

**Izmir University of Economics**  
**Department of Software Engineering**

**SE 216 Final Document**

**Hüseyin Atacan Akgün – 20220601001**

**Erkan Efe Yeşilçimen – 20220601074**

**Bartu Erdem – 20220601025**

**Samet Tolga Esen – 20210601024**

**Emin Buğra Aksoy – 20210601004**

**Batuhan Can – 20210601051**

## Outline

1. Project Proposal .....	2
1.1. Problem Definition .....	2
1.2. Objectives .....	3
2. High Level Functionality .....	3
2.1. Requirements.....	3
2.1.1. Functional Requirements .....	3
2.1.2. Non-Functional Requirements .....	4
3. Stakeholders .....	5
3.1. Internal Stakeholders .....	5
3.2. External Stakeholders .....	5
4. Software Process.....	6
5. Software Risks.....	7
5.1. Likelihood Risk List.....	7
5.2. Impact Risk List.....	8
5.3. Combined Risk List.....	9
6. Project Needs.....	10
6.1. Software Needs .....	10
6.2. Hardware Needs .....	11
6.3. Support Needs .....	11
7. Software Measurements .....	12
8. Software Tools .....	14
8.1. Database Management.....	14
8.2. Analytics and Reporting .....	15
8.3. Live Support .....	16
9. Project Schedule.....	17
10. Project Payoffs.....	18
11. User Interfaces .....	18
11.1. Mobile App Interface.....	18
11.2. Desktop App Interface .....	19
12. Conclusion.....	19

# 1. Project Proposal

## 1.1. Problem Definition

Some of the biggest problems today are misuse of medicines, waste of paper and uninformed prescribing. People often store their medication for future use and may consume it without realizing it. However, this can be dangerous if the medication has expired, potentially harming their health. To prevent this, notifications and reminders are sent when medication has expired.

Most medicines are chemicals, and some people may be allergic to one of these chemicals, or the reaction of two different chemicals can cause dangerous health problems. People know they are allergic but may forget which active ingredients they are allergic to, but allergies are recorded in health systems. If the answer to the question a doctor asks a patient when prescribing medication is wrong, the medication the patient takes can be harmful and lead to worse consequences. To prevent this, the contents of the prescribed medicines, the medicines and substances that should not be used together, and the system organized according to the patient's allergic status, will give an error and eliminate this problem.

In Turkey, a large number of medicine boxes are sold, for example 2.4 billion medicine boxes in 2020. Our program offers a premium membership for pharmaceutical companies, which generates revenue for the state and application writers by supporting sales and marketing. The program provides a file of the most prescribed drugs that are sold in the market to premium companies, allowing them to produce drugs in areas where they are needed. The range of drugs available in pharmacies is set to expand. With this increased diversity, pharmaceutical companies will compete to produce higher quality drugs. This competition will result in an improvement in drug quality, while prices will decrease, making it easier for the public to access high-quality, affordable drugs. 2.4 billion labels were printed accordingly and there was a waste of paper, and a significant portion of these medicines consist of drugs prescribed by doctors for profit. As the world's population grows and the need for green spaces increases, our application will be able to track both the wrong medication given to the patient and the location of the printed labels through the pages in the application. The drugs entered by the doctors from the panel will be monitored and if they exceed a certain quota, the system will send a notification to make an audit.

Finally, after the adoption of e-prescription in Turkey, the drugs prescribed in different cities can be taken by people who are not sick and this cannot be tracked. Our application will be able to provide their medicines with the QR code that is instantly generated by the pharmacist and the QR code that is instantly received on the phone, and in this way the use of drugs by people who should not be using drugs will be prevented.

## 1.2. Objectives

- Keeping a record of the prescribed medication and recording it quickly in the system.
- Only authorized people can access the system, prescribe and monitor medication.
- Information on the frequency of use and side effects of medicines recorded in the system.
- Keeping the health record up-to-date by establishing a connection with electronic health systems.
- To prevent paper waste by reducing the labels used on medicines.
- Reducing unconscious and wrong drug use.
- To help patients quickly and accurately find answers to their questions.
- Provide companies with separate analyses for sales and marketing to increase competition in the market.

## 2. High Level Functionality

The notifications provided to patients in the application will inform them about the upcoming medicine. Furthermore, we aimed to provide accurate information to patients through both live support and educational resources.

The controls we will implement in our system will enable doctors to prevent potential mistakes when prescribing medicines. The QR code will be generated after a doctor prescribes a medicine will allow pharmacists to give the medicine to patients more quickly and efficiently. Furthermore, the controls incorporated into the system will enable the authorities to be informed of any potential irregularities in the medicines and brands prescribed by a doctor. This will facilitate the detection and investigation of suspicious situations.

The premium service provided by the system will enable companies to gain a more detailed understanding of the specific pharmaceuticals required in different regions, allowing them to make more informed investment decisions.

### 2.1. Requirements

#### 2.1.1. Functional Requirements

- The system should allow doctors to prescribe medication and record it quickly in the system. In cases where it is necessary, doctors should be able to prescribe risky medications.

- QR codes should be generated for prescriptions, allowing easy access to medication details and preventing misuse.
- The system should provide comprehensive information on the frequency of use and potential side effects of prescribed medicines.
- Users should receive notifications and reminders for taking their medication at the prescribed times even they are offline.
- Users should have access to live support, including both artificial intelligence and 24/7 human assistance, to resolve queries. The use of AI assistance will be limited to instances where there is a query regarding the use of the app.
- The system should offer an extensive library of videos and animations to educate users on the correct usage of prescribed medicines.
- The system should offer premium memberships to pharmaceutical companies, providing them with sales and marketing analyses.
- The system should monitor prescribed medications and notify authorities if any irregularities or overprescribing occur.

