Project Plan

Smart Guard

Place, date: Enschede, 27th of February 2024

Drawn up by:

Ahmed Al-Ganad - 528820

Aryan Abbasgholitabaromran - 540346

Benjamin Iversen - 530791

Rafael Balan - 527065

Saif Ba Madhaf - 527028

Yusuf Isapaşa - 526352

Version: 0.1

Place, date:	Enschede, 12th of February 2023		
Prepared by:	Company Name		
	Ahmed Al-Ganad	528820	528820@student.saxion.nl
	Aryan Abbasgholitabaromran	540346	540346@student.saxion.nl
	Benjamin Iversen	530791	530791@student.saxion.nl
	Rafael Balan	527065	527065@student.saxion.nl
	Saif Ba Madhaf	527028	527028@student.saxion.nl
	Yusuf Isapaşa	526352	526352@student.saxion.nl
Version Number	0.1		
Project:	Project Integration		
Clients:	Hans Stokkink		
	Erik Karstens		
	Asif Khan		
	Umit Guler		

Abbreviation

ESP32: Espressif Systems 32-bit microcontroller

OLED: Organic Light Emitting Diode

GPS: Global Positioning System

IoT: Ineternet of things

UI: User Interface

PCB: Printed Circuit Board

ACS: Applied Computer Science

EIE: Electrical Information Engineering

Glossary

Table of Contents

Risk analysis

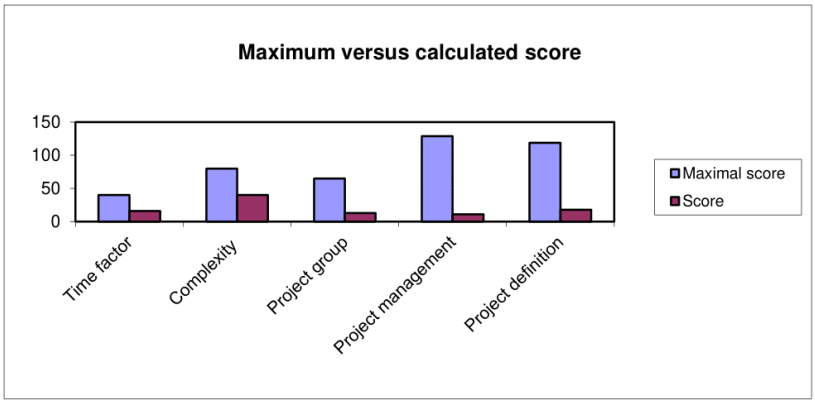
Risk analysis continued ….

	Risk	Value *	Factor **	Weight **	Total risk
Project definition		↓maak keuze↓			
24	Are project members sufficiently aware of problems and objectives?	yes, everybody	0	5	0
25	Is the field of result (scope) sufficiently defined?	yes	0	5	0
26	Is there enough distinction between this project and other projects?	reasonable	1	4	4
27	Has enough time been resereved for coordination and decision-making?	reasonable	1	4	4
28	Are the boundaries and preconditions clear?	yes	0	4	0
29	Are the boundaries limiting enough?	moderately	2	5	10
			Total		**98**
			Risk percentage ***		**22.63%**

** Value chosen by project manager*
*** Defined factor in this model.*
**** Risk percentage is total score divided by 433 (maximal score) times 100.*

The risk percentage provides an overview.
If the percentage is higher than 50%, the project should not carried out in this form.
The next table shows the risk percentage per catagory.

Category (maximum score versus calculated score)				
Time factor	Maximal	40	Score	16
Complexity	Maximal	80	Score	40
Project group	Maximal	65	Score	13
Project management	Maximal	129	Score	11
Project definition	Maximal	119	Score	18



1 Background

Smart Guard is a health care technical company that was founded by six students of computer scientists and electrical information engineers in 2024. The project takes place in Enschede, the Netherlands.

1.1 Departments Involved

There are multiple departments within SmartGraud:

- The research and development department which is the hub of searching for ideas, concepts, components, and programs.
- The software engineering department which will be the team that will code the embedded part of the healthcare devices, as well as the mobile application UI.
- The hardware engineering department is mainly responsible for the hardware design and functionality.
- The user experience and testing department which will make sure that the design is accessible and intuitive for elderly and individuals with chronic conditions.

1.2 Purpose of the project

Elderly people and individuals with chronic conditions such as heart diseases, diabetes, epilepsy, and more, face danger of fainting or passing out, and in some cases, this can cause death if the individual did not take action during and after fainting.

That is why Safeguard team decided to develop and design two concepts that help those people:

1- Developing an Automated Medication Reminder System that offers a reliable and user-friendly way for the elderly and those with chronic conditions, to manage their medication schedules with ease.
2- Designing a Personal Safety Device that serves as a portable safe guard, providing features such fall detection, panic button, and location tracking, to ensure the patient's safety 24/7.

1.3 Relationship With Previous Projects

With rich experience in multiple previous projects our team has experience in embedded systems, designing intuitive User Interfaces, databases architecture, and knowledge in IoT sensors technology. These skills will help design and implement a healthcare system that meets the needs of the users.

