**EGERTON UNIVERSITY**

**DEPARTMENT OF COMPUTER SCIENCE**

**[FACULTY OF SCIENCE]**

**COMP 493**

**COMPUTER SYSTEM PROJECT**

**ON**

**DRUGSTORE MOBILE APPLICATION**

**BY**

**DENNIS MBATIA MATHENGE**

**S13/03116/18**

**SUPERVISOR**

**MR EDWARD KIRUI**

**APRIL, 2023**.

# PART 1: PROJECT PROPOSAL

## ABSTRACT

The proposed pharmacy flutter app aims to provide a convenient and efficient way for customers to order prescription medicines, over-the-counter products, and health supplies from a local pharmacy. The app will feature a user-friendly interface that allows users to search for products, view product information, and add items to their shopping cart. Customers will also be able to upload their prescription, receive notifications on order status, and track their delivery. The app will also integrate a secure payment system for transactions, enabling users to pay through various payment methods. Additionally, the app will have a section that provides health-related news and tips, as well as a pharmacy locator tool for users to find the nearest pharmacy. The pharmacy flutter app will streamline the pharmacy ordering process, enhance customer experience, and improve access to essential health products.

## CHAPTER 1: INTRODUCTION

### ****1.1 LITERATURE REVIEW****

The literature review for the proposed pharmaceutical flutter app for ordering medicine aims to provide an overview of the current state of the pharmaceutical industry, specifically in the area of medication ordering and delivery. The review highlights the challenges facing the industry and explores the role of mobile apps in addressing these challenges.

The pharmaceutical industry has been expanding rapidly in recent years, with a focus on research and development, production, and distribution of medication. However, the process of ordering and delivering medication is still a challenge, and patients often face long waiting periods and difficulty in accessing medication. Additionally, pharmacies struggle to manage their inventory efficiently and provide timely and accurate service to their customers.

Mobile apps have emerged as a solution to the challenges facing the pharmaceutical industry. These apps offer a more convenient and accessible way of ordering medication, and they have the potential to improve the overall patient experience. The use of mobile apps has also been shown to improve medication adherence, leading to better health outcomes for patients.

Several mobile apps for medication ordering and delivery have been developed, such as MYDAWA, M-Tiba and My NHIF. These apps offer features such as medication catalogue, medication search and filter, medication ordering, medication delivery tracking, payment gateway integration, user profile management, and pharmacy inventory management. These features have helped to simplify the process of ordering medication and make it more accessible to patients.

However, there are still challenges to be addressed in the development and implementation of mobile apps for medication ordering and delivery. One of the key challenges is ensuring the security and privacy of user data, especially with regards to personal health information. Additionally, there are technical challenges such as ensuring compatibility with different devices and operating systems, and ensuring that the app is user-friendly and accessible to all users.

In summary, the literature review highlights the challenges facing the pharmaceutical industry in the area of medication ordering and delivery, and explores the potential of mobile apps to address these challenges. The review identifies the features of existing mobile apps in this space and the challenges that need to be addressed in the development and implementation of new apps.

### ****1.2 PROBLEM STATEMENT****

The problem statement for the proposed pharmaceutical flutter app for ordering medicine aims to identify the challenges facing patients and pharmacies in the process of medication ordering and delivery, and how a mobile app can address these challenges.

The current process of medication ordering and delivery can be time-consuming and cumbersome for patients. Patients often have to physically visit a pharmacy, wait in long lines, and then wait again for their medication to be prepared. This process can be particularly challenging for patients with chronic conditions who require regular medication. Additionally, pharmacies may struggle to manage their inventory efficiently, resulting in delays in medication delivery and poor customer service.

The COVID-19 pandemic has further highlighted the need for a more convenient and accessible way of ordering medication, as patients are advised to stay at home and avoid unnecessary physical contact. This has resulted in an increased demand for online medication ordering and delivery services.

Mobile apps have emerged as a solution to the challenges facing the pharmaceutical industry. However, many existing medication ordering and delivery apps have limitations in terms of functionality, user experience, and accessibility. Some apps may not have a comprehensive medication catalogue or may have limited delivery options. Other apps may be difficult to navigate or may not be compatible with all devices and operating systems.