### 2.1.2. Non-Functional Requirements

- The system should ensure data security and privacy to protect users' personal and health information.
- The system should be reliable, with minimal downtime and accurate medication records.
- The system should be able to scale up to accommodate a growing user base and increasing data volumes.
- The user interface should be intuitive and user-friendly, catering to users of varying technical proficiency.
- The system should comply with relevant regulations and standards for healthcare and data management.
- The system should provide reliable customer support services to address user concerns and issues promptly.

## 3. Stakeholders

### 3.1. Internal Stakeholders

- **Developers:** Developers play a crucial role in the ongoing development of the MediShare project, as well as in the secure storage and management of increasing amounts of data. Their main responsibilities include maintaining the program's security, updating the interface, and ensuring that software is reliable.

### 3.2. External Stakeholders

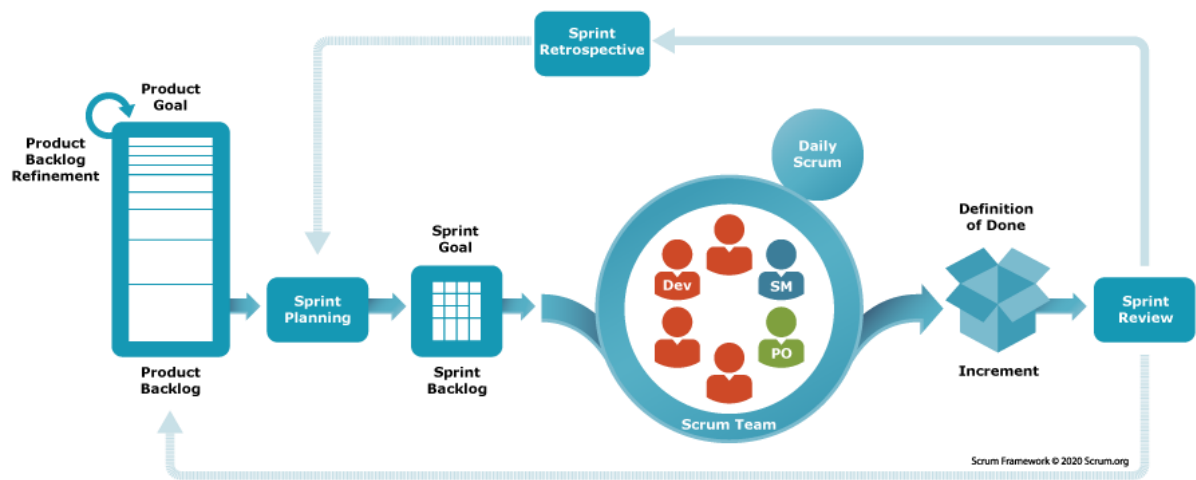
- **Doctors:** The system's controls are going to prevent doctors from prescribing medicines that will cause side effects. If it is necessary to prescribe a risky medicine, doctors can prescribe that medicine after filling out a form to control the situation easily. If the project is successful, the unethical practice of prescribing medication only from a certain company will be prevented. Lastly, the live support system will reduce the doctors' workload by answering the minor medical questions of patients. Since pharmacy and healthcare are unpredictable, doctors can inform us of unforeseen circumstances so we can modify the program. Also, we can adjust the design for ease of use since they will be using it.
- **Nurses:** In urgent situations, nurses can intervene more quickly, effectively, and safely by reviewing the patient's previous treatments and allergies stored in the program.
- **Hospital Receptionist:** Thanks to the live support feature that comes with our program, hospital receptionists may need to consider alternative career paths.
- **Patients:** Reducing the likelihood of incorrect treatment and medication will prevent undesirable consequences. The use of medication will be improved thanks to the alarm system and medication guide. Additionally, the added live support feature will ensure faster and more accurate responses to patients' questions.
- **Pharmacist:** Pharmacists can provide medication to patients more quickly and accurately by using QR codes instead of prescription numbers.

- **Pharmaceutical Manufacturers:** Successful implementation of the MediShare system will enable manufacturers to adjust their production accordingly and better target the market, resulting in increased profitability. If drug manufacturers provide information about allergenic substances in medicines, they can be quickly and accurately added to the system, preventing potential issues.
- **Doctor Supervisors:** If the program proves to be successful the Doctor supervisors can both identify suspicious situations and investigate the situation easily using the tracked data of the suspicious doctor.
- **Lawyers:** Lawyers will be providing us with legal information to protect the rights of both users and our company.
- **Government:** The government will provide us with resources, necessary permits, and allow us to build systems using their data.

## 4. Software Process

We have selected Scrum as the methodology for this project. The primary reason for this decision is the continuous stakeholder collaboration that Scrum provides. As the majority of requirements will be provided by stakeholders, it is crucial for us to facilitate their participation in the project. These are the additional reasons why we selected this model:

- **Flexibility:** Pharmaceuticals and healthcare are unpredictable, requiring us to adapt constantly to new requirements, diseases, medicines, and laws.
- **Manageability:** The system's high requirements and stakeholders increase workload, but dividing it into smaller parts and regularly reviewing can make it more manageable.
- **Transparency:** Increased workload visibility enables more effective workload management by preventing misunderstandings and ensuring clear progress for developers and stakeholders, despite the lack of familiarity with the field.
- **Testability:** Collaborating with stakeholders in the health and pharmaceutical sector can provide us with valuable insights into extreme cases, enabling us to prepare for potential testing scenarios they may not have considered.
- **Enhanced Quality:** By cooperating with stakeholders and adapting to new information, we can ensure that our application is reliable, useful, and complete for users. This approach ensures sustainability and reliability due to continuous updates and live support systems.
- **Detecting Mistakes:** The project's collaborative review model enables efficient error detection and collaboration among team members, making it an optimal choice for complex projects involving human health.