1.4 Stakeholders Involvement

Stakeholders play an important role in this project. Safeguard Internal stakeholders include the departments mentioned above that are responsible for developing and executing the project. External Stakeholders are responsible for guidance, feedback, and insight of the user's need.

2 Project Results

This project is taken into action to address two needs, enhancing medication compliance through an Automated Medication Reminder System, and increasing personal safety for those in need with a Personal Safety Device. These systems are important for improving the quality of life for elderly and individuals with chronic conditions, and for providing medical security for the users.

2.1 Objectives

The main objectives of this projects are:

- Developing an Automated Medication Reminder System that is user-friendly and reliable.
- Designing a Personal Safety Device that is compact, wearable, and equipped with essential safety features.

2.2 SMART Objectives

SMART objectives are chosen in this project to set well-described and precise objectives.

SMART objectives are a way to plan and set goals that are clear and reachable. Each letter stands for a word that describes the qualities of the goal which will be shown in the following bullet points.

- **Specific**: Creating two distinct products – A medication reminder and a safety device.
- **Measurable**: The system will be measured and evaluated in the testing phase where the devices will be ensured to fulfil and pass real-life scenarios.
- **Acceptable**: The system will be aimed towards elderly and people in need of high personal medical care.
- **Realistic**: Our time is experienced in embedded systems, IoT devices, intuitive UI designing, and hardware skills. Which makes this objective within our capabilities.
- **Time-phased:** The project is scheduled in phases with deadlines and a time of 2 quarters to fulfil the assignment.

2.3 Sub-objectives

Sub-objectives include:

- Implementing an application with UI design that is intuitive for the elderly. It should also send a notification through the app for the medication times.

- Ensuring reliable communication for the safety device.
- Integrating GPS and a fall detection technology for the safety device.
- Implementing a database to save the schedules inserted by the user through the mobile application.
- Designing a microphone to the safety device that saves the sound of the user and utilize it to remind him of the medication with his own sound.

2.4 User Requirements

The result of the project will be:

- A medication reminder system that allows users to schedule and receive notifications for their medications through a mobile app.
- A wearable safety device that offers a panic button, fall detection, and sends location to contacts or a central monitoring service powered by ESP32 microcontroller that offer wireless communications.

3 Project Activities

3.1 Methodological Approach (V-Model)

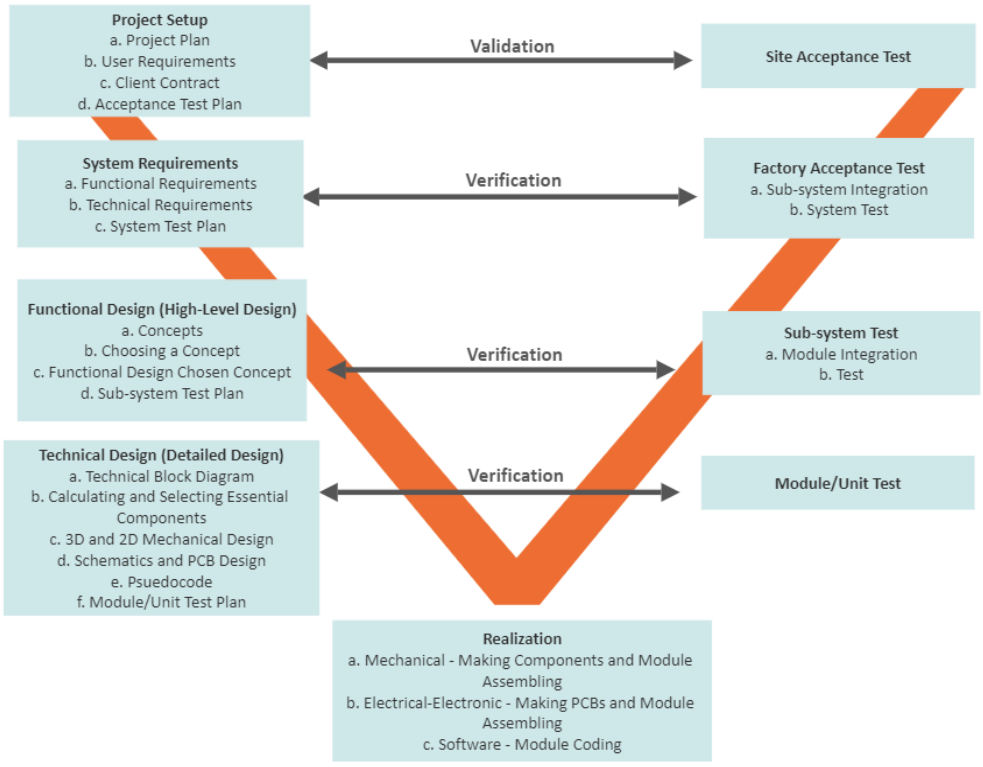


Figure 1 V-Model

The V-Model, also known as the Verification and Validation Model, involves executing operations in a V-shape design. It has important phases, tasks that must be done, and the outcomes that must be achieved during the products development. On the left side of the v-model is the systems requirements and on the right is the systems integration.

