Also the robot will react to detected signals of the sensors to autonomously execute actions.
2. Processing Module - This module validates the input provided by the instruction module and calls the service corresponding to the command issued
3. Service Module - Contains the main functionalities of the robot.
4. Local information Storage - Contains information relevant to the robot’s settings and user information
5. Monitoring Module - Used to monitor the product user’s health metrics
6. Networking Module - Allows the robot communicate over network connections
7. Movement Module - Used by other modules to allow the robot to move
8. Output Module - Used to provide feedback (information) to the user via the speakers and the display screen.

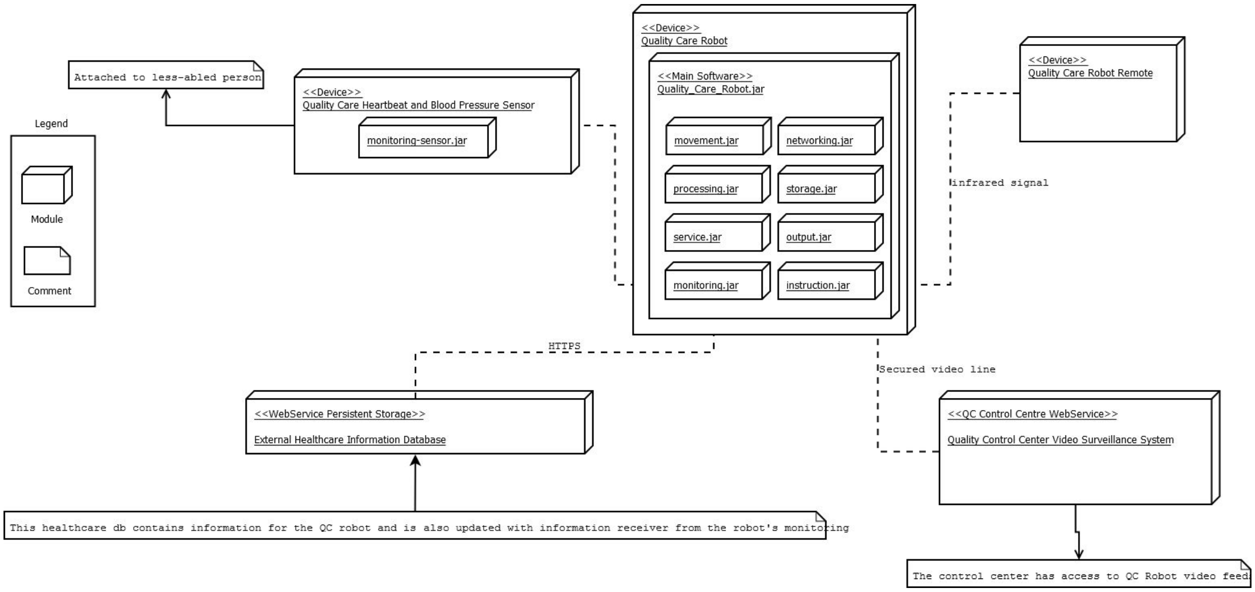

Diagram 3: Deployment diagram

[View image on github]

5.4 Implementation view

The structure of the code, the directories and files of the system, the environment and in general everything that concerns the programmers of the system, is represented in the following diagram.

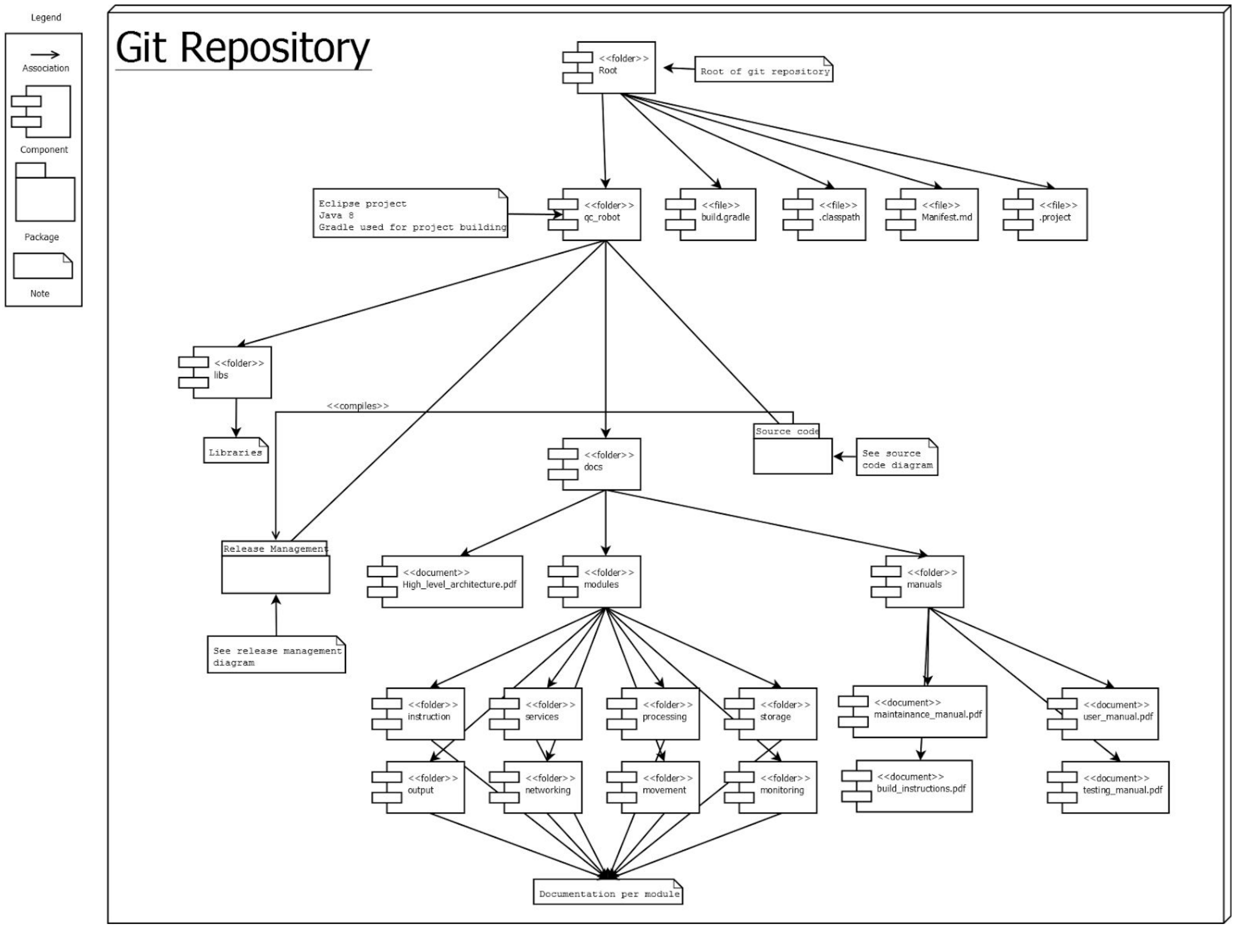

Diagram 4: Implementation View

[View image on github]

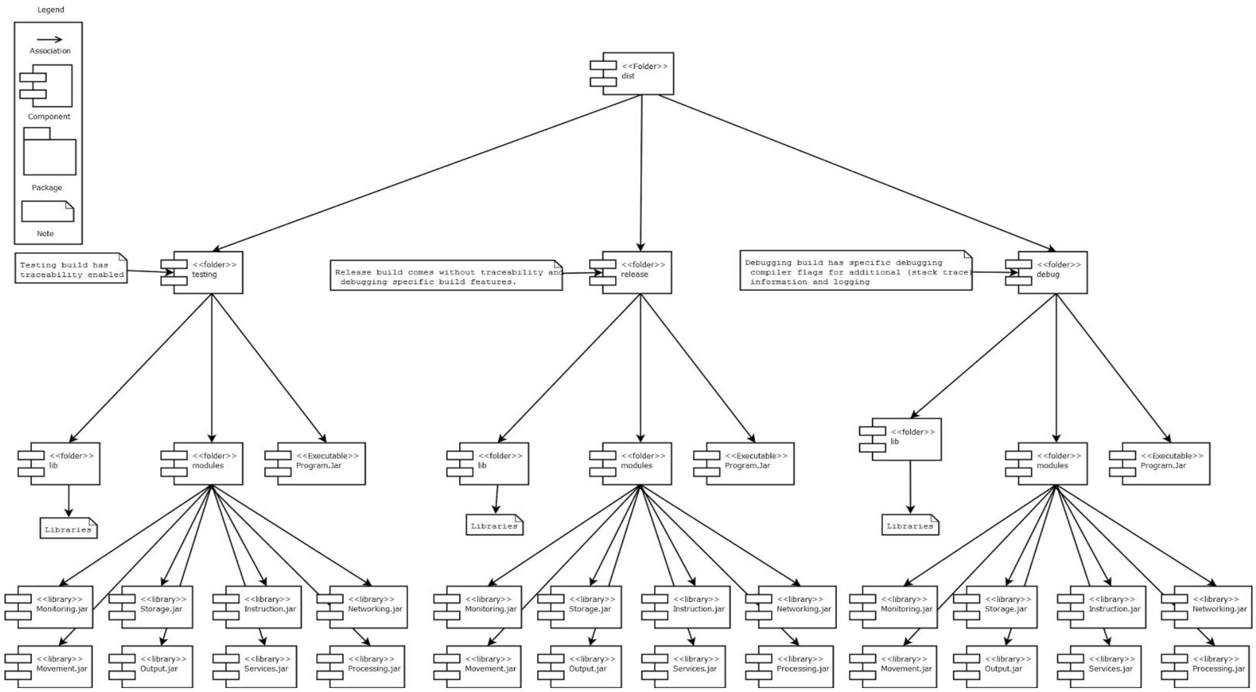

Diagram 5: Release management diagram

[View image on github]()