**Figure 1: Scrum Framework**

## 5. Software Risks

### 5.1. Likelihood Risk List

LIKELIHOOD RANK	RISK DESCRIPTION
1	<b>Change in Requirements</b>
2	<b>Data Loss</b>
3	<b>Testing</b>
4	<b>Design Complexity</b>
5	<b>Rate of Bugs</b>



6	<b>Integration with Old Systems</b>
7	<b>Project Expansion</b>
8	<b>Budget</b>
9	<b>Tools</b>
10	<b>Change in Laws</b>

## 5.2. Impact Risk List

IMPACT RANK	RISK DESCRIPTION
1	<b>Testing</b>
2	<b>Rate of Bugs</b>
3	<b>Budget</b>
4	<b>Data Loss</b>
5	<b>Change in Laws</b>
6	<b>Integration with Old Systems</b>
7	<b>Project Expansion</b>
8	<b>Change in Requirements</b>
9	<b>Design Complexity</b>
10	<b>Tools</b>

### 5.3. Combined Risk List

LIKELIHOOD RANK	IMPACT RANK	COMBINED RANK	RISK DESCRIPTION
3	1	4	<b>Testing:</b> Testing in the health and pharmaceutical sectors can be unpredictable, making it highly likely that test cases will be missed. Inexperienced developers such as ourselves may overlook certain cases, so it is important to maintain constant communication with stakeholders to ensure all cases are covered. However, this may lead to unplanned changes in the project, such as a prolonged testing phase, which could delay the start of the next sprint and result in significant costs.
2	4	6	<b>Data Loss:</b> We started learning PostgreSQL during the development of the project. Therefore, there is a high risk of data loss during our operations. Reverting to the old system to recover lost data could be a significant waste of time for developers. Late discovery of data loss may lead to skipped tests and require revisiting those cases to check for bugs, ultimately impacting the project budget and planning negatively.
5	2	7	<b>Rate of Bugs:</b> A high error rate may undermine the government's confidence in us. Additionally, correcting these errors will require a considerable amount of time, which increases the risk of exceeding the project's time and budget limits. This risk is further compounded when combined with other factors such as complex requirements, technical difficulties, or poor communication.
1	8	9	<b>Change in Requirements:</b> Our stakeholders may not anticipate every situation, or they may have explained it incorrectly, or we may have misunderstood it. Therefore, we may need to modify the requirement.
8	3	11	<b>Budget:</b> Other risks can negatively impact planning and cause the budget to be exceeded.
6	6	12	<b>Integration with Old Systems:</b> It is important to understand the documentation of old systems and their integration into our own system. This may lead to unexpected changes.
4	9	13	<b>Design Complexity:</b> Designing an application that is user-friendly for people of all ages can be challenging, as everyone has different needs and preferences.
7	7	14	<b>Project Expansion:</b> The team may have underestimated the size of the project. Additional and changing features may complicate the project's development.

10	5	15	<b>Change in Laws:</b> Legal changes during project development may alter the process, particularly with regards to data confidentiality.
9	10	19	<b>Tools:</b> We need to learn some tools for this project. During the learning process, we may encounter issues or need to customize the tools to meet our specific requirements.

## 6. Project Needs

### 6.1. Software Needs

- **Version Control:** We will use Github Desktop version 3.3.13, which is the latest available. Each developer should install it on their own computer. Our team does not require an upgrade unless there is a security update. If an update is necessary, it should be done on a Saturday, unless it is critical. In such cases, it should be done as early as possible and during break times.
- **Languages:** We will be using Flutter 3.22 and Python 3.12 for our project. Each developer will be responsible for setting up their own computer. Unless there is a security update, we will not be updating. If there is a security update, we will update Flutter and Python the following Sunday.
- **Database:** PostgreSQL version 15.3.1 is a reliable and scalable database capable of handling large amounts of data smoothly. It is recommended that each developer performs their own setup of PostgreSQL. Any updates should be made on one computer firstly then other computers on weekends.
- **IDE:** In MediShare, Visual Studio version 1.88 will be used. Each developer will set up its own computer. We will not need any update.
- **Dialogflow:** The latest version (April 4 2024) of Dialogflow will be used, and installation will be carried out by the IT team during the specific sprint. Any updates will be monitored and installed at 6 PM.
- **Documentation:** Google Docs will be used as it is an online platform that does not require installation or updates.
- **Statistic:** We will be using Excel 2021 and Pandas 2.2.2. Each developer is responsible for installing the required Excel application and Pandas library. Updates will be held on Sunday if necessary.
- **Communication:** For our project's communication, we will be using Discord version 284140. Each developer is required to install Discord on their own computer. Discord updates are quick and error-free, but to be safe, we will have an update break on Sundays.

## 6.2. Hardware Needs

- **Servers:** Our project will use a client-server architecture, with each feature (entering medication, making appointments, looking at old treatments and medications, live support) processed on a separate server. This approach prevents disruptions to other processes in case of a crash. We require at least five servers with powerful processors and high GB RAM. Finally, these servers must implement load balancing in case the process is overloaded and other operations are not performed. For this algorithm, we have considered weighted minimum connection because it is both dynamic and we do not assume that each process will consume the same power. Furthermore, these servers may require additional features, such as camera systems and cooling systems, to be installed.
- **Network:** A device that connects and directs data between different devices in a network. It allows these devices to communicate with each other. The network setup requires additional items like Ethernet cables and mounting equipment. This setup is done at the start of the project to enable communication between devices. System administrators are responsible for setting it up and managing it.