3.2 Activities

3.2.1 Project Setup

Main Activity:

This phase involves setting up project, establishing a methodological approach, making project objectives, creating a project timetable, and establish a communication system with the members. It also involves assigning roles and responsibilities for each team member.

Sub Activities:

- Developing a comprehensive project plan.
- Make project goals and objectives.
- Create a detailed timetable.

- Gather user requirements to know what the elderly need in a device.
- Assign a project leader and give roles and responsibilities for each member.

3.2.2 System Requirements

Main Activity:

This phase involves much research to come up with what the user needs to create a safety device for the elderly. Additionally, it involves making testing plans.

Sub Activities:

- Develop functional requirements.
- Develop technical requirements.
- Make testing plans.

Research Question:

What are the challenges faced by elderly users with current safety devices?

3.2.3 High-Level Design

Main Activity:

This stage involves dividing the system into smaller modules. In this context, the team will create block diagrams for both the software and hardware section explaining how the safety device for elderly will work briefly.

Sub Activities:

- Design functional concepts and then choose the final concept.
- Create block diagrams (Software and Hardware) to explain the overall system.

Research Question:

What specific functionalities must the device have to address the safety concerns of the elderly, such as fall detection, emergency alerts, and location tracking?

3.2.4 Detailed Design

Main Activity:

This stage involves dividing the system into smaller detailed modules. Unlike the functional design the technical design is more specified. The team will design the PCB and draw a detailed software flowchart outlining the device functionalities.

Sub Activities:

- Make a technical block diagram.
- Design schematic and PCB
- Draw software flowchart.

Research Question:

How can the team ensure that the PCB design is as small as possible to be a portable device?

3.2.5 Realization

Main Activity:

This phase involves implementation of the project plan. The team will begin building the hardware and writing the software.

Sub Activities:

- Assembling the hardware components.
- Writing the code for the components.

3.2.6 Unit Testing

Main Activity:

In this step the team begins with testing components individually. Testing involves checking if the hardware components function correctly and testing the software on the component itself. With these testing's the team can then identify bugs and issues.

Sub Activities:

- Testing hardware components
- Test software code on each component.
- Fix bugs and issues found.

3.2.7 System Testing

Main Activity:

In this phase of the project, the system is fully built, and the team begins by testing the system as whole to ensure that the components work together perfectly, and the code functions as intended.

Sub Activities:

- Perform testing on the system.
- Check the requirements made in the system requirements to see if the final system meets the requirements.
- Fix bugs and issues encountered.

3.2.8 Factory Acceptance Testing

Main Activity:

In this stage the team reviews the final product to see if it meets the customer requirements.

Sub Activities:

- Review the final product if it meets the customer requirements.

3.2.9 Site Acceptance Testing

Main Activity:

In the final stage, the team deploys the device to the consumer for testing.

Sub Activities:

- Give the safety device the consumer.
- Gather feedback from the consumer.
- Adjust the final product according to the feedback received.

4 Project Boundaries

4.1 Timetable

Week number	Task	Deliverable	Complete by
3.1 12-02-2024 - 16-02-2024	Introduction Research and develop project plan		
3.2 26-02-2024 – 01-03-2024	Research and writing Project Plan	**Project Plan**	01-03-2024
		Proposal circuit design EIE	01-03-2024
3.3 04-03-2024 – 08-03-2024	Introduction Design Requirements	Research Assignments ACS, EIE	08-03-2024
3.4 11-03-2024 – 15-03-2024		Design Requirements report	15-03-2024
3.5 18-03-2024 – 22-03-2024	Introduction functional design Prepare for demonstration and finishing reports		
3.6 25-03-2024 – 29-03-2024		**Demonstration** Functional design PCB design Research report (ACS)	**25-03-2024** 29-03-2024 29-03-2024 29-03-2024
3.7 01-04-2024 – 05-04-2024	Introduction Technical design	Assessment: Introduction research	01-04-2024
3.8 08-04-2024 – 12-04-2024	Technical Design		
3.9 15-04-2024 – 19-04-2024		Technical design	19-04-2024
3.10 22-04-2024 – 26-04-2024	Assembling, unit testing	Retake Assesment Introduction Research	

4.1 29-04-2024 – 03-05-2024	Holiday		
4.2 06-05-2024 – 10-05-2024	Implement circuit EIE Develop final code ACS		
4.3 13-05-2024 – 17-05-2024	Integrate software with circuit ACS, EIE		
4.5 20-05-2024 – 24-05-2024	Testing Working on final report		
4.6 27-05-2024 – 31-05-2024		**Final Report**	31-05-2024
4.7 03-06-2024 – 07-06-2024		**Demonstration Assesment**	03-06-2024
4.8			
4.9 17-06-2024 – 21-06-2024		**Retake**	

4.2 MoSCoW

The MoSCoW method is used to categorize the prioritization of key requirements and features.