Therefore, there is a need for a mobile app that addresses the challenges facing patients and pharmacies in the process of medication ordering and delivery. This app should be user-friendly, accessible, and secure, with features such as a comprehensive medication catalogue, medication search and filter, medication ordering, medication delivery tracking, payment gateway integration, user profile management, and pharmacy inventory management. By addressing these challenges, the proposed pharmaceutical flutter app for ordering medicine will improve the patient experience and streamline the process of medication ordering and delivery.

### ****1.3 JUSTIFICATION****

The justification for the proposed pharmaceutical flutter app for ordering medicine aims to provide a rationale for why the app is necessary and how it will benefit patients and pharmacies.

Firstly, the proposed app will provide a more convenient and accessible way for patients to order medication. Patients will no longer have to physically visit a pharmacy, wait in long lines, and then wait again for their medication to be prepared. Instead, they can use the app to browse a comprehensive medication catalogue, place an order, and have their medication delivered to their doorstep. This will save time and effort for patients, especially those with chronic conditions who require regular medication.

Secondly, the proposed app will improve the patient experience by providing features such as medication search and filter, medication delivery tracking, and user profile management. Patients will be able to easily find the medication they need, track their medication delivery in real-time, and manage their user profile and medication history. This will make the process of medication ordering and delivery more transparent and efficient, leading to higher patient satisfaction.

Thirdly, the proposed app will benefit pharmacies by helping them to manage their inventory more efficiently. The app will provide features such as pharmacy inventory management, which will enable pharmacies to track their stock levels and restock medication as needed. This will help to prevent stockouts and delays in medication delivery, resulting in better customer service and increased revenue for pharmacies.

Fourthly, the proposed app will address the need for online medication ordering and delivery services, which has become more important during the COVID-19 pandemic. The app will provide a safe and secure way for patients to order medication from the comfort of their own homes, without having to physically visit a pharmacy. This will help to reduce the risk of transmission of COVID-19 and other infectious diseases.

In summary, the proposed pharmaceutical flutter app for ordering medicine is necessary to address the challenges facing patients and pharmacies in the process of medication ordering and delivery. The app will provide a more convenient and accessible way for patients to order medication, improve the patient experience, benefit pharmacies by helping them to manage their inventory more efficiently, and address the need for online medication ordering and delivery services during the COVID-19 pandemic.

### ****1.4 RESEARCH QUESTIONS****

The research questions for the proposed pharmaceutical flutter app for ordering medicine aim to guide the research process and provide a framework for investigating the feasibility and effectiveness of the app. The research questions are as follows:

1. What are the current challenges facing patients and pharmacies in the process of medication ordering and delivery?

This research question aims to identify the specific challenges facing patients and pharmacies in the medication ordering and delivery process, such as long wait times, inventory management issues, and poor customer service.

1. How can a pharmaceutical flutter app for ordering medicine address these challenges?

This research question aims to explore the features and functionality that should be included in the app to address the challenges identified in the first research question, such as a comprehensive medication catalogue, medication search and filter, medication ordering, medication delivery tracking, payment gateway integration, user profile management, and pharmacy inventory management.

1. What is the feasibility of developing a pharmaceutical flutter app for ordering medicine?

This research question aims to assess the technical and financial feasibility of developing the app, such as the availability of resources, the cost of development, and the compatibility of the app with different devices and operating systems.

1. How will the proposed app impact the patient experience and pharmacy operations? This research question aims to evaluate the effectiveness of the app in improving the patient experience and streamlining pharmacy operations, such as reducing wait times, improving medication delivery, and increasing revenue for pharmacies.
2. What are the security and privacy considerations that should be taken into account when developing a pharmaceutical flutter app for ordering medicine?

This research question aims to identify the potential security and privacy risks associated with developing a medication ordering and delivery app and explore ways to mitigate these risks, such as using secure payment gateways, implementing user authentication and authorization, and complying with data protection regulations.

By answering these research questions, the proposed pharmaceutical flutter app for ordering medicine can be developed and evaluated in a systematic and evidence-based manner, leading to a more effective and efficient medication ordering and delivery process for patients and pharmacies.