## 6.3. Support Needs

- **Technical Support:** Technical support will be required, particularly if any issues arise after the server installation and during the project development phase. They will also assist in resolving any server-related problems encountered during testing. Hardware and software related to both computers and servers should be sold with or without the help of support technicians. Support technicians should provide remote or onsite support.
- **Training Support:** As the project requires the acquisition of new skills, it is necessary to provide training to the team by experts in the field. It is essential that these training sessions are planned and completed in advance of their requirement in the project.
- **Stakeholder Support and Feedback:** It is crucial that we maintain engagement with our stakeholders, as they can provide us with vital information and feedback. By understanding their perspective, we can redefine our requirements, identify situations not covered in our test cases and add new requirements.

## 7. Software Measurements

There are a few number of questions we want to answer during the development of the project:

1. How reliable is this application?
2. What accomplished during certain period of time?
3. How much has the architecture of this system changed?
4. How well does the project meet user requirements and expectations?
5. What lessons have been learned from project successes and challenges?
6. How effective was the sprints?
7. Has everyone done their part?
8. How much effort did this project require?

This questions requires several software measurements to be included.

- **Total Number of Tests For Each Sprint:** The number of tests conducted to enhance and increase the reliability of our program will be documented in the project table prior to each sprint review. Based on the test results, we will determine whether the program meets the requirements. We will also compare the number of these tests with the number of newly added components and functions to achieve a close ratio. If the ratio is lower than this, we will review the tests and add new ones after discussing with stakeholders. (Associated Question : 1)
- **Total Number of Errors After Each Test Phase:** The test errors will be reviewed, and the task distribution will be reassessed using git commits. For instance, if an individual consistently receives errors, it may indicate that the workload is too much for one person, or that someone else should take over their tasks. (Associated Question : 1)
- **Lines of Code:** To evaluate the project's progress, we will review the number of codes before each sprint review. This will provide insight into the amount of work completed and the efficiency of resource utilization. (Associated Questions : 2, 6)
- **Updates on the Codes (Git Commits):** Git commits will be used to evaluate task distribution and performance of an individual. They will also be used for identifying who received an error during testing. (Associated Questions : 2, 6, 7)
- **Number of Methods Added in That sprint:** This number will be updated prior to each sprint review. This will help estimate the project's progress and determine the number of tests required. (Associated Questions : 2, 6)
- **Number of Changed Requirements After Each Sprint:** By examining this number, we can determine the effectiveness of our planning and identify areas that require attention in future planning. Additionally, it provides insight into our ability to meet user needs. (Associated Questions : 3, 4)

- **Stakeholder Reviews:** After each sprint, user reviews will be kept in a separate file. The file will contain the user's name and feedback. Based on this feedback, requirements will be added and written in different colors, while recommendations will be written in a separate color. The color red indicates that a requirement must be implemented based on stakeholder feedback. Blue indicates that a requirement has been successfully implemented after the feedback, and green indicates considerations for improving the usability of the developed application. This approach will demonstrate the extent to which we meet users' needs and can serve as a reference for future projects. (Associated Question : 4)
- **Time Spend on Project:** This data will provide information on individual contributions and sprint planning and adherence. We will use this information to optimise project progress by adjusting sprint duration and task allocation. We will attempt to progress the project optimally by replanning sprint durations and workload distribution. (Associated Question : 8)
- **Time Spend on Sprint Retrospective:** By examining the time spent in the sprint retrospective, it will become apparent whether the sprint planning was effective. If the retrospective period is lengthy, it suggests that the sprint planning was flawed from the outset. This indicates that the developers may not have fully grasped the Scrum framework or that there may be miscommunication between them. (Associated Question : 5)

## 8. Software Tools

### 8.1. Database Management

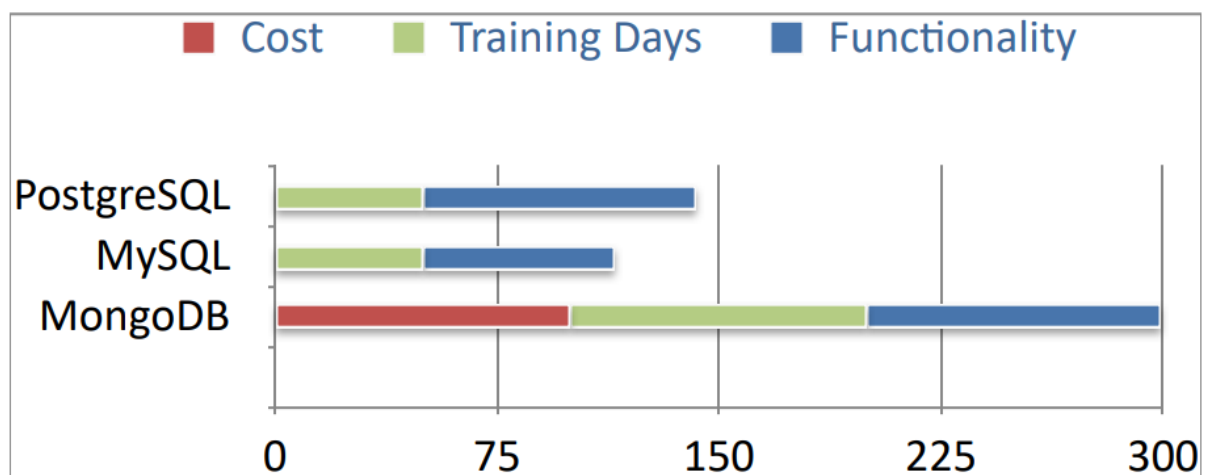
#### 8.1.1. Tool Cost/Training/Functionality Table