- **Must-have category** the most critical requirements that are essential.
- **Should-have category** requirements that are important but not essential.
- **Could-have category** requirements that are desirable but not necessary.
- **Won't-have category** requirements that are explicitly excluded.

4.3 Requirements

Using MoSCoW we can list the safety device and the scheduling app requirements for this project as follows:

must-have:

- Reliable safety device controlled by ESP32.
- Wearable device.
- Emergency alert system for in the safety device to notify contacts or services.
- Fall detection method to increase the safety of individuals in need.
- Panic button that sends location contacts or a central monitoring service.
- GPS tracking for location monitoring in the safety device.
- User-friendly mobile app interface for setting up medication schedules.
- Notifications for the medication reminder sent to the user's phone.

should-have:

- Integration of the medication reminder system with email/SMS for additional notification options.
- Data encryption for user information for privacy.
- Alert tones or messages for the wearable safety device to alert for the medication.
- Water-resistance properties for the wearable safety device.

could-have:

- Voice commands for managing the medication app.
- Multi-language support in the mobile app and the wearable safety device.

won't-have:

- Indoor navigation support for GPS tracking.
- Advances AI diagnostics for the patient's health conditions.
- Self-maintenance capabilities for the wearable safety device that detects technical issues.
- Solar charging or extended battery life features for the wearable safety device.

5 Intermediate Results

Intermediate results provide checkpoints throughout the making of the project, allowing teams to monitor their progress and ensure they are on track to meet their goals and deadlines.

The company is looking forward to completing the following:

- Circuit designs and PCB layouts for the hardware components.
- Research reports on the ESP32 board.
- Hardware prototype construction and testing.
- Integration of hardware and software components.
- Testing sessions to evaluate system effectiveness and user experience.
- Testing battery life and power consumption.
- Testing for safety standards regarding wearable technology.
- Refining of the product based on prototype testing and user feedback.

- The finalized project plans.
- The functional design report.
- User manuals and guides.
- System requirements report.
- User requirements and site acceptance test reports.

6 Quality

Ensuring the quality of the product is a crucial aspect on this project, because of that V-model methodology which is described in Chapter 3 is used to ensure quality of the healthcare product by going through phases in order to achieve the best quality.

6.1 User Interface Design

For the Automated Medication Reminder System, the user interface (UI) design is critical in ensuring that users can easily navigate and manage their medication schedules. Given the target user base, which includes the elderly and individuals with chronic conditions, the UI needs to be intuitive and accessible.

- **Simplicity**: The app will feature large, readable text and simple navigation menus to accommodate users with varying levels of tech-savviness and possible visual impairments.
- **Personalization**: Users will have the ability to personalize settings, such as medication times, dosages, and notification preferences, making the app adaptable to their specific needs.
- **Interaction with the ESP32 Device**: The UI will seamlessly integrate with the ESP32-controlled medication dispenser, providing real-time feedback on medication dispensing activities and allowing users to interact directly with the device through the app, enhancing the user experience.

6.2 Reliability and Accuracy

The effectiveness of the Automated Medication Reminder System hinges on its reliability and accuracy in dispensing medication and sending reminders. This is crucial for building user trust and ensuring adherence to medication schedules.

- **Testing Protocols**: Implement rigorous testing protocols, including unit testing for individual components, integration testing to ensure seamless system operation, and user acceptance testing with real-world scenarios to validate the system's functionality.
- **Error Handling**: Develop robust error-handling mechanisms to deal with potential issues such as dispenser jams, low medication supplies, or communication errors between the app and the dispenser, ensuring the system remains operational.
- **Feedback Loops**: Incorporate feedback loops in the app, allowing users to confirm the receipt of medication and to manually log doses if needed. This dual-check system enhances accuracy and reliability.

6.3 Compliance and Standards

Adhering to healthcare and data protection regulations is paramount for the Automated Medication Reminder System to ensure user data security and system compliance.

- **Healthcare Regulations**: Follow healthcare industry standards and regulations, such as HIPAA (Health Insurance Portability and Accountability Act), to protect patient health information.
- **Data Protection**: Implement state-of-the-art encryption and secure data storage solutions to safeguard user data, including medication schedules and personal information, both in transit and at rest.
- **Regular Audits and Updates**: Conduct regular security audits and system updates to address new vulnerabilities and ensure compliance with the latest data protection laws and healthcare standards.

7 Project Organisation

7.1 Organisation

For the project leader we have selected Ahmed, the group has yet to be divided into smaller more specialised groups although when it does it will likely be split into a few different teams that change members during the design process, for example: the hardware design team may need an extra person to do 3d print work and that person will be moved from another team temporarily.

Members	**Email Addresses**	**Phone Numbers**
Ahmed Al-Ganad	528820@student.saxion.nl	+31 686456952
Aryan Abbasgholitabaromran	540346@student.saxion.nl	+31 627657317
Benjamin Iversen	530791@student.saxion.nl	+230 57340220
Rafael Balan	527065@student.saxion.nl	+31 641191461
Saif Ba Madhaf	527028@student.saxion.nl	+31 684121320
Yusuf Isapaşa	526352@student.saxion.nl	+90 5379281583

All responsibilities and tasks will be recorded through a dedicated discord channel all group members will have access to.