### ****1.5 RESEARCH OBJECTIVES****

The research objectives for the proposed pharmaceutical flutter app for ordering medicine are specific and measurable goals that aim to address the research questions and guide the research process. The research objectives are as follows:

1. To identify the specific challenges facing patients and pharmacies in the process of medication ordering and delivery. This objective aims to conduct a thorough review of the literature and gather data from patients and pharmacies to identify the challenges they face in the medication ordering and delivery process. This objective will help inform the features and functionality of the app.
2. To design and develop a pharmaceutical flutter app for ordering medicine that addresses the challenges identified in the first research objective. This objective aims to use the findings from the first research objective to design and develop a pharmaceutical flutter app for ordering medicine that addresses the challenges identified. The app will include features such as a comprehensive medication catalogue, medication search and filter, medication ordering, medication delivery tracking, payment gateway integration, user profile management, and pharmacy inventory management.
3. To evaluate the feasibility of developing a pharmaceutical flutter app for ordering medicine. This objective aims to assess the technical and financial feasibility of developing the app by conducting a cost-benefit analysis, evaluating the availability of resources, and testing the compatibility of the app with different devices and operating systems.
4. To evaluate the effectiveness of the app in improving the patient experience and streamlining pharmacy operations. This objective aims to evaluate the effectiveness of the app in improving the patient experience and streamlining pharmacy operations by conducting user testing and gathering feedback from patients and pharmacies. The evaluation will focus on metrics such as wait times, medication delivery, and revenue for pharmacies.
5. To identify and address security and privacy risks associated with developing a pharmaceutical flutter app for ordering medicine. This objective aims to identify potential security and privacy risks associated with developing a medication ordering and delivery app and develop strategies to mitigate these risks, such as using secure payment gateways, implementing user authentication and authorization, and complying with data protection regulations.

By achieving these research objectives, the proposed pharmaceutical flutter app for ordering medicine will be developed and evaluated in a systematic and evidence-based manner, leading to a more effective and efficient medication ordering and delivery process for patients and pharmacies.

## CHAPTER 2: PROPOSED SOLUTION

### ****2.1 REQUIREMENTS ANALYSIS****

The app will have the following features:

1. User Registration and Login:

User registration and login is a crucial feature in the proposed pharmaceutical flutter app for ordering medicine. This feature enables users, including patients, pharmacists, and delivery personnel, to create an account on the app and access its various features. During registration, users will be required to provide their personal information, such as name, address, and phone number, which will be used to facilitate medication ordering and delivery. Upon successful registration, users can log in to the app using their username and password. The registration and login process will be designed to ensure that user data is secure and protected from unauthorized access.

1. Medication Catalogue:

The medication catalogue is a comprehensive list of the medications available for ordering through the app. The medication catalogue will include information on medication names, dosages, indications, contraindications, side effects, and pricing. The medication catalogue will be regularly updated to ensure that it remains up-to-date with the latest medications available in the market.

1. Medication Search and Filter:

The medication search and filter feature will enable users to quickly and easily find the medications they need. This feature will allow users to search for medications by name, dosage, indication, or other relevant criteria. Users will also be able to filter medications by price, availability, or other relevant criteria. The medication search and filter feature will help to improve the user experience by enabling users to find the medications they need quickly and easily.

1. Medication Ordering:

The medication ordering feature will allow users to place an order for the medications they need. Users will be able to add medications to their cart, select a delivery method, and provide payment information. Users will also be able to review their order before submitting it. Once an order is submitted, the app will send a confirmation to the user and the pharmacy.

1. Medication Delivery Tracking:

The medication delivery tracking feature will allow users to track the status of their medication orders. This feature will enable users to see when their medications have been dispatched from the pharmacy, and when they are expected to arrive. The medication delivery tracking feature will help to improve the user experience by providing users with real-time updates on the status of their orders.

1. Payment Gateway Integration:

The payment gateway integration feature will allow users to make payments for their medication orders directly through the app. The payment gateway integration feature will only support M-PESA as the payment method. The payment gateway integration feature will be designed to ensure that user data is secure and protected from unauthorized access.

1. User Profile Management:

The user profile management feature will allow users to manage their personal information, such as their name, address, and phone number. Users will be able to update their profile information as needed. The user profile management feature will help to improve the user experience by enabling users to keep their information up-to-date.