Tool	PostgreSQL	MySQL	MongoDB
Cost	0 US\$	\$0.24	57 US\$
Training Days	30	30	60
Functionality	65	45	70

#### 8.1.2. Normalized Cost/Training/Functionality Data

Tool (Normalized)	PostgreSQL	MySQL	MongoDB
Cost	0	0.42	<b>100.0</b>
Training Days	50	50	<b>100.0</b>
Functionality	92	64	<b>100.0</b>

#### 8.1.3. Normalized Tool Graph



**PostgreSQL:** We selected PostgreSQL because it is more reliable, more convenient than others. We eliminate MongoDB because we are more familiar with SQL languages. PostgreSQL's data integrity features, including constraints, foreign key relationships, and transactions, ensure the accuracy and consistency of stored data, crucial for maintaining medication records and patient health information. PostgreSQL is a scalable database that can handle large data volumes (32 TB) and concurrent user access without compromising performance as your application grows. PostgreSQL's active community offers tutorials and forums for troubleshooting issues and optimizing database performance also tutorials about tools such as pgModeler that can simplify the design process.

## 8.2. Analytics and Reporting

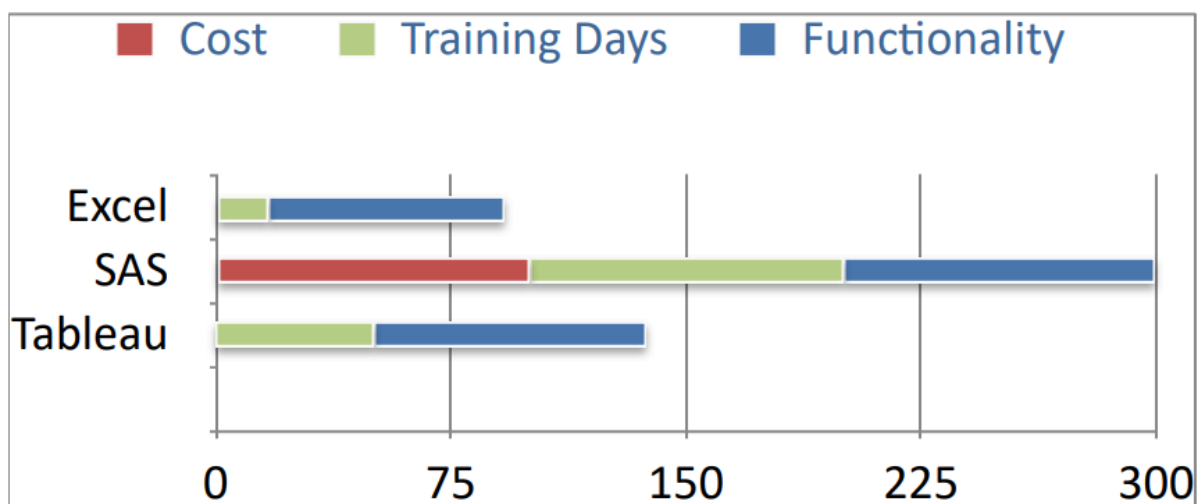
### 8.2.1. Tool Cost/Training/Functionality Table

Tool	Excel	SAS	Tableau
Cost	0 US\$	10000 US\$	900 US\$
Training Days	5	30	15
Functionality	60	80	70

### 8.2.2. Normalized Cost/Training/Functionality Data

Tool (Normalized)	Excel	SAS	Tableau
Cost	0	100.0	0.09
Training Days	16.7	100.0	50
Functionality	75	100.0	87.5

### 8.2.3. Normalized Tool Graph



**Excel:** We selected Excel as our preferred software. The decision to use Excel was made for several reasons. Firstly, it is a cost-effective solution. Secondly, there is a wealth of learning resources available. Thirdly, we believe it will be more useful to the developer team in the long term. Finally, it allows us to write the necessary functions in the program if necessary.



## 8.3. Live Support

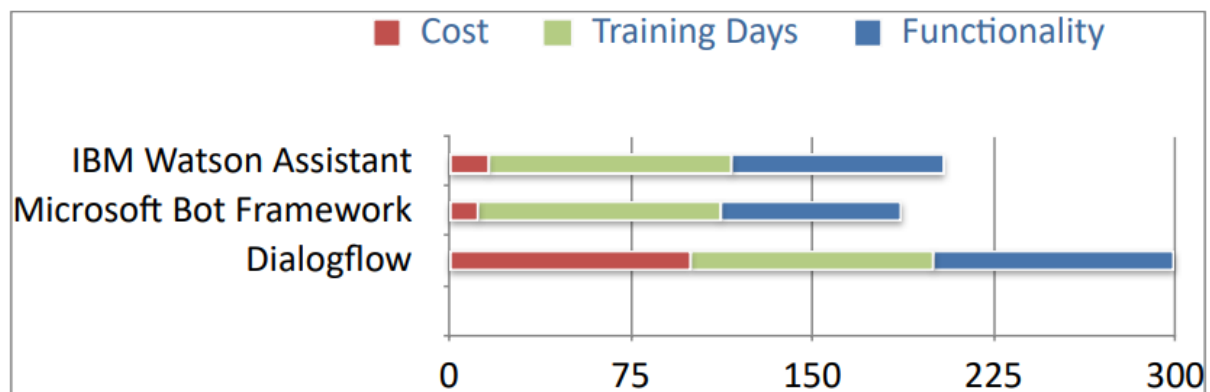
### 8.3.1. Tool Cost/Training/Functionality Table