7.2 Information

Information within the team will be communicated through discord as well as though WhatsApp, communication with the project managers will be handled through email and meetings whenever necessary.

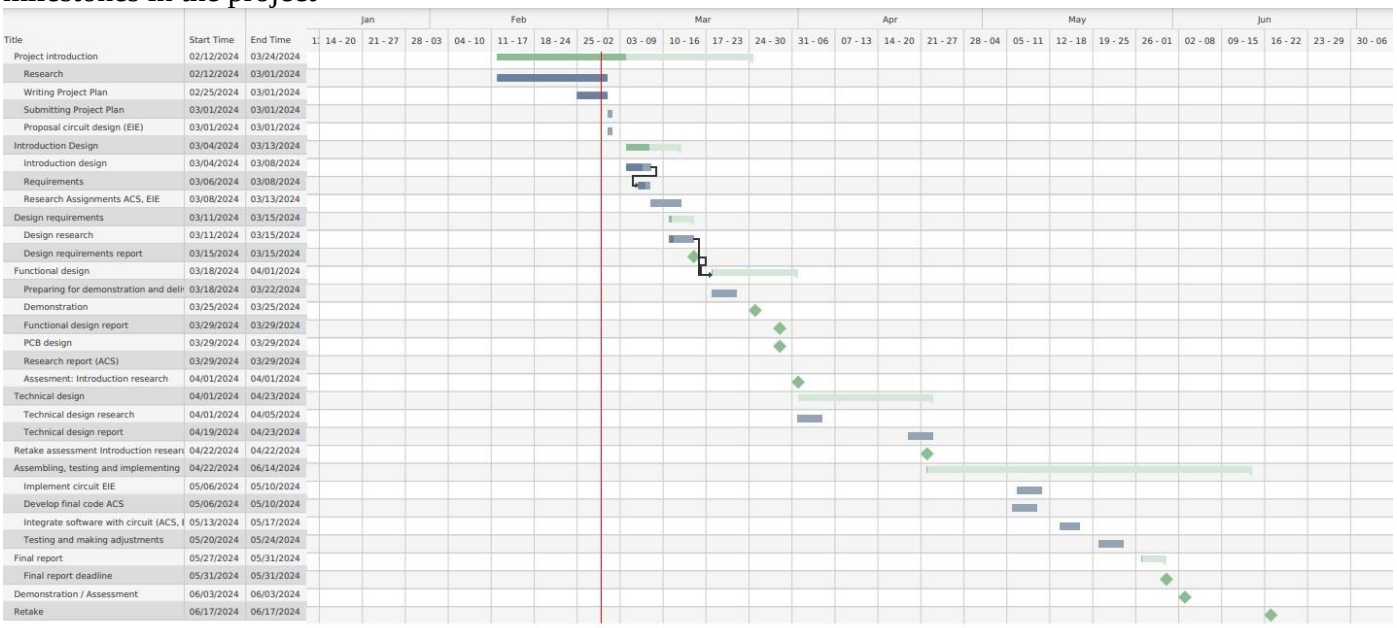
The meetings with the project managers will ideally be scheduled once every few weeks to make sure the project scope doesn't drift too far off course.

Meetings within the group will happen once in person to delegate tasks and discuss ideas then a few online meetings per week to discuss progress and potential issues.

Documents will be stored in a shared OneDrive folder.

8 Planning

In the planning section we will look at our gantt chart which is highlighting the most important milestones in the project



Title	Start Time	End Time
Project introduction	02/12/2024	03/24/2024
Research	02/12/2024	03/01/2024
Writing Project Plan	02/25/2024	03/01/2024
Submitting Project Plan	03/01/2024	03/01/2024
Proposal circuit design (EIE)	03/01/2024	03/01/2024
Introduction Design	03/04/2024	03/13/2024
Introduction design	03/04/2024	03/08/2024
Requirements	03/06/2024	03/08/2024
Research Assignments ACS, EIE	03/08/2024	03/13/2024
Design requirements	03/11/2024	03/15/2024
Design research	03/11/2024	03/15/2024
Design requirements report	03/15/2024	03/15/2024
Functional design	03/18/2024	04/01/2024
Preparing for demonstration and deli	03/18/2024	03/22/2024
Demonstration	03/25/2024	03/25/2024
Functional design report	03/29/2024	03/29/2024
PCB design	03/29/2024	03/29/2024
Research report (ACS)	03/29/2024	03/29/2024
Assesment: Introduction research	04/01/2024	04/01/2024
Technical design	04/01/2024	04/23/2024
Technical design research	04/01/2024	04/05/2024
Technical design report	04/19/2024	04/23/2024
Retake assessment Introduction researc	04/22/2024	04/22/2024
Assembling, testing and implementing	04/22/2024	06/14/2024
Implement circuit EIE	05/06/2024	05/10/2024
Develop final code ACS	05/06/2024	05/10/2024
Integrate software with circuit (ACS, I	05/13/2024	05/17/2024
Testing and making adjustments	05/20/2024	05/24/2024
Final report	05/27/2024	05/31/2024
Final report deadline	05/31/2024	05/31/2024
Demonstration / Assessment	06/03/2024	06/03/2024
Retake	06/17/2024	06/17/2024