1. Pharmacy Inventory Management:

The pharmacy inventory management feature will allow pharmacies to manage their inventory of medications. Pharmacies will be able to update their inventory as needed, add new medications to the medication catalogue, and remove medications that are no longer available. The pharmacy inventory management feature will help to ensure that the medication catalogue remains up-to-date and that users can order the medications they need.

### ****2.2 PLANNING****

The planning phase of the proposed pharmaceutical flutter app for ordering medicine is a critical step that involves defining the project scope, developing a timeline, and identifying resources needed to complete the project. The following sections describe the key aspects of the planning phase.

**Project Scope:**

The first step in the planning phase is to define the project scope. The project scope will define the overall objectives of the project, as well as the features and functionalities that will be included in the app. The project scope will be developed based on the findings from the requirements analysis phase and will be used to guide the development and testing of the app.

**Timeline:**

Once the project scope has been defined, the next step is to develop a timeline for the project. The timeline will include specific milestones and deliverables that must be achieved to complete the project successfully. The timeline will also identify dependencies between tasks and will be used to track progress and ensure that the project is on track.

**Resources:**

The planning phase will also involve identifying the resources needed to complete the project. This will include identifying the team members who will be involved in the project, as well as any external resources, such as consultants or contractors, that may be needed. The planning phase will also involve identifying the technology and infrastructure needed to develop and test the app, including hardware and software.

**Risk Assessment:**

The planning phase will also involve a risk assessment to identify potential risks that could impact the project's success. The risk assessment will identify the likelihood of each risk occurring, as well as the potential impact of the risk on the project. Once risks have been identified, the project team will develop strategies to mitigate each risk and minimize the impact on the project.

**Project Management:**

The planning phase will also involve developing a project management plan that will define how the project will be managed, including the communication plan, change management plan, and quality assurance plan. The project management plan will outline the roles and responsibilities of each team member, as well as the processes and tools that will be used to manage the project.

**Budget:**

Finally, the planning phase will involve developing a budget for the project. The budget will include the costs associated with developing and testing the app, as well as any other costs, such as marketing and support costs. The budget will be used to track project expenses and ensure that the project remains within budget.

### ****2.3 ARCHITECTURAL DESIGN****

The architectural design of the pharmaceutical flutter app for ordering medicine is a critical step that involves designing the structure and components of the app. The following sections describe the key aspects of the architectural design phase:

**System Architecture:**

The first step in the architectural design phase is to define the system architecture. The system architecture will define the overall structure of the app, including the components, modules, and services that will be included. The system architecture will also define how these components will interact with each other to achieve the objectives of the app.

**User Interface Design:**

Once the system architecture has been defined, the next step is to design the user interface of the app. The user interface design will define how the app will look and feel, including the layout, typography, colours, and other design elements. The user interface design will also define how users will interact with the app, including the navigation, buttons, forms, and other interactive elements.

**Backend Architecture:**

The architectural design phase will also involve designing the backend architecture of the app. The backend architecture will define the server-side components and services that will be used to support the app, including the database, APIs, and web services. The backend architecture will also define the security and authentication mechanisms that will be used to protect user data and ensure secure communication between the app and the server.

**Data Management:**

The architectural design phase will also involve designing the data management system of the app. The data management system will define how user data will be stored, retrieved, and updated within the app, including the database schema, data access layer, and data caching mechanisms.

**Integration with Third-party Services:**

The architectural design phase will also involve integrating the app with third-party services, such as payment gateways, delivery services, and other external systems. The integration with third-party services will be designed to ensure seamless communication and data exchange between the app and these services.

**Performance and Scalability:**

Finally, the architectural design phase will involve designing the performance and scalability aspects of the app. The app will be designed to perform efficiently under various user loads, including peak usage times. The app will also be designed to scale easily, allowing for the addition of new users and features without affecting the performance of the app.