Tool	IBM Watson Assistant	Microsoft Bot Framework	Dialogflow
Cost	1680 US\$	1188 US\$	10000 US\$
Training Days	20	20	20
Functionality	70	60	80

### 8.3.2. Normalized Cost/Training/Functionality Data

Tool (Normalized)	IBM Watson Assistant	Microsoft Bot Framework	Dialogflow
Cost	16,80	11,88	<b>100.0</b>
Training Days	<b>100.0</b>	<b>100.0</b>	<b>100.0</b>
Functionality	87.5	75	<b>100.0</b>

### 8.3.3. Normalized Tool Graph



**Dialogflow:** We selected Dialogflow for its ability to provide high scalability and reliability in handling large user requests and concurrent interactions. The key factor in our decision was our ability to integrate Dialogflow with Flutter.

## 9. Project Schedule

This section is for showing the schedule of our project. Each sprint in the development phase is shown separately, with the database and testing activities presented as ongoing activities throughout the project.

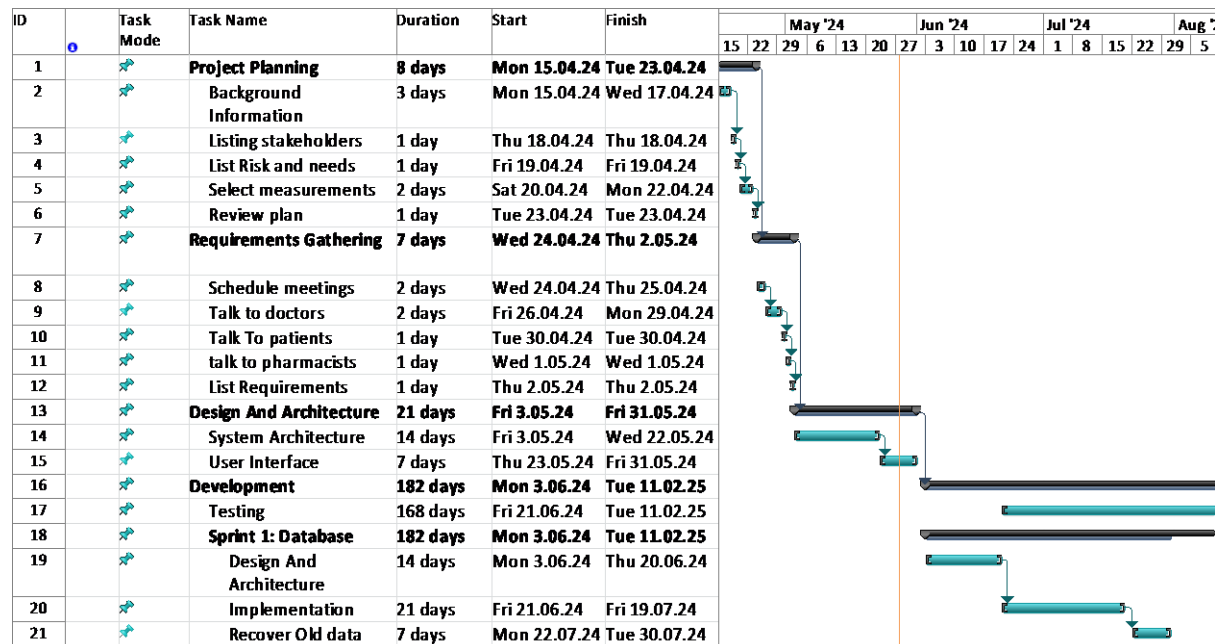


Figure 2: Schedule of the Project and Gantt Chart of the Schedule (1-21)

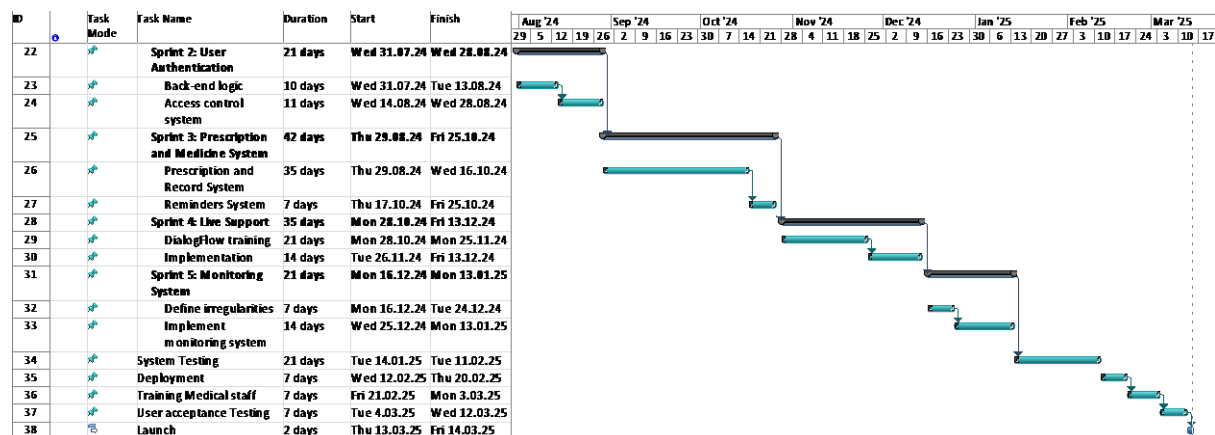


Figure 3: Schedule of the Project and Gantt Chart of the Schedule (22-38)