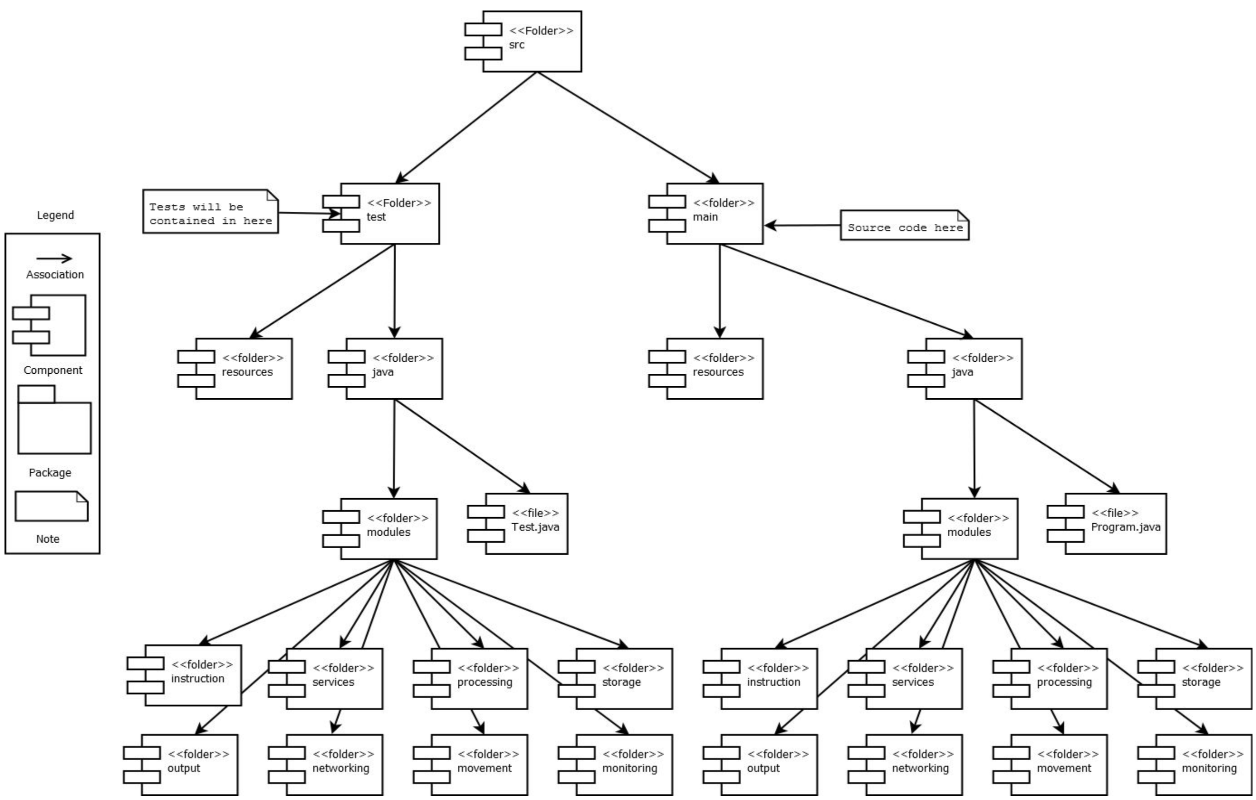

Diagram 6: Source folder

[view image on github]()

5.5 Use case view

Below we can see the interactions between the main stakeholders and the processes of QC-Robot. Two of the use-cases below are attached with an automated process which checks if actions are required.

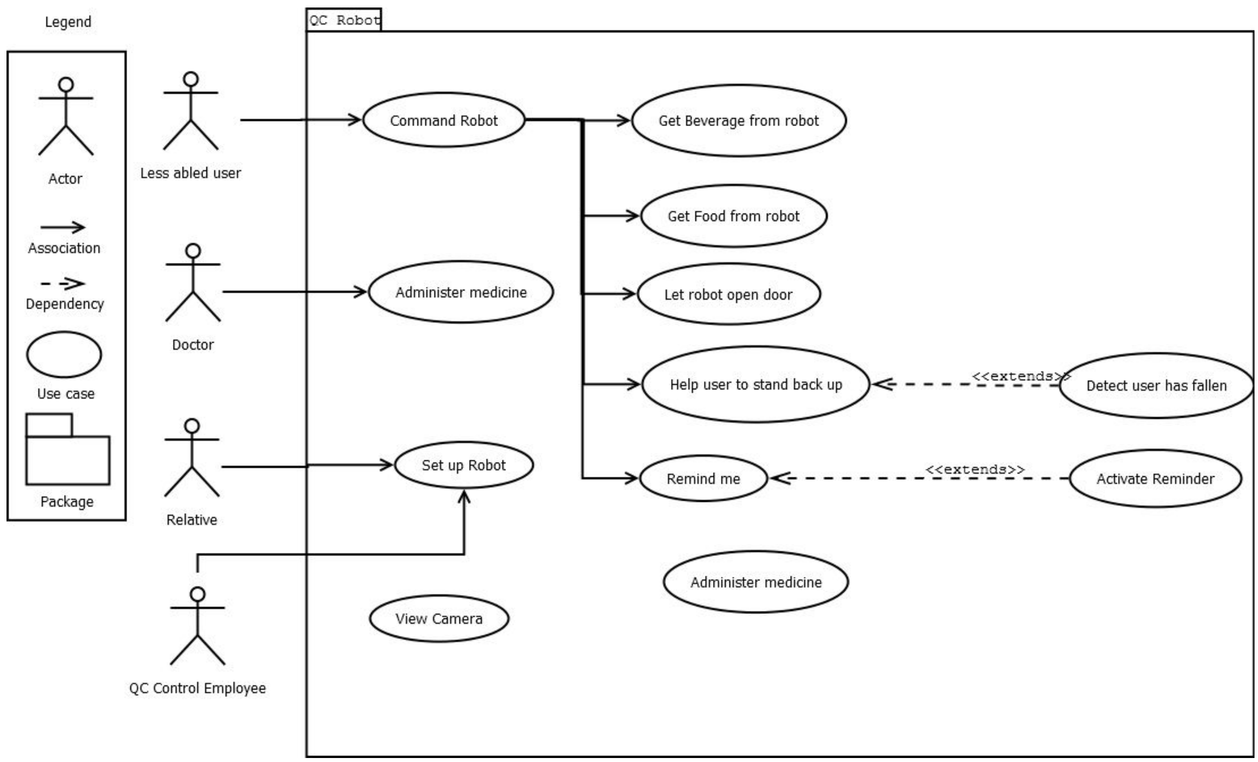

Diagram 7: Use case diagram (also called scenarios in context of 4+1).
[View image on github]

6. Architecture patterns

Except for the pipe-filter pattern, we have studied a couple of architecture patterns that may or may not be useful in the creation of QC-Robot's architecture. By listing these patterns and weighing in their pro's and con's, we can make the right decision in order to apply the right pattern(s). First, we mention some patterns that we believe could fit QC-Robot's architecture:

1. Client-Server pattern. Even though server side could affect system's performance in a really bad way, this pattern could easily be used to allow the QC control centre to access the video feed of a QC robot. Each robot will be able to communicate with the QC control center to make its video feed available.

2. Layer pattern. We decided to incorporate a layer architecture in our system to allow the different modules to be developed independently and to ensure that the system will be able to adapt to any future changes. Modules offer a cohesive set of services, where each layer can use any other layer that is below it, on the same level as it or to the right of it. The structure of this pattern is shown below.

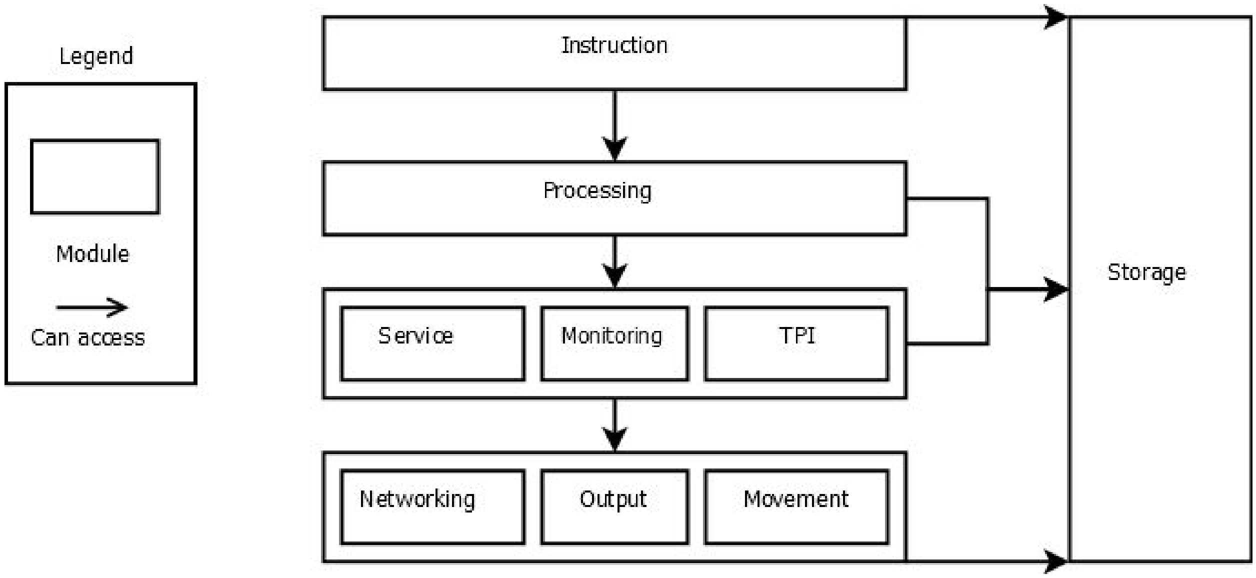

Diagram 8 : Layer Pattern

View image on github

3. Model - View - Controller pattern. This pattern could easily fit in our architecture, and provide our system with some certain advantages. One of the main advantages for example would be that it would make our system more flexible, compared e.g. to the layer pattern. There is a need to mention though, that complexity may not worth it, because we want the User Interface to be as simple as possible, as mentioned in the system's requirements.