### ****2.4 SOFTWARE DEVELOPMENT****

The software development phase of the pharmaceutical flutter app for ordering medicine involves writing the code to implement the design specifications. This phase includes several key steps, as described below:

**Coding:**

The coding phase involves writing the actual code that will implement the design specifications. The code will be written using the Flutter framework, which will allow for the development of cross-platform mobile applications. The coding phase will be broken down into smaller tasks, which will be assigned to individual developers on the team. The code will be reviewed and tested by other developers to ensure that it meets the quality standards.

**Unit Testing:**

Unit testing is a critical step in the software development phase. Unit tests will be written to test individual functions and methods in the code. Unit tests will be automated and will be run frequently throughout the development process. The unit tests will help to identify bugs and errors in the code and ensure that the code meets the functional requirements.

**Integration Testing:**

Integration testing involves testing how different components of the app interact with each other. Integration tests will be performed to ensure that the various components of the app are working together as expected. Integration testing will also involve testing how the app interacts with external systems, such as the payment gateway and delivery services.

**Quality Assurance:**

Quality assurance is an essential part of the software development phase. Quality assurance will involve conducting comprehensive testing to ensure that the app meets the functional and non-functional requirements. Quality assurance testing will include functional testing, performance testing, security testing, and usability testing. Quality assurance testing will be conducted by a team of dedicated testers who will work closely with the development team to identify and resolve any issues.

**Bug Fixing:**

Bug fixing is an ongoing process throughout the software development phase. Any bugs or errors identified during testing will be logged and prioritized. The development team will work to resolve these bugs and ensure that the code meets the quality standards.

**Version Control:**

Version control is an essential part of the software development phase. The development team will use a version control system, such as Git, to manage the codebase. The version control system will allow the team to track changes to the code, collaborate on the codebase, and manage different versions of the code.

**Documentation:**

Documentation is a crucial part of the software development phase. The development team will document the codebase, including the functions, methods, and APIs used in the code. The team will also create user documentation, including user manuals and help files, to assist users in using the app.

### ****2.5 TESTING****

The testing phase is a critical part of the pharmaceutical flutter app development project. This phase involves thoroughly testing the app to ensure that it meets the functional and non-functional requirements. The testing phase includes several key steps, as described below:

**Functional Testing:**

Functional testing is the process of testing the app's features to ensure that they work as intended. Functional testing will include testing the user registration and login features, medication catalogue, search and filter functions, medication ordering, payment gateway integration, user profile management, pharmacy inventory management, and medication delivery tracking. Functional testing will be performed manually and automated, and will include both positive and negative testing scenarios.

**Performance Testing:**

Performance testing is the process of testing the app's performance to ensure that it can handle a high volume of users and transactions. Performance testing will include testing the app's response time, load testing, stress testing, and scalability testing. Performance testing will be conducted using automated tools to simulate real-world usage scenarios.

**Security Testing:**

Security testing is the process of testing the app's security features to ensure that it is secure against various types of attacks. Security testing will include testing the app's authentication and authorization mechanisms, data encryption, input validation, and other security features. Security testing will be performed manually and automated, and will include both black-box and white-box testing.

**Usability Testing:**

Usability testing is the process of testing the app's ease of use and user experience. Usability testing will be conducted with a sample of users who will use the app and provide feedback on the app's usability. The feedback will be used to improve the app's user interface, user experience, and overall usability.

**Regression Testing:**

Regression testing is the process of retesting the app's features after changes have been made to the code. Regression testing will ensure that changes to the code do not introduce new bugs or break existing features. Regression testing will be performed manually and automated and will be conducted after each new release of the app.

**Acceptance Testing:**

Acceptance testing is the final stage of testing and involves testing the app with the client to ensure that it meets their requirements. Acceptance testing will be conducted in collaboration with the client, and the client will provide feedback on the app's functionality, usability, and other aspects. Any issues identified during acceptance testing will be addressed before the app is deployed.

**Documentation:**

Documentation is an essential part of the testing phase. The testing team will document the testing process, including test plans, test cases, and test results. The team will also document any issues identified during testing and the steps taken to resolve them. The testing team will also create user documentation, including user manuals and help files, to assist users in using the app.