## 10. Project Payoffs

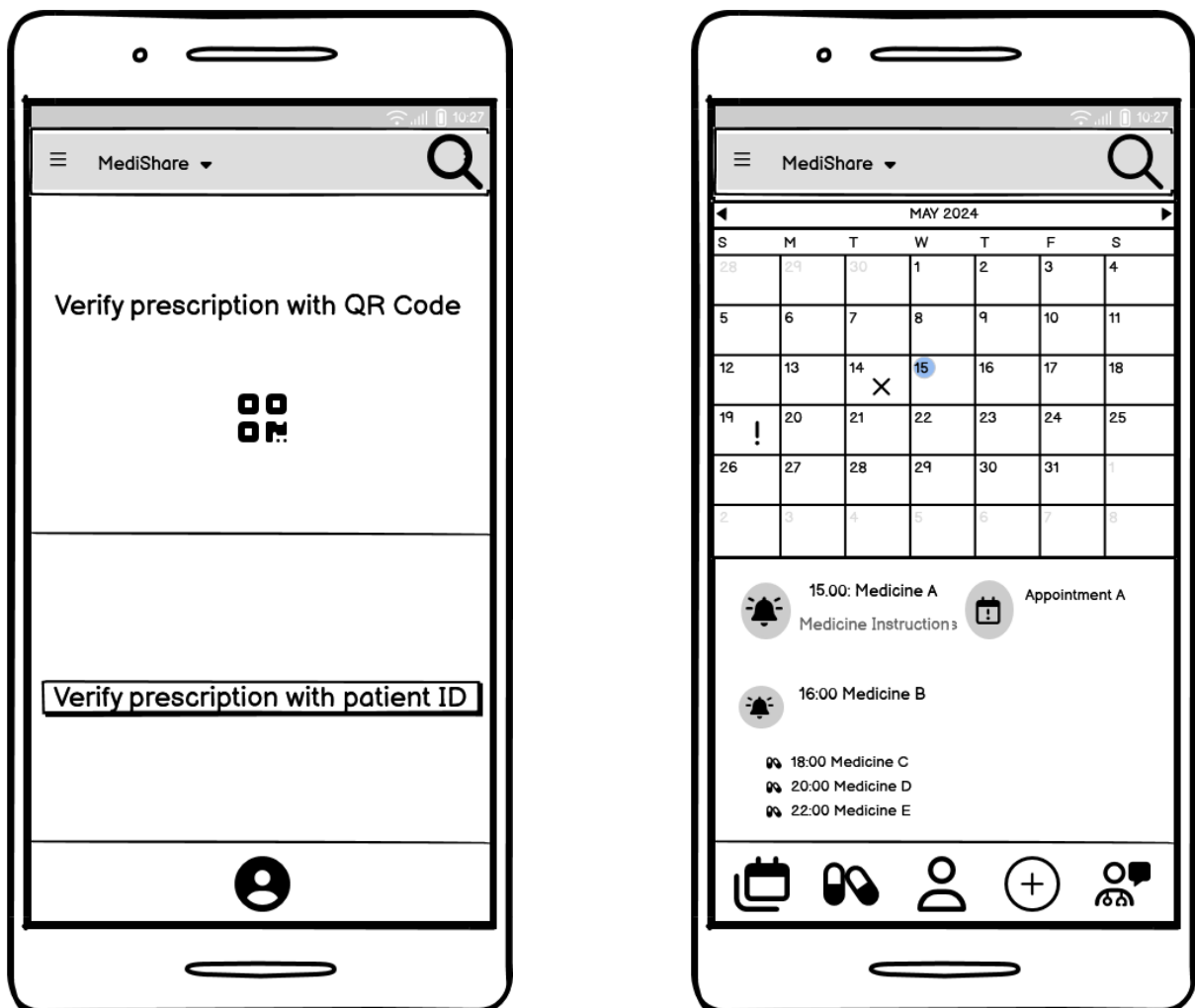
If the project is successfully implemented;

- Pharmaceutical companies will be able to make their investments more accurately.
- The value of developers in the team will increase as they gain new skills and experience.
- Patients will benefit from enhanced appointment scheduling, reliable information and follow their medication more easily.
- All measurements will be useful for us as they can be referenced in future projects.
- Government will be able to detect irregularities earlier and more easily.