Now, we mention some patterns that we do not believe would appropriately fit QC-Robot's architecture.

1. Peer-to-peer pattern. There is no need for the QC-Robot to communicate with the control centre in a peer to peer fashion. The target audience of the QC-Robot is not solely elderly people and the robot is more focused to helping those who are living on their own. This makes it unlikely that multiple QC-robots would ever come into contact with one another. The client server architecture is a more optimal solution.

2. Multi-tier pattern . Although this pattern could somehow fit QC-Robot's architecture in order to fulfil some requirements of the system, such as its reusability, it is more suited to components running on different systems and so layering pattern would be preferable.
3. Broker pattern. This pattern would not fit for our robot's software as the robot's services would all be contained within it's own software and not on external web servers. The broker pattern creates an intermediary object to interact with these external services and creates an abstraction between itself and the other layers because those layers don't know which service they're communicating with. However this pattern would simply not apply to our current architecture.

7. Quality Attributes

For our first part we have decided to focus on the security of the storage part of our system. The storage part of the robot is responsible for saving and reading various types of information. Among these types of information some parts are really privacy sensitive and for other types of information a less secure type of storage can be utilized.
For the second part we have chosen to focus on the monitoring module of QC-Robot. This part of the system is very important, as it is related to the monitoring of the end user's health and the actions that need to be performed when necessary.

7.1 Storage decisions

For documenting our storage decisions we have used the template proposed in the paper written by Tyree and Akerman[1]. Our architecture decisions in week 4 are centered around the storage part of the robot. As our robot deals with different types of information (medical, environment and settings) we have chosen to focus on the medicinal information first as this type of information storage poses the largest architectural challenge out of the three types (diagram 10).
To store these types of information a separation of data is made beforehand. All medical data will not be stored in the same storage component as the settings and environment data. This division is made because there are very different security constraints for the medical data than the other types. It is also assumed that the storage will not be directly attached to the internet, although information can be exchanged with the computer infrastructure at the QC Headquarters not every bit of information stored on the robot will be readily accessible from the outside unless there is a valid reason to exchange data.

Storage decision #1

Issue	A security measure needs to be taken to prevent third parties to be able to access medical information on the robot
Decision	Using (HIPAA) compliant encryption mechanisms to encrypt the data stored on the robot.
Status	Approved
Group	Storage
Assumptions	• It is assumed that medical information such as vital signs, prescription information and patient medical history need likely to be stored on the robot. • It is assumed that there are rules and regulations regarding the storage of medical data. These rules and regulations are determined by the regulatory agencies. • Regulatory agencies require encryption of medical data whenever it is stored. • Due to privacy concerns relating to the data it is assumed the encryption algorithm that is prescribed by the regulatory agency will have a negative performance impact.
Constraints	If the decryption key is ever lost the information would become irrecoverable
Positions	**Store sensitive data outside of the robot** The data could also be stored elsewhere (not on the robot). This would shift the priorities of securing hardware on the robot to securing a service in a data center and thusly securing the data center itself as the data should be protected. Why not: the approach above would work for our robot as it would save costs developing mechanisms for security and hardware but simultaneously would cost a lot more in data center and (physical) security where it might not be worth it to invest in this alternative. Furthermore, by moving the data outside of the robot, one adds an extra constraint to fetching the data, which is that when there is no internet connection on either party, there is no data.Thus only moving the problem of securing the data.
Argument	Personal data such as health information is of big importance and so preventing accessibility from others is necessary.
Implications	Using (strong) encryption usually results in a lowered performance due to encrypting and decrypting the data while using the storage

	device.
Relation Decisions	-
Related Requirements	The storage of medical information.
Related Artifacts	Hardware and software storage mechanisms
Related Principles	This decision affects the Performance, Reliability and Security of the system. CIA
Notes	Architecture Tactics: Resist attack: Encrypt data, Limit access, Limit exposure
Quality Attribute Scenario	Source: An unauthorized attacker Stimulus: An attack Artifact: Medical Information Environment: Fully operational Response: Encrypted Data Response Measure: The amount of health data comprised should be 0%

Storage decision #2

Issue	How to restrict access to the storage and who should be authorized actors?
Decision	The only authorized actor should be the doctor. He is the only one responsible for tracking user's health status, and updating some information if needed. See diagram 9
Status	Pending
Group	Storage
Assumptions	• The stored information must be able to be permanently (destroyed/erased) somehow. The nature of the data requires that it be possible to permanently remove the medical information from the system. (E.G. corrupting the sectors using a magnetic field) • There will be data encryption • The data should be readable (accessible) and writable. • A service engineer should be able to replace the storage unit in case of damage but not have access to the actual data.
Constraints	The CRUD operations that update the user's health information must be automatic. This means that, there should be algorithms

	implemented, that check user's health status. (e.g. calculating average blood pressure for last days, and updating the dosage if necessary) By only allowing the doctor to access the medical data, software which allows a unique login is required, specified for only said involved doctor. To have such a login, there is already existing software such as OAuth which can be easily implemented.
Positions	**More stakeholders can have access to the information** An alternative way would be that more than one of our stakeholders would have access to the information. Why not: Although, this could be a positive aspect for the user, (i.e. if he really doesn't feel well he would be able to take one pill more than he should), but it is a fact that such actions should be approved by the doctor first.
Argument	This way, there is no need for actors apart from doctor to make CRUD operations on database
Implications	Doctor is able to use a simple UI to collect/update data. The health information stored will have software security that ensures that only authorized users have access to this data. In this case a medical professional such as the main stakeholder's doctor.
Relation Decisions	The choice of encryption (storage decision 1)
Related Requirements	The storage of medical information
Related Artifacts	Hardware and software storage mechanisms
Related Principles	Confidentiality, integrity and availability (CIA)
Notes	Architecture Tactics: Resist attack: Iden/Auth/Auth actors
Quality Attributes Scenario	Source: An unauthorized human Stimulus: unauthorized attempt to access data Artifact: Medical information Environment: Fully operational Response: The failed access attempt gets logged Response Measure: The amount of actors who were correctly denied access

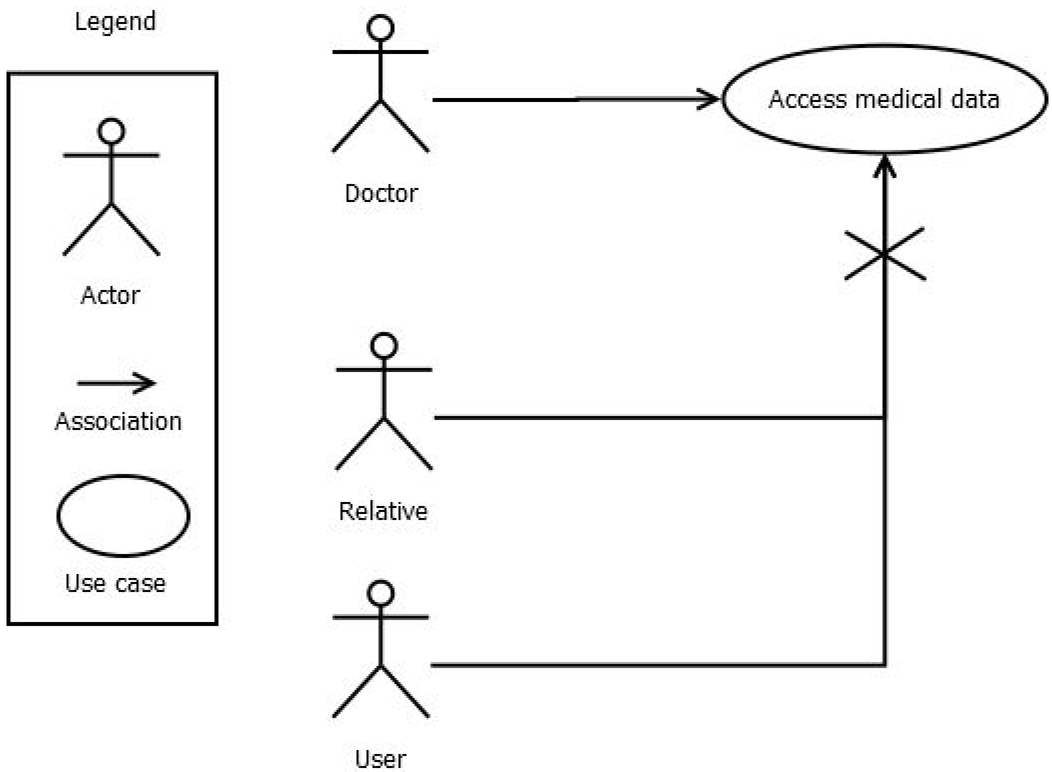

Diagram 9: Use case diagram of accessing medical data stored on robot

View image on github

Storage decision #3

Issue	Should the system use only software or hardware level security?
Decision	Usage of hardware security and software security.
Status	Pending
Group	Storage
Assumptions	The system must be secured on at least one level
Constraints	-
Positions	**Secure information on a hardware level** Secure the health information storage on a hardware level so that it is only possible for someone specialized from Quality Care to remove it. Why not : See below

	Secure information on a software level Secure the health information storage on a software level so that only authorized individuals can access it. Why not: Having redundancy by combining more actions will lead to a more reliable security system due to their not just being one point of failure.
Argument	A combination of hardware and software level security is more effective than just one or the other. The two levels will complement each other and if one is comprised it does not necessary mean that the system is breached. Using an encryption protocol will ensure data security if it is properly implemented.
Implications	It will have a positive effect on the security of the system without too much of a loss to usability.
Relation Decisions	-
Related Requirements	The storage of medical information
Related Artifacts	-
Related Principles	Confidentiality, integrity and availability (CIA)
Notes	Architecture Tactics: Detect Intrusion, Separate entities, Inform actors (in case of attack), Revoke access
Quality Attributes Scenario	Source: End user Stimulus: Authorized attempt Artifact: Medical information Environment: Fully operational Response: System provides access to data (only view) Response Measure: After 5-10 seconds user is having a view of requested data