9 Costs and Benefits

9.1 Cost

Hardware Components:

- **ESP32 Module**: The core of the system, providing WiFi and Bluetooth connectivity for app communication and controlling the medication dispenser. The cost per unit varies based on the volume of the order and specific model features.
- **OLED Display**: For user interaction directly on the device, displaying medication schedules, reminders, and system statuses. Bulk purchasing can reduce costs.
- **Medication Dispensing Mechanism**: Includes motors, sensors, and storage compartments. The complexity of the design and the materials used will significantly influence the cost.

9.2 Benefits

Improved Medication Adherence

- The system directly addresses the challenge of medication non-adherence, significantly improving health outcomes for users, especially the elderly and those with chronic conditions. This reduces the risk of complications and hospital readmissions.

Ease of Use and Accessibility

- With a user-friendly interface and automated dispensing, the system simplifies managing medication schedules. This is particularly beneficial for users with cognitive impairments or those who manage multiple medications daily.

Enhanced Safety and Peace of Mind

- Real-time notifications and reminders ensure that users do not miss their medication doses, enhancing safety. For family members and caregivers, the system provides peace of mind knowing that medication management is being actively supported.

Data Insights for Healthcare Providers

- With user consent, the system can provide valuable data insights to healthcare providers, enabling personalized care plans and monitoring the effectiveness of treatments.

Cost Savings in Healthcare

- By improving medication adherence, the system can lead to significant cost savings in the healthcare system, reducing emergency room visits, hospitalizations, and the use of expensive medical treatments.

10 Risks

Internal or External risk:	Types of Risk:	Potential problem:	Potential solution:
Internal	Organization and time-related problems	Project plan isn't completed thoroughly Objectives, goals, roles are not defined or unattainable. Time wasted. Lack of communication Teammate leaving the team.	Ensure the team is working together Weekly meetings Making a timetable, to distribute the tasks evenly. Team members actively communicating.

		Team members unwilling to do the work or not motivated. Different work style and methodology to do work.	
Internal and External	Operational Risks	Users might struggle adapting to the new systems, making the product less desirable. Data breaches may occur, compromising the data of the user. Disruptions in the supply chains may lead to delays in meeting the deadlines.	Involve users in the testing procedures and listen to feedback. Provide learning resources. Implement encryption between systems and authentication measures. Test for vulnerabilities. Diversify suppliers, and order spare parts.
Internal	Unrealistic Goal	Focus too much on want instead of focusing on must and should. The goals should be SMART.	timetable, complete the project plan and execute the project plan properly.
Internal	Manufacturing problems	Hardware components might break or fail, causing delays and time wasted. Integrating hardware and software components may pose challenges.	Follow the V-Model and make sure each step is met. For example, unit testing can help find any potential broken part. Allocate sufficient time for troubleshooting and debugging.
Internal	Software related problems	Bad programming practice.	Follow project plan, functional design,

		Lack of comments/documentation. Overly complicated code Inefficient code. Spaghetti code. Compatibility issues.	technical design plan, integration plan, Do it one step at a time
Internal	Poor time management	Not properly following the timetable.	Request help when the problem appears
External and Internal	Safety related issues	Components catching on fire, or electrocuting personnel.	Make the environment we are working on be built ready for the case of emergency. Be well prepared in case of the worst, such as having a fire extinguisher nearby.
External and Internal	Financial risks	Going over the allocated budget, due to unforeseen expenses. Sponsor risk. Such as when the sponsor decided not to give any financial support.	Conduct cost estimations. Management should also keep track of finances. Monitor expenditure throughout the whole time.
Internal	Liability risk	Lawsuit Causing damages that lead up to lawsuits, such as electrocuting the person wearing the smart watch.	Make sure the team's work is original and create trademarks for the product to ensure protection.

		Environmental damage. Break of contract. Copyright infringement / intellectual property breach.	Keeping the product confidential. Make sure the team is properly executing the plan so there is no added cost.
Internal	Quality	The quality of components, or software process is bad Teammates putting in low effort may lead to low quality products.	Make sure order lists are done properly and well prepared. Make sure the quality of the components is good before assembly. Various tests should be conducted to ensure quality Proper management and communication to make sure teammates are doing their best. Creating rules and guidelines.

10.2 Risks Analysis conclusion

Risk analysis is an important process in software and hardware development that involves finding, assessing, and managing potential risks to ensure the success of the project. A thorough risk analysis can help the team to efficiently address issues before they become major problems and can ultimately improve time management, efficiency, and success odds.