## 11. User Interfaces

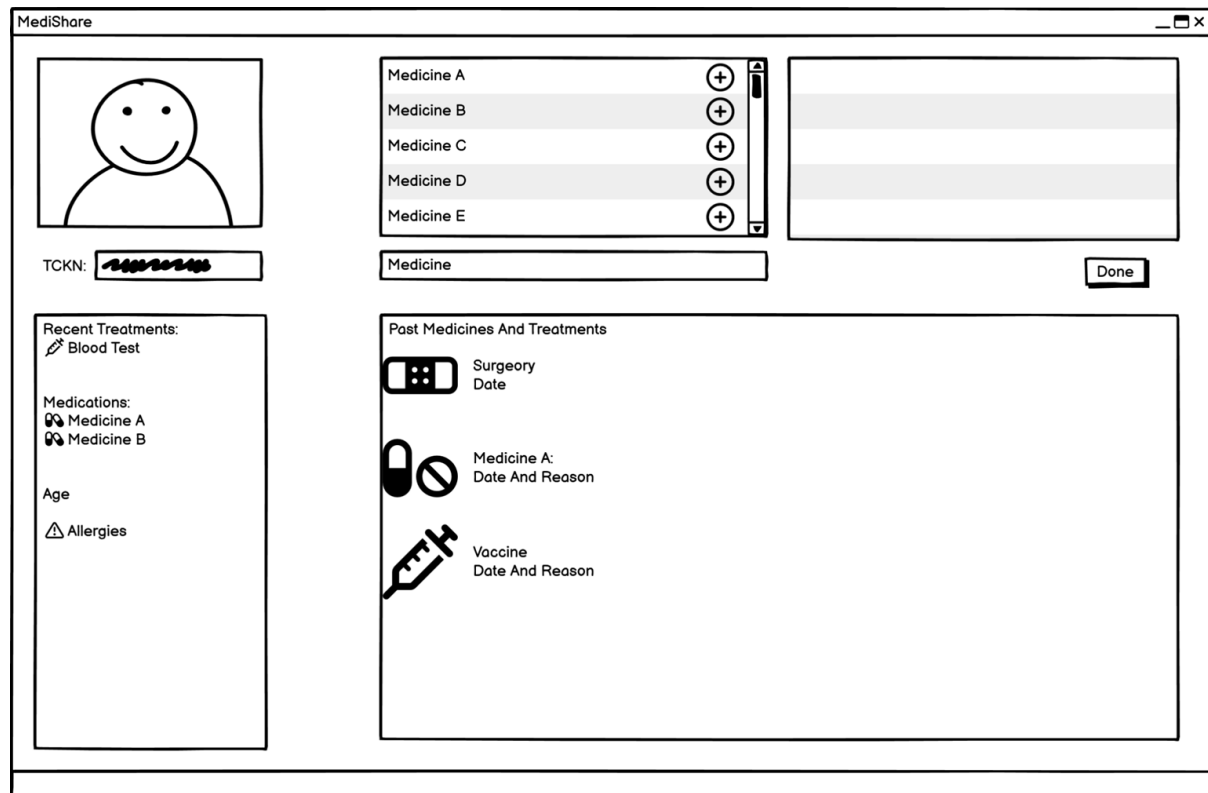
### 11.1. Mobile App Interface

The mobile app will be for pharmacist and patient. These are the example main pages for our mobile app.



## 11.2. Desktop App Interface

The desktop app will be for doctors. These are the example main pages for our desktop app.



## 12. Conclusion

To summarize our project in detail we explained each section shortly.

In terms of high-level functionality, our project's primary objective is to ensure that all users, regardless of age, can utilise the application with ease and access it with minimal effort.

In our view, every stakeholder is important to us. The feedback we receive allows us to identify areas for improvement and inform our future work. Doctors and nurses will be able to prescribe medication quickly when necessary, pharmacists will be able to work more efficiently thanks to the QR code. As competition intensifies among pharmaceutical companies, the drive for innovation will intensify as well. This will lead to the development of more effective and profitable drugs. Furthermore, it will prevent the unnecessary expenditure of resources by focusing on the most appropriate drug. Some professions, such as doctor supervisors, medical receptionists and lawyers, will experience significant changes in workload and work frequency. From the state's perspective, additional income will be generated through the premium, with the potential for significant progress in the health sector.

In terms of project needs, we will leave the needs such as security and testing to the relevant companies, thereby reducing our burden. We will also try to balance the cost by using company employees to a

large extent in needs such as technical support, training and updates. This will help us to balance the resource-time-feature triangle.

We selected Scrum as our software process because it is straightforward to identify and address errors, and because progress is transparent. Its flexibility, manageability and stakeholder collaboration were key factors in our decision.

When we reached the schedule section, we proceeded by creating a schedule according to our sprints and allocating too much time for everything.

When we are looking at the measurements, we sought to identify a consistent trend by focusing on numerical data and using it to assess the project's progress, the efficiency of the working environment, and the completion times of various tasks. We also incorporated feedback from our stakeholders to reinforce our numerical data.

In the project risks, we brainstormed with our team and experts in the field to come up with a list of which risks are most and least likely to occur. From this list, we looked at which risk would cause the most damage if it did occur, and finally we did our best to mitigate these risks by combining them into a single list.

We had three main criteria for tools: Training days, functionality and cost. In addition to these three criteria, we also looked at the languages and libraries we would be using. As a result, we chose the most optimal tools we could use.

As for the payoffs, we have already touched on these points throughout the project, but to go over them again, with the statistics provided with the premium membership, pharmaceutical companies will make their investments in a single area or even a single drug, and will both spend their money correctly and increase their own profits by increasing competition. Thanks to the different tools, methods and techniques used, as well as the fact that the project is being carried out outside the usual environment, the developers involved in the project will be able to acquire new skills and improve their existing ones. The features offered to patients using the program will make their work easier in many areas. They will be able to learn how to use their medicines, make appointments more easily, find reliable sources of information more quickly and not be exposed to information pollution.

We believe this project will be a success due to the potential for doctors to utilise it, as it will reduce their workload and the risk of errors. Furthermore, as it reduces the likelihood of errors, its use by patients may also increase. Furthermore, by providing accurate and accessible information to patients, including details on drug use and content, as well as the implementation of a drug reminder system, will increase the usage of our application by patients, thereby enabling pharmacists to dispense medicines to patients in a safer and more efficient way. Consequently, companies that wish to leverage the data for sales purposes will be able to offer a premium service.