Storage decision #4

Issue	The composition of the hardware for storing sensitive health information. (See diagram 10)
Decision	The health information storage will be separated from the rest of the information the robot needs to store as it contains very personal

	information. It will have its own separate piece of hardware that is difficult to access with the proper procedure. The system will “revoke access” (The data on the drive will be deleted) to the storage unit in the case of an attempt at unauthorized access to the device.
Status	Pending
Group	Storage
Assumptions	- The robot will be connected to internet but this specific part will not have direct internet access
Constraints	The component will be more difficult to service.
Positions	**Hardware level drive encryption.** Why not: This position was rejected because it did not provide adequate security compared to the chosen position. While this position would not have any of the downsides of the chosen solution it would not have provided satisfactory security. Hardware (and software) level checking for tampering with the system. Why not: although this position in itself is a good way of ensuring hardware integrity it alone is not enough to provide adequate security. Instead it was decided to implement this as part of the whole solution (see storage decision #4). **Storing health information in the same hardware as the rest of the robot’s data.** Why not: it was decided to store data on different storage devices because the security constraints are much more strict for the medical data and it would not permit the types of data to be mixed on a single storage device, therefore this position cannot be used. Also the other types of data stored on the robot will be accessed more often than the medical information and would benefit from increased performance (by not having to adhere to security constraints)
Argument	The medical data stored on the robot should not be accessible via means of a simple harddisk component. As this would open the robot up to attacks if the robot was ever disassembled or captured. Therefore it was chosen to encapsulate the storage device in an encasing hardware component to make sure the data (even though it’s encrypted) cannot be accessed during maintenance. Instead the hardware component containing the storage device will be able to be swapped as a whole by the service engineer instead.
Implications	Due to the security concerns there has been more emphasis on the protection of stored information at a hardware level. While these hardware changes make the robot easier to service (due to the

	swappability of the secure storage component) but harder to test and maintain due to the black box nature of the secure storage component. Although a downside of checking for hardware tampering is that the system that checks for potential tampering should also be monitored against hardware tampering so a redundancy must be built in to safeguard against these types of attacks.
Relation Decisions	Storage decision #3
Related Requirements	The storage of medical information
Related Artifacts	Hardware and software storage mechanisms
Related Principles	Confidentiality, integrity and availability (CIA)
Notes	Architecture Tactics: Separation of Entities, Revoke Access, Detect Intrusion
Quality Attribute Scenario	Source: Human Stimulus: Unauthorized attempt to directly access health information hardware Artifact: Medical/Health information Environment: System non-operational and dismantled Response: The hardware will delete the stored information Response Measure: A data comprise of 0%

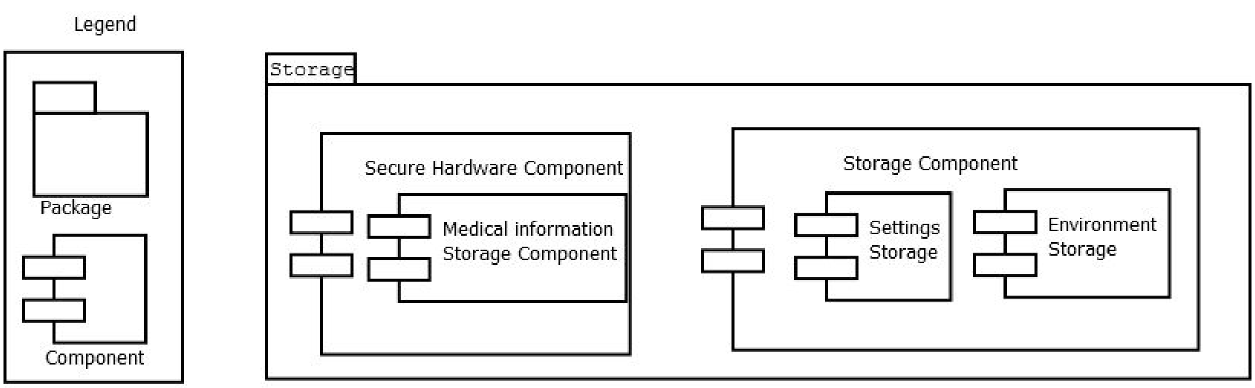

Diagram 10: Component diagram for storage part

[View image on github]

7.2 Monitoring Decisions

The second part of which the design decisions will be described is the monitoring part of the robot. The primary function of the monitoring part is to measure the life signs of the resident and to act on the measurements if necessary. To maximize the usability of the robot measurements should be able to be performed automatically as much as possible without the resident having to perform manual steps.

For the monitoring part of the robot the following quality attributes are very important:

- Maintainability
- Reliability
- Portability

Falling detection #1

Issue	How is the QC-Robot going to detect that the user has fallen down
Decision	To detect the fall a number of sensors will be utilized: - A camera - A microphone - An accelerometer
Status	Pending
Group	Monitoring
Assumptions	● It is assumed that the robot can navigate around the house ● It is assumed that the robot will use a camera to direct its movements ● It is assumed that the robot has a working microphone ● It is assumed that the robot has speakers ● It is assumed that the decision to build or buy the three components above has already been decided in another design decision. ● It is assumed that the user’s heart rate will need to be measured based on earlier functionalities. To accurately measure heart rate a device attached to the user is required.
Constraints	That this system is only usable for users who are capable of interacting with the systems using these mechanisms (See design decisions)

<table>
<tr>
<td>Positions</td>
<td>
Camera system installed in the entire house

A camera system installed throughout the house which the robot can retrieve information from. The cameras would be installed in all major living areas of the house except for some rooms, because of privacy concerns(e.g. the bathroom). The robot can use the information provided by the cameras to navigate around the house.

Why not: A camera system throughout the house is a very intrusive method of monitoring whether the user has health issues. These cameras would be a cause for concern in the privacy department. The privacy of residents could be violated if certain parts of the house were under 24 hour surveillance. Using this position would heavily influence cost because of the added costs for both the installation and the buying of the camera system. Furthermore it would cause problems if a user had any health troubles in an area where there is no surveillance. This would render the robot useless as it would be unable to navigate to the user to help him or her. Another side effect of this method of monitoring is that it might discourage the purchase of the product due to privacy concerns.

To summarize this position was rejected because;

- Serious privacy concerns
- Less functionality (See: “the why” not section) for the robot
- Higher barrier to entry (high costs)

Camera attached to the QC-Robot
The camera attached to the robot is a good idea, because that way the robot can go out and investigate when the user is in trouble. Rather than have a static position for the camera, which would mean if the user needs to install cameras everywhere in the house which would increase the cost for the total system even further.
The robot has the ability to move around the house with the camera. As the robot is in control of it’s own camera movement it can adjust the camera for more viewing angles.

Cons:
- This is not sufficient because the robot will not always be in the same room as the main stakeholder.

Device attached to the main stakeholder.
By having an external device attached to the main stakeholder the robot can determine whether it needs to check up on the main stakeholder when it’s not in vision of the camera. There will also be (a) button(s) on this device, so the main stakeholder can call for help through the attached device.

Cons:
- The user may not be physically able to use the device
</td>
</tr>
</table>

	Thermal camera There could be a thermal camera attached to the robot which could monitor changes to the environment on the floor level which could be used to detect the someone falling down. Why not: A thermal camera would not be beneficial in this scenario. A thermal camera is a camera that captures heat signatures via infrared. However this functionality would not provide enough added benefit to warrant the (excessive) costs this type of component brings. And the camera that is included already provides most of the required functionalities. Pros - Would make detecting humans on the ground or partially obscured lifeforms easier especially when they've fallen down. Cons: - An additional component will need to be included in the robot - The functionality this component provides is very limited for the cost and therefore will return very little benefit compared to the high costs. - Possibly intrusive way of monitoring. **Accelerometer** The accelerometer will measure how fast an object (e.g. the stakeholder wearing the sensors) moves. The accelerometer is useful to detect whether our main stakeholder starts accelerating towards the ground (see falling). Cons: - The sensor may not be able to detect the difference between regular movement and falling. If this is the case a lot of false alarms may occur. To make sure this scenario is avoided the sensor should be accurate enough to detect regular movement and calibrated to not set off any false alarms. **Microphone and speaker** The robot will then be able to communicate with the user through the use of the microphone and speaker. The microphone and speaker will be attached to the heart rate monitor and are different from the speaker & microphone in the robot. The microphone in the attached device will monitor the sounds the user makes so it can pick up on possible falling sounds and cries for help. Cons:

	- It may not be the case the user is able to speak into the microphone or be able to react to the sound from the speakers.
Argument	To check if a user has fallen down is difficult as it can be measured with many different sensors in different situations, however the QC robot should be able to detect falling anywhere in the house even if the resident falls out of audible range or sight. To accomplish this feat no single type of sensor would provide sufficient coverage in the scenario mentioned above. So instead a choice was made for the following components: **Microphone** A microphone can be used to detect if a user has fallen. If the robot detects the sound of something falling it can ask if the main stakeholder is alright. **A device attached to the user** To make accurate verdicts about whether the user has fallen down an additional device outside of the robot is required. This device will contain a number of components to measure the health of the resident. The device will be able to communicate with the robot via an internal network setup for communication between the two entities. The device will contain an accelerometer, heart rate monitor, a microphone and speaker and a button for emergencies. More information about the device will be specified in decision #3. **A camera** The robot will be able to use its camera to detect people that have fallen down. The camera is also used for various other functionalities the robot performs. In our evaluation these detection measures should be adequate to detect if a person has fallen down anywhere in the house as long as the person is within transmission range of the robot. If the user were to walk out of transmission range the device attached to the main stakeholder would be unable to perform its duties. A recurring important characteristic is reliability in monitoring. All the different sensors should not experience any downtime or the user might get hurt and the robot might not be aware of it until later. To mitigate this problem redundancy could be used to make the system more resilient. However, a more sensible tactic would be to use a watchdog process to check the status of the different sensors. The pinging will originate from another hardware component inside the robot. This component will monitor all the external sensors and verify their response. if one of of these sensors fails to respond to a status query it can be marked as defective and the user can be notified of the malfunction. This tactic increases the reliability of the service because faults can be prevented if a component is detected as

	defective before it is required to detect an emergency situation. **Quality attribute tradeoffs** The cost of this decision is not excessively high as all the selected internal sensors are already present in the robot, so the only additional cost would be additional software specific to fall detection. A part that might be a costly endeavor would be the wristband which the user should use. If this was developed from scratch this could be an expensive process as the wristband requires a number of measurement devices to be implemented with it. A way to drive these research and development (R & D) costs down would be to buy a component but it's not decided what course of action to take yet. (see #5)
Implications	The decision must be made to build the components in house or purchase an off the shelf solution.
Relation Decisions	Falling detection #3, #5
Related Requirements	-
Related Artifacts	-
Related Principles	-
Notes	Tactics: Watchdog tactic, redundancy
Quality Attribute Scenarios	Source: Human Stimulus: Main stakeholder has fallen down Artifact: A detection Environment: System operational Response: The robot will be notified the user has fallen down Response Measure: The robot will detect that user has fallen down within 30 seconds

Falling detection #2

Issue	What action will be performed in case the user falls down and needs more help
Decision	QC-Robot will send a signal to the QC Control Centre
Status	Decided
Group	Monitoring
Assumptions	• If user doesn't reply within 5 seconds then immediate action is necessary.

	- There is always (100%) an employee at QC Control Centre - Secure-line video is always (100%) available to QC Control Centre - The sequence of action QC-Robot makes when detecting that the user has fallen down is as follows. As it is described in Appendix IV, if the user presses the emergency button, a signal is sent immediately to QCCC. First action QC-Robot does when detecting that user has fallen down, is asking for a response and trying to detect the user. If no response has been given from the user within 5 seconds, then a signal is sent to QCCC. If a response has been received, then depending on the user's response, a signal might be sent or not. For a flowchart of the sequence see Appendix IV.
Constraints	WiFi connection is needed in order for QC-Robot to communicate with QC Control Centre (either way there will be a connection for the secure-line video)
Positions	**Send a signal to QC Control Centre** First thing the QC-Robot does when the user needs help is, let the QCCC (Quality Care Control Center) know. The employee at the control center can then decide whether a call to a relative or to 911 is needed. **Call a relative** First action of the QC-Robot after confirming that the user needs more help could be calling a relative, whose name and phone number would have been given when setting up the robot. Why not: Although this could be another decision, sending a signal to QCCC is preferable. The employees at the control centre, have optical view as well and so they can form a better opinion about the user's accident. Of course, if necessary, relatives will be informed by the QCCC. It is worth mentioning that it is not guaranteed that a relative will be available, in contradiction to the QCCC and 911. **Call emergency (911**) An alternative solution would be to immediately call 911 Why not: Again, although sometimes might have been the right decision, still QCCC can first form a better opinion about whether an emergency call is needed. Setting this call as first priority of QC-Robot might result in many unnecessary calls to 911

Argument	First of all, a major factor that we made this decision, was that it ‘contains’ the other positions, meaning that , as mentioned above, the QC-employee might call a relative or 911 if necessary, depending on the user’s status. Furthermore, QC Control Centre has access to the secure line video and so they can reasonably assess and evaluate the situation and take the appropriate actions. We also have to mention, that by making this decision the robot isn’t burdened with more functionalities, as it would be if there was a need to make a phone call.
Implications	The connection between the robot and the QC-Control center must be reliable.
Relation Decisions	Falling detection #4
Related Requirements	-
Related Artifacts	-
Related Principles	
Notes	
Quality Attribute Scenario	Source: User Stimulus: User has fallen down Artifact: A signal Environment: Fully operational Response: Robot will try to communicate with the main stakeholder for a response. Response Measure: if the QC robot does not get a response after asking the main stakeholder after 5 seconds, it will contact the QCCC.

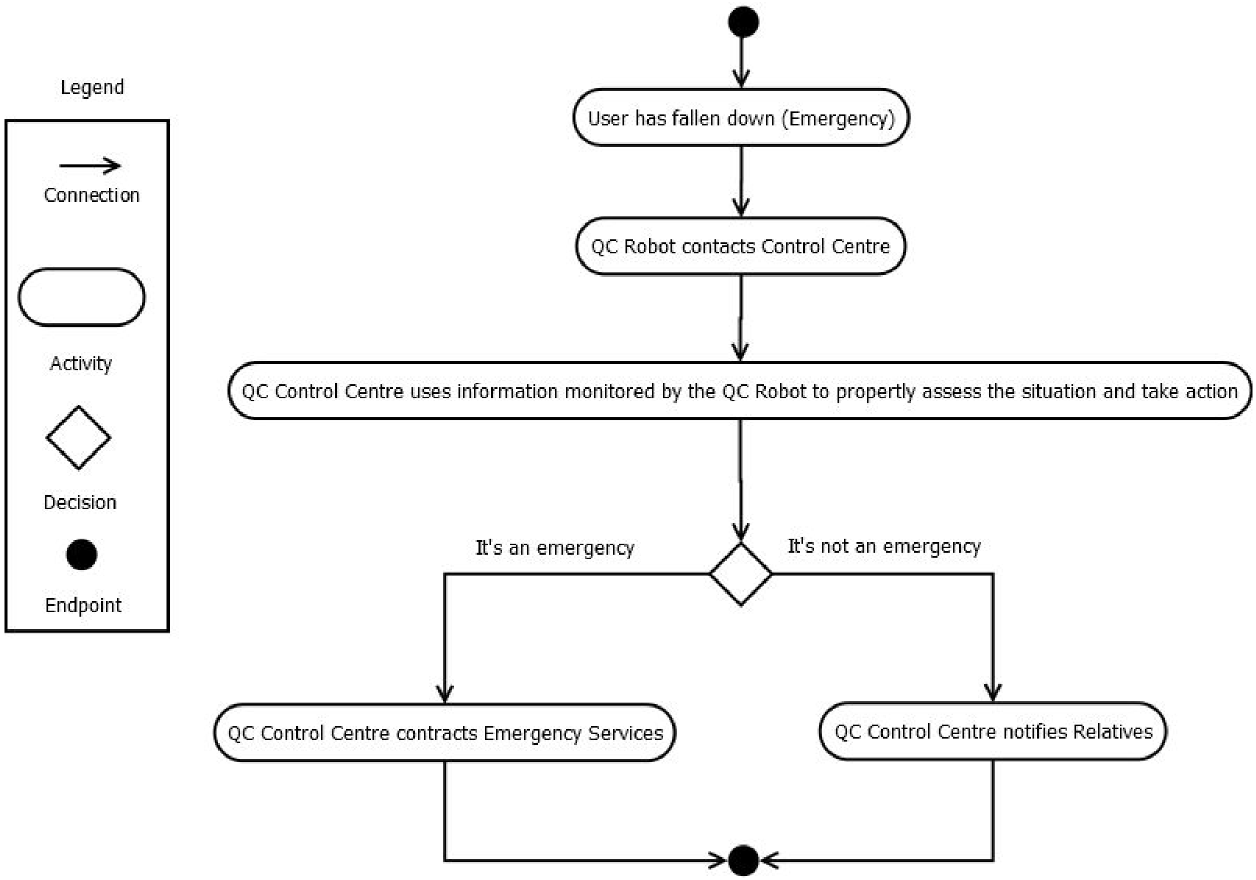

Diagram 11: Sequence of action QC-Robot makes when user has fallen down

[View image on github]

Falling detection #3

Issue	What kind of device should be attached to the user?
Decision	A wristband was chosen because it is the most accessible attachment for the main stakeholder.
Status	Pending
Group	Monitoring
Assumptions	• The robot is able to receive and send information to and from the device using WiFi. • It is assumed that there is only one button present on the

	wristband to eliminate the risk that the user presses the wrong button and ensures that if an emergency does arise the user will not be confused about which button to press. ● The latency of the information sent from and to the robot should not exceed 250 milliseconds. ● The device will have the following functionalities: ○ Heart rate measurement ○ Blood pressure measurement ○ Microphone ○ Speaker ○ Accelerometer ○ Emergency button ○ WiFi
Constraints	● The user has a wrist which the wristband can be attached to. ● The user must be able to communicate in order to use the microphone on the wristband.
Positions	**A wristband** The wristband is an easy to implement and accessible solution and won't require much effort in order to attach.The wristband can offer the whole selection of assumed functionalities without the need of any extra attached hardware to the main stakeholder. **A necklace** Why not: The necklace would not suffice to offer the full list of assumed functionalities, such as heart rate, which requires the necklace to be fully strapped around the main stakeholders neck, which would be very inconvenient **A chest-strap** Why not: The chest-strap is not a good solution because it does not offer a very convenient way to be attached to the user. The main stakeholder might not be able to attach the chest-strap themselves, thus need extra help in order to attach said solution. **An arm strap** Why not: although the arm strap would have been the next best option after the wristband, we decided to not go for this option, because it would be hard to use the full functionality such as the build in microphone when it is underneath clothing, thus requiring another device in order to achieve the communication aspect. **The waist** Why not: Although a waist band is an easy to reach spot, it could not measure heart rate properly, nor can it practically utilize the microphone, thus needing another external device attached to the user for full functionality.

	Headband Why not: the headband would not suffice to offer the full list of assumed functionalities, such as heart rate, which requires the headband to be fully strapped around the main stakeholders head, which would be very inconvenient.
Argument	We chose the wristband because it is the most accessible attachment. It satisfies the mentioned constraints . convenient, lightweight (watch)
Implications	The robot can setup a local network, which will be used in order to communicate with wristband and robot, thus eliminating the risk of external device outage (such as a router which the user does not have access to). The range of the WiFi must reach the entire surface of the household. The WiFi signal strength must be reliable over the entire range.
Relation Decisions	-
Related Requirements	-
Related Artifacts	-
Related Principles	-
Notes	**-**
Quality Attribute Scenario	Source: User Stimulus: The wristband being attached Artifact: Wristband Environment: Fully operational Response: the user presses the E.B button Response Measure: The button will trigger after 6 whole newtons of pressure is applied thus avoiding accidental presses.

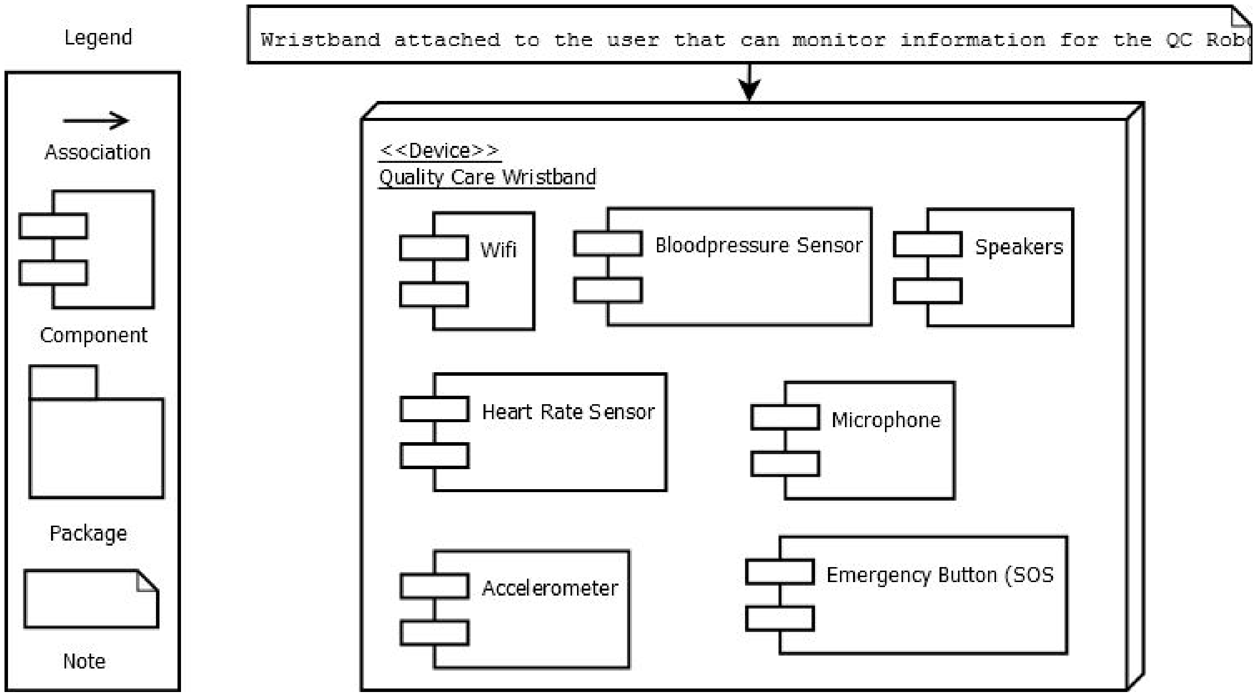

Diagram 12: Components for wristband

[View image on github]

Falling detection #4

Issue	Ensuring the that the QC control centre can be reliably (availability of at least 99.9%) notified during emergencies
Decision	The QC robot will have the ability to also connect via satellite to notify the QC centre of an emergency. Satellite connections are extremely reliable. (99.9% to 99.95%)
Status	Pending
Group	Monitoring
Assumptions	• The QC Robot itself can already contact the control centre using its internet connection
Constraints	The speed of the satellite connection is relatively slow and can only be used for notifying the QC centre
Positions	**Put emergency button in home** Fit the home with an emergency button which notifies QC and emergency services via the phone line. Phone lines are more

	reliable than internet connections over cable. The robot would be able to use this button in case that it cannot connect to centre using its internet connection Why not: It is possible that any issue with the home's internet connection may also have an effect on the phone line since these two services are often provided by the same company.
Argument	Satellite systems have a large amount of redundancy and are more reliable than cable and phone lines having less outages per year. See notes.
Implications	The robot will be required to be able to use an internet connection provided via satellite. A satellite's receiver will need to be distributed with the robot for placing in the main stakeholder's residence which will increase the costs.
Relation Decisions	Falling decision #2
Related Requirements	The robot must be able to assist the user in case they fall down
Related Artifacts	-
Related Principles	-
Notes	http://www.satelliteinternet-now.com/Satellite-Internet-Reliability.html http://www.vsat-systems.com/Education/Satellite-Internet-Explained/Performance/reliability/ The reliability can be measured as the amount of outages per year.
Quality Attribute Scenario	Source: User Stimulus:User is having a medical emergency Artifact: Connection with QC Control Centre Environment: Fully operational Response: Robot will communicate with the QCCC to determine next course of action Response Measure: The QCCC should be available at least 99.9% of the time to respond to incoming requests for communication by QC robots. Source: User Stimulus:User is having a medical emergency Artifact: Connection with QC Control Centre Environment: Partially operational Response: Robot will communicate with the QCCC to determine next course of action, however due to an outage of the phone grid this type of communication is not possible Response Measure: If a communication failure is detected a fallback communication method will be used after retrying the first method three times.

Falling detection #5

<table>
<tr><td>Issue</td><td>Build or purchase a solution for the wristband?</td></tr>
<tr><td>Decision</td><td>The building of the wristband will be outsourced to another company.</td></tr>
<tr><td>Status</td><td>Decided</td></tr>
<tr><td>Group</td><td>Monitoring</td></tr>
<tr><td>Assumptions</td><td>Software exists
List of components on the wristband:
Heart rate measurement
Blood pressure measurement
Microphone
Speaker
Accelerometer
Emergency button
WiFi</td></tr>
<tr><td>Constraints</td><td>Explicit requirements of the wristband should be given to the outsourcing company see chapter 8 for all the specifications.</td></tr>
<tr><td>Positions</td><td>Build
Building the wristband in house is an option. New personnel will be employed to build the product and QC would have greater control of the quality of the product.

Why not:
This options would require employing new specialists and acquiring the facilities to manufacture the device. These would result in a higher costs and make it favourable.

Pros
- More control of quality
Cons
- High cost
- Need specialist(s)
- Need to manufacture

Outsource
Outsourcing the product would lead to creating a contract with another company in which they will build the wristband for use with the QC robot. The external company would have the knowledge and experience to create the device and only require instructions on the features of the product. This has the downside that they require very clear and concrete specifications and QC will also need to perform</td></tr>
</table>

	additional tests to ensure the quality of the device. Pros - Lower cost compared to build - Don’t have to hire specialists - Don’t have to setup a manufacturing plant Cons - Outsourcing may result in unwanted side effects if the requirements are not clear or ambiguous resulting in the mass production of unusable products - Needs additional Quality assurance in order to see if the manufactured goods are up to the specifications - **Buy off the self** The components used for the wristband could possibly be bought off the shelves. Buying the wristband is much cheaper than creating it in-house which is not part of the core business of QC. There are two types of buying an off the shelves solution: either the solution is the whole package where the manufacturer supplies proprietary software which may or may not suit your needs. Or the second possibility is that hardware is provided with software that can be modified to add further functionalities. The second option is less desirable than the first but it’s harder to find a solution in the first category that describes the **exact** functionality which is called for as the solutions that fall in the first category cannot be modified. Why not: The buying of the component is simply not possible. There is a huge market of devices that have both a heart rate monitor and accelerometer but not so much with wi-fi and a microphone and speaker combination. Some preliminary research turned up one solution that was really close to the desired specifications (see notes) but unfortunately not modifiable so it wasn’t a viable alternative. Pros - Cheapest solution Cons - More quality assurance - Strict requirements on selected solutions
Argument	Outsourcing the product is the most reasonable option due to there being a lack of products on that can be purchased off the shelf and immediately comply with all requirements. This is also significantly cheaper than building the product in-house because it is not the core-business of QC, thus new personnel would need to be hired. The only downside is that QC has less control on the on quality of the product and all requirements for the system must be formalized in a contract with the outsourcing company.

Implications	The outsourcing company should be able to be contacted easily to help make sure the product that is manufactured is the product that was intended by Quality Care.
Relation Decisions	Falling detection 3
Related Requirements	-
Related Artifacts	-
Related Principles	-
Notes	http://www.ixonos.com/news-blogs/blog/design-blog/ixonos-wutuband-a-smart-wristband-for-the-elderly.-part-2
Quality Attribute Scenario	There are no measurable quality attributes in this decision as it is centered around costs.

8. Details of the wristband

In this chapter the specifications for the various hardware components of the wristband will be given. These specifications will be used when outsourcing or building the wristband for the robot.

8.1 Accelerometer

The Accelerometer will have an 3-axis (x,y,z) detection plane which will provide the right results when the fall is detected. Possible falls will be detected through an algorithm which uses a certain threshold which then can be used to distinguish normal movements, from the main stakeholder falling down. The main stakeholder's Quality Care robot will interpret a detected fall and will verify if the main stakeholder is in trouble or not, by means of following the method in : Appendix IV.

8.2 Microphone

The microphone will be attached to a wristband and it will allow the user to communicate with robot. It will also be able to notify the robot of the user falling due to sound picked up by microphone. Due to these reasons, it was decided that the microphone should have the following specifications:

- Omni-directional - Able to pick up sounds from every direction so that the user need not directly speak into the microphone
- 1 channel - It only needs to be able to record sound in one channel. There is no need for stereo quality sound as the microphone is not intended to be used for other purposes.

- It will need to have a sensitivity of about 85 dB due to being attached to the user's wrist and it should not pick-up too much background noise.

8.3 Speakers

The speakers will be integrated into the wristband and will be used for communication with the user. The speakers can be initiated if there is communication with the robot and the user via voice commands. This is a useful addition if the robot is further away from the user but the user is still in WiFi range in order to communicate. With the following decisions in mind the following specifications were selected for the speakers

- The speakers should be capable of producing noises within the range of 20 dB to 110 dB to accommodate users with impaired hearing.
- There is no requirement for more than one speaker so mono sound output is sufficient

8.4 Heart rate

The heart rate will be measured using the wristband which enables heart rate measurements. This will be done once every day between 11:00 - 14:00 and during emergency situations to check the main stakeholder's health.