Appendix A: Risk Analysis

Risk analysis

Risk Analysis	Print date
SmartGuard	2/27/2024

	Risk	Value *	Factor **	Weight ***	Total risk
Time factor		↓choose one↓			
1	Estimated duration of project	3 - 6 months	1	4	4
2	Does the project have a definite deadline?	Yes	2	4	8
3	Is there enough time to complete the project within the time permitted?	enough	1	4	4
Complexity of project		↓choose one↓			
4	Number of functional subsectors involved	3+	3	4	12
5	Number of functional subsectors that will make use of the results of the project	4	2	2	4
6	Is is a new project or one that will be adapted?	New project	3	5	15
7	To what extend do current authorizations in the organization have to be adjusted?	Not	0	5	0
8	Are other projects dependent on this project?	No	0	5	0
9	How are users (of the project results) likely to respond to it?	Enthousiastic	0	5	0
10	Is the project broken down into phases and does the project depend on coordination between them?	Strongly	3	3	9
The project group					
11	Where do the project workers come from?	Mainly internally	0	4	0
12	Where is the project located?	1 location	0	2	0
13	How many project members work for more than 80% at peak hours?	5-10	2	5	10
14	Balance between subject experts and project experts	Good	0	5	0
15	Are users involved in the project?	to a reasonable extend	1	3	3
Project management		↓choose one↓			
16	Does the project management team have any knowledge of the subject?	a lot	0	3	0
17	Does the project management have any knowledge of how to plan a project?	a lot	0	3	0
18	How much experience does the project manager have with a project like this?	a reasonable amount	1	3	3
19	Does the adviser have much knowledge of the field of the project?	a lot	0	5	0
20	Do the subject experts have much knowledge of the field?	a lot	0	5	0
21	How involved are responsible managers in the project?	very	0	5	0
22	Is there any chance that the project team will change during the project?	little chance	0	5	0
23	Is the project team using existing methods or are they creating their own tools?	some existing methods	2	4	8

Risk analysis

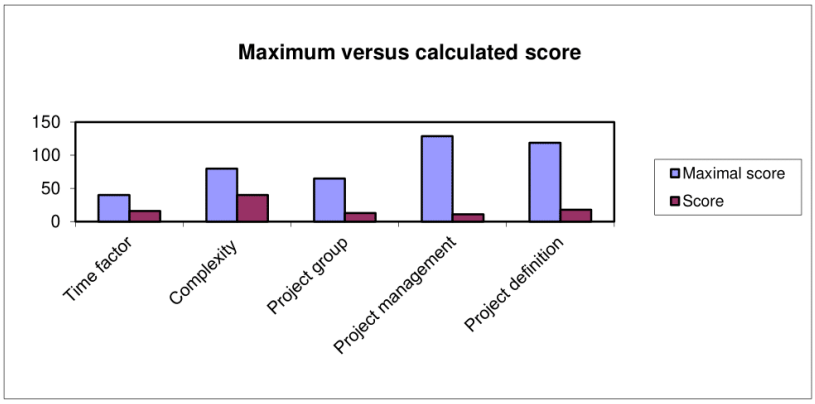
Risk analysis continued ….

	Risk	Value *	Factor **	Weight **	Total risk
Project definition		↓maak keuze↓			
24	Are project members sufficiently aware of problems and objectives?	yes, everybody	0	5	0
25	Is the field of result (scope) sufficiently defined?	yes	0	5	0
26	Is there enough distinction between this project and other projects?	reasonable	1	4	4
27	Has enough time been resereved for coordination and decision-making?	reasonable	1	4	4
28	Are the boundaries and preconditions clear?	yes	0	4	0
29	Are the boundaries limiting enough?	moderately	2	5	10
			Total		**98**
			Risk percentage ***		**22.63%**