### ****2.6 DEPLOYMENT****

The deployment phase is the final stage of the pharmaceutical flutter app development project. This phase involves releasing the app to production and making it available to users. The deployment phase includes several key steps, as described below:

**Release Management:**

The release management process involves planning and managing the app's release to production. The release management team will create a release plan, which includes the release date, release notes, and any other relevant information. The release management team will also manage the release process, ensuring that the app is properly packaged, tested, and deployed to production.

**Deployment Environment Setup:**

The deployment environment setup involves preparing the production environment for the app's deployment. The deployment team will set up the servers, databases, and other infrastructure required to host the app. The team will also configure the servers and databases to ensure that they are secure and scalable.

**Deployment Process:**

The deployment process involves deploying the app to the production environment. The deployment team will use automated deployment tools to deploy the app to production. The deployment team will also conduct manual checks to ensure that the app is functioning correctly after deployment.

**Monitoring and Maintenance:**

Monitoring and maintenance are critical to ensure that the app is functioning correctly after deployment. The monitoring team will set up monitoring tools to monitor the app's performance, availability, and other metrics. The team will also set up alerts to notify them of any issues with the app. The maintenance team will address any issues identified during monitoring and perform regular maintenance tasks, such as software updates and backups.

**User Training:**

User training is an essential part of the deployment phase. The training team will create user training materials, including user manuals and help files, to assist users in using the app. The training team will also conduct training sessions for users to teach them how to use the app.

**Post-Deployment Support:**

Post-deployment support is essential to ensure that the app continues to function correctly after deployment. The support team will provide support to users who encounter issues while using the app. The team will also address any issues identified during monitoring and maintenance. The support team will also provide ongoing maintenance and updates to the app to ensure that it remains secure and up to date.

## CHAPTER 3: FINAL RESULTS

The expected final results of this project are:

1. A fully functional Flutter application for ordering medication from pharmacies.
2. A user-friendly interface that allows users to register and login, search for medications, place orders, track deliveries, and manage their profiles.
3. A medication catalogue with detailed information on each medication, such as its name, price, dosage, and possible side effects.
4. A search and filter function that allows users to easily find the medication they need.
5. A secure payment gateway that accepts various payment methods and protects users' personal and financial information.
6. A pharmacy inventory management system that enables pharmacies to manage their stock levels and receive orders from the app.
7. A medication delivery tracking system that allows users to track their orders in real-time and receive notifications on their delivery status.
8. A testing and debugging process that ensures the app is fully functional, secure, and meets all user requirements.

The expected final results of this project will provide a convenient and accessible way for users to order medication, especially for those who may have mobility or transportation issues. The app will also streamline the process of managing pharmacy inventory and fulfilling medication orders, making it easier for pharmacies to manage their businesses.

## CHAPTER 4: CHALLENGES, OBSTACLES, AND RISKS

Despite the thorough planning and testing process, there are several potential challenges, obstacles, and risks that may arise during the development and deployment of the pharmaceutical flutter app. These include:

1. Technical Challenges: There may be technical difficulties during the development process that can delay the project's progress or cause errors. Technical challenges may include integrating the payment gateway or ensuring that the medication search and filter function is working correctly.
2. Security Risks: The app will contain sensitive information such as user profiles, medication orders, and payment details, which make it susceptible to security breaches. As such, there is a risk of hackers trying to exploit vulnerabilities in the app's security features, which can lead to loss of data and financial losses.
3. User Adoption: While the app's features and benefits have been designed to meet the needs of the users, there is a possibility that some users may not adopt it due to various reasons such as preference for other channels, trust issues, or lack of awareness.
4. Regulatory Compliance: The pharmaceutical industry is highly regulated, and there are specific rules and regulations that must be adhered to when developing and deploying such an app. Failure to comply with these regulations can lead to legal challenges and penalties.
5. Budget Constraints: Developing and deploying a high-quality pharmaceutical flutter app can be expensive. Any unforeseen costs or budget constraints can impact the delivery of the project, leading to delays or a reduction in the app's quality.

To mitigate these challenges, obstacles, and risks, the project team will adopt a proactive approach. This includes conducting regular testing and audits to ensure the app's functionality, partnering with reputable payment gateways, implementing robust security features, and adhering to industry regulations. Additionally, the team will conduct user education and awareness campaigns to promote the app's adoption and provide excellent user experience. The budget will be regularly reviewed, and