- It will measure the heart rate as BPM
- The average deviation of the measurements should be less than 10 BPM to ensure its precision

8.5 Wi-Fi

The wristband will have built in WiFi support. All the measurement results from the sensors will be sent over WiFi to the robot. Due to being the only way of communication WiFi is a very important component of the wristband. The following specifications are specified for the wifi component:

- To make the wifi component as universally applicable as possible it will have support for two frequency bands. Both the 2.4 GHz and 5GHz bands will be supported by the wristband to ensure it is compatible with every type of wifi on the market today.
- The wifi component will have a maximum range of 150 meters to pick up and connect to wifi networks
- The wifi component will have support for the wireless variants AC, N and G

8.6 Emergency Button

The emergency button can be used by the user to notify that they need help immediately. This button will immediately notify the QC control centre and the local emergency services. This implies that it should not be possible to accidently press the button. The button will only respond to sufficiently level of force:

- The minimal force required for button activation is 6 Newtons.

- Button should be placed in a way that it can not be accidentally activated by bumping into it.

8.7 Blood pressure

In order for blood pressure to be measured, this functionality will be build into the wristband. Blood pressure will be used in order to check on the main stakeholder once a day at a given time (e.g. 11 am) and if an emergency occurs it will check on it once more in order to verify the main stakeholder's status. The blood pressure will be done using the following attributes:

- Flow measure
- Inflatable strap will be used in order to properly measure the blood pressure
- Adjustable wristband so the user can make the checking of blood pressure more accurate.

Appendix I: Reference Architectures

- JAUS reference architecture: uses a message passing protocol to provide interoperability among subsystems and components that compose systems resulting from this architecture
- ACROSET reference architecture: ACROSET is a component-oriented reference architecture for tele operated service robots and its main characteristic is the reuse of components from different systems. This architecture is composed by subsystems such as UIS (User Interaction Subsystem)
- Servicebots reference architecture: it was designed to service robots in indoor environments and is composed by 3 subsystems, service bots, fix bots and softbots. The type service bots is a robot capable of driving autonomously in an average complex environment only with sensorial information. The fix bots present sensor and actuators distributed all over the environment, having their own intelligence, whereas the softbots refers to the software agents executing various tasks for the requesting user
- Robot tele operation reference architecture: its main identified components are graphical representation, collision detection , user interface, communications and controller.

Reference architecture 1:

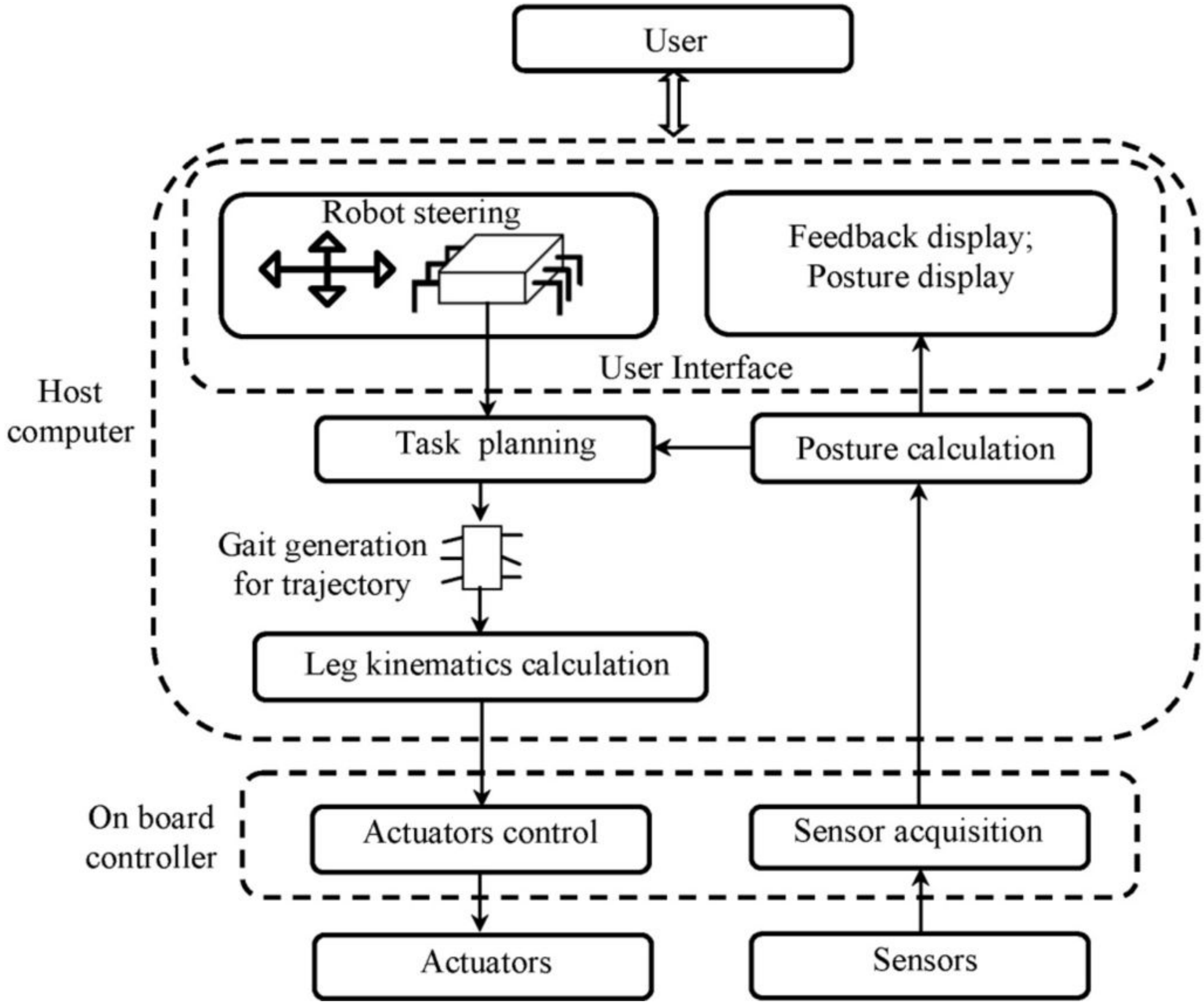

Diagram 13: Reference architecture #1

Source:http://www.mdpi.com/robotics/robotics-03-00181/article_deploy/html/images/robotics-03-00181-g007-1024.png

Reference architecture 2:

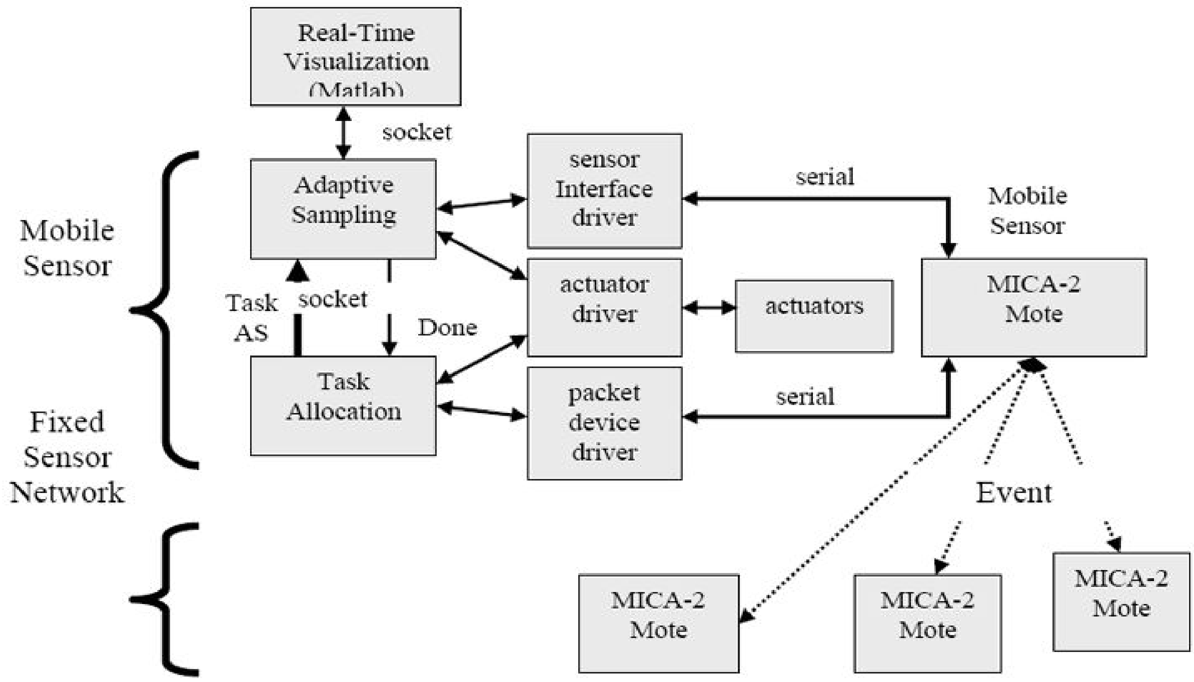

Diagram 14: Reference Architecture #2

Source: http://research.cens.ucla.edu/projects/2005/NIMS/software_arch/Figure1.gif

Reference architecture 3:

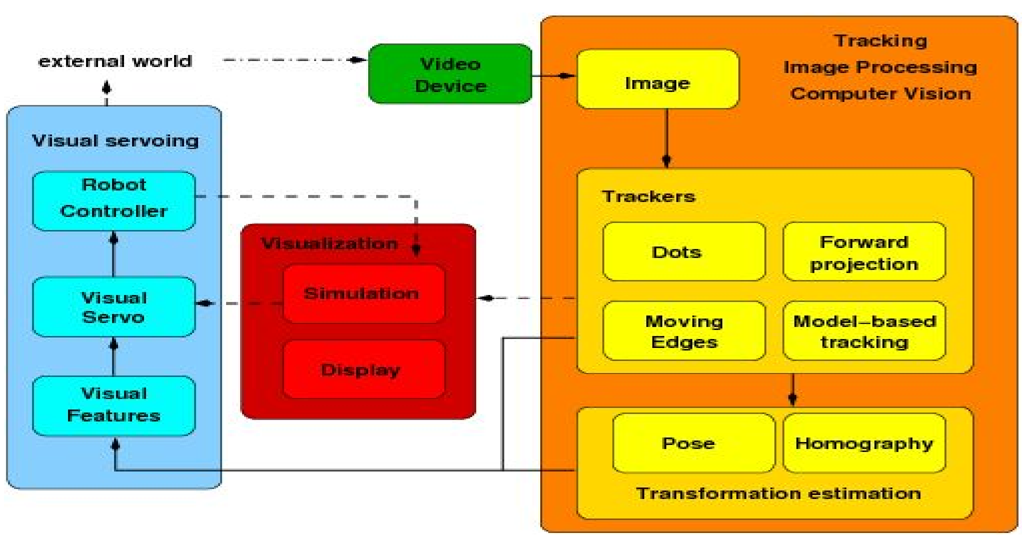

Diagram 15: Reference architecture #3

Source: http://raweb.inria.fr/rapportsactivite/RA2009/lagadic/1.png

Reference architecture 4:

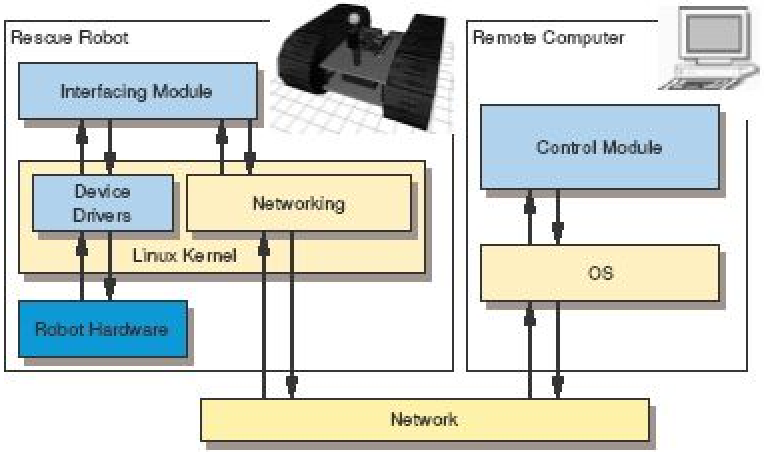

Diagram 16: Reference architecture #4

Source: http://www.eng.newcastle.edu.au/eecs/fyweb/Archives/2004/c2007000/images/softwareA.png

Reference architecture 5:

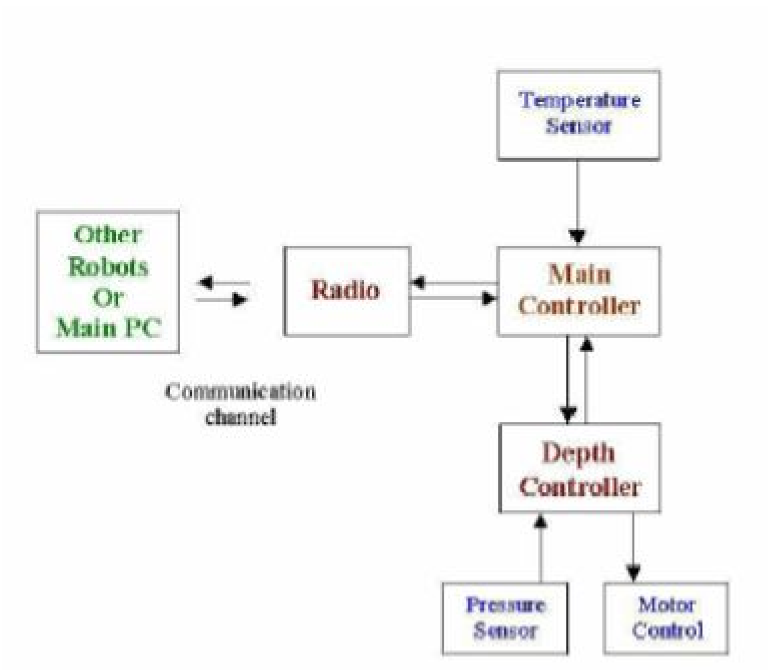

Figure 25: Software Architecture Schematic

Diagram 17: Reference architecture #5

Source: http://cens.ucla.edu/projects/2005/Actuation/testbeds/Figure25.GIF

Appendix II: Reflection implementation week 2

In week 4 our group was assigned to implement some examples using a pipe filter pattern. However the pipe filter implementation was not our own but one of our colleague's. We were supposed to use their implementation to create a number of example functions. We learned that working with someone else's codebase is quite a different experience than working on your own.

The implementation we were provided was reliant on the newest Java Development Kit (JDK) and the newest IDE versions. These technology choices were fine however none of our teammates had any experience with working with these new features in the JDK. Although this was no obstacle whatsoever it did give us some perspective on whether to use the newest technologies or staying with some older technologies of which knowledge may be assumed. It is important to specify the knowledge requirements in the documentation when creating components that will be reused by others. The last assignment made it clear that patterns shouldn't be forced to fit a design and they should be used when there is reason to.

To summarize:

- It is (sometimes) difficult work with the codebase of another team
- The choice of using established and new versions of technologies is complicated
- The knowledge required to use a component should (also?) be documented
- Patterns should not be forced to fit into any particular design. They are only solutions to certain problems and always require a clear rationale before they should be applied to a design.

Link to implementation:
https://github.com/felixbarten/software-architecture/tree/master/ft_7_code/dev/src/nl.uva.se.funcpipes

Appendix III: Links and references

[1] Jeff Tyree, Art Akerman, "Architecture Decisions: Demystifying Architecture", IEEE Software, vol. 22, no. 2, pp. 19-27, March/April, 2005

Appendix IV: Flow chart

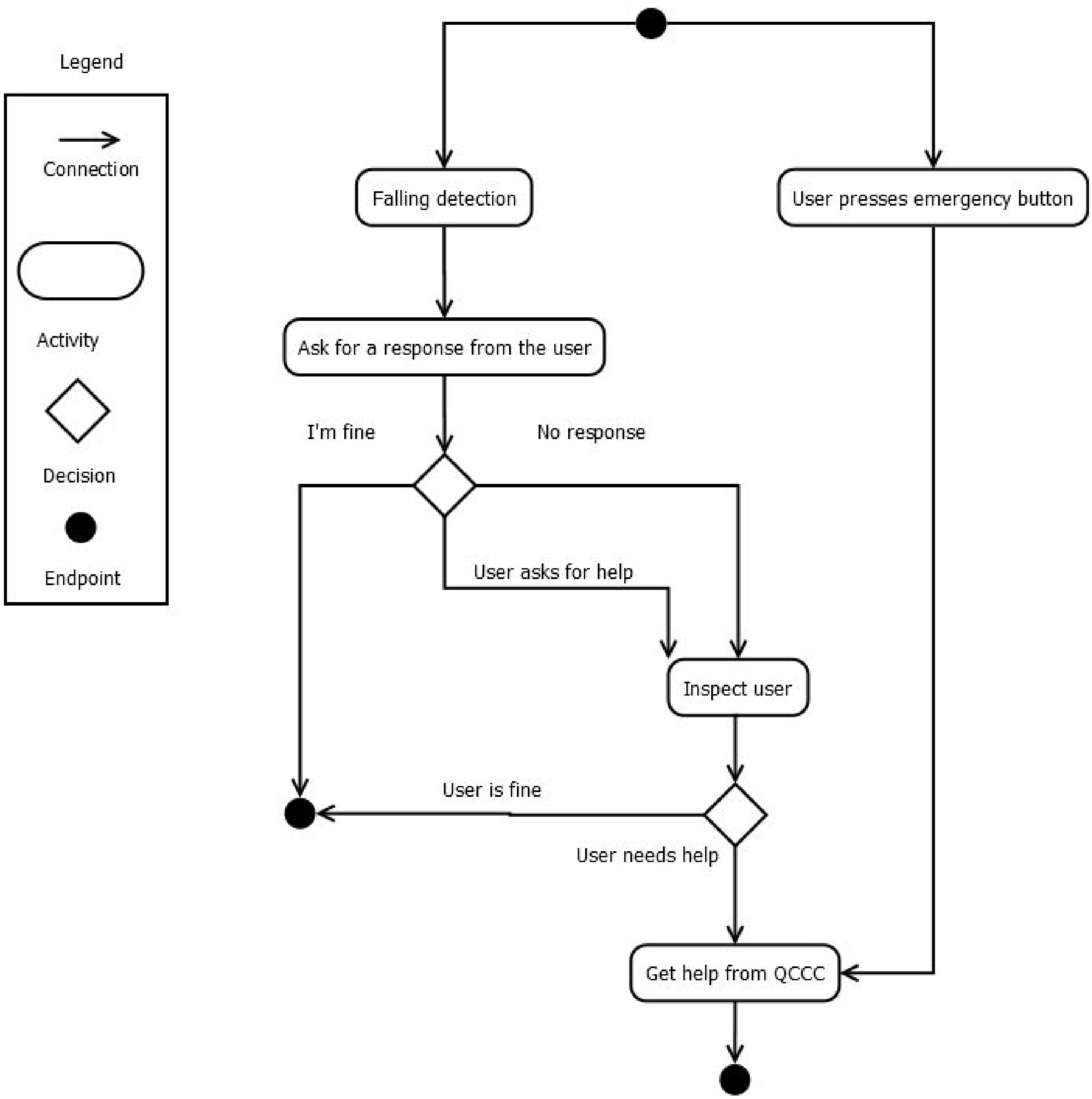

Diagram 18: Flowchart of falling detection

[View image on github]

Appendix V: Review goal

Review goal: Improving the quality of our storage decisions

Review questions:

1. Have we sufficiently covered the storage portion of our architecture?
2. Do you feel that have we dedicated enough time to the security of the medical information?
3. Is it consistent?