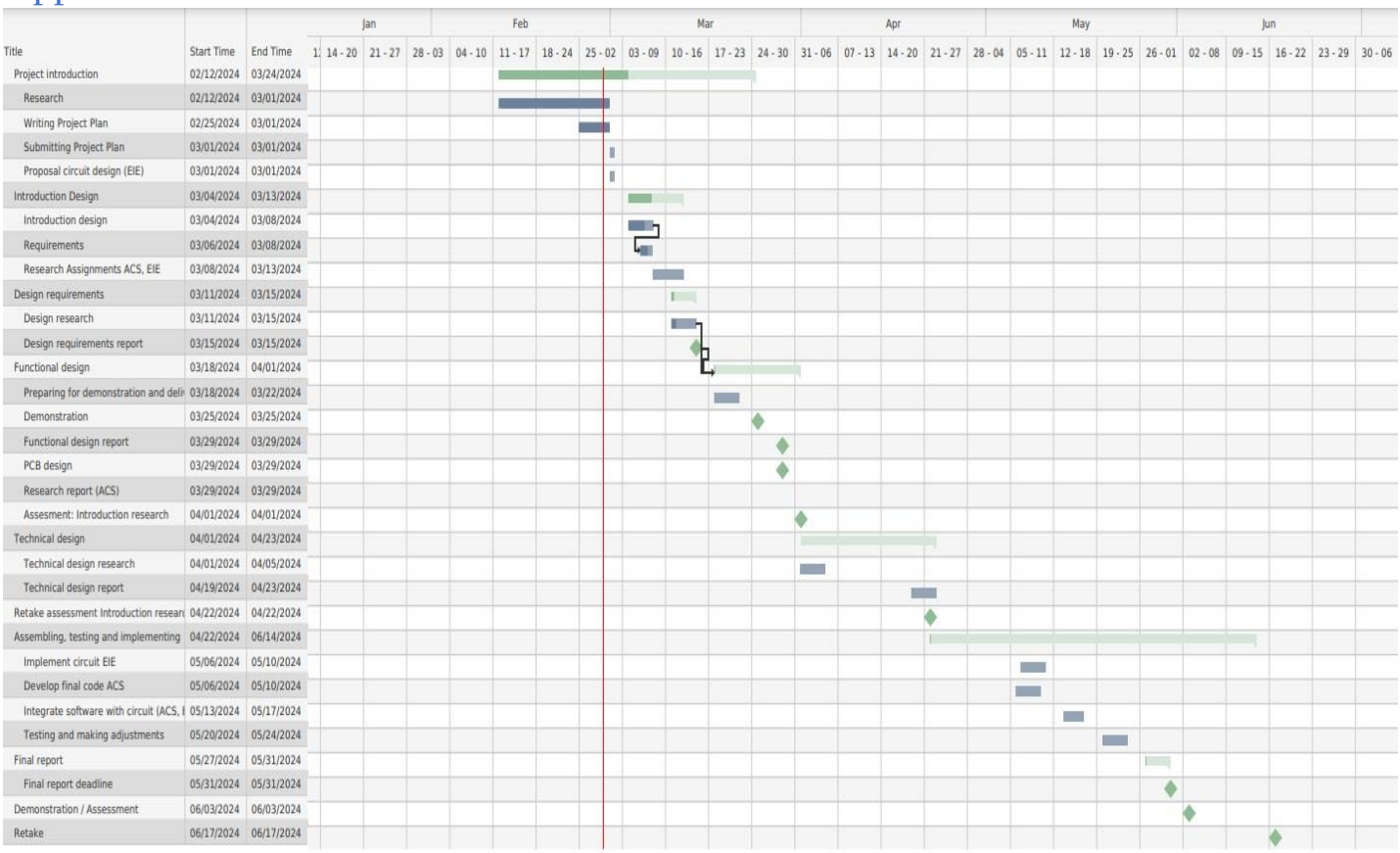
** Value chosen by project manager*
*** Defined factor in this model.*
**** Risk percentage is total score divided by 433 (maximal score) times 100.*

The risk percentage provides an overview.
If the percentage is higher than 50%, the project should not carried out in this form.
The next table shows the risk percentage per catagory.

Category (maximum score versus calculated score)				
Time factor	Maximal	40	Score	16
Complexity	Maximal	80	Score	40
Project group	Maximal	65	Score	13
Project management	Maximal	129	Score	11
Project definition	Maximal	119	Score	18



Appendix B: Gant Chart



Title	Start Time	End Time
Project introduction	02/12/2024	03/24/2024
Research	02/12/2024	03/01/2024
Writing Project Plan	02/25/2024	03/01/2024
Submitting Project Plan	03/01/2024	03/01/2024
Proposal circuit design (EIE)	03/01/2024	03/01/2024
Introduction Design	03/04/2024	03/13/2024
Introduction design	03/04/2024	03/08/2024
Requirements	03/06/2024	03/08/2024
Research Assignments ACS, EIE	03/08/2024	03/13/2024
Design requirements	03/11/2024	03/15/2024
Design research	03/11/2024	03/15/2024
Design requirements report	03/15/2024	03/15/2024
Functional design	03/18/2024	04/01/2024
Preparing for demonstration and deli	03/18/2024	03/22/2024
Demonstration	03/25/2024	03/25/2024
Functional design report	03/29/2024	03/29/2024
PCB design	03/29/2024	03/29/2024
Research report (ACS)	03/29/2024	03/29/2024
Assesment: Introduction research	04/01/2024	04/01/2024
Technical design	04/01/2024	04/23/2024
Technical design research	04/01/2024	04/05/2024
Technical design report	04/19/2024	04/23/2024
Retake assessment Introduction researc	04/22/2024	04/22/2024
Assembling, testing and implementing	04/22/2024	06/14/2024
Implement circuit EIE	05/06/2024	05/10/2024
Develop final code ACS	05/06/2024	05/10/2024
Integrate software with circuit (ACS, I	05/13/2024	05/17/2024
Testing and making adjustments	05/20/2024	05/24/2024
Final report	05/27/2024	05/31/2024
Final report deadline	05/31/2024	05/31/2024
Demonstration / Assessment	06/03/2024	06/03/2024
Retake	06/17/2024	06/17/2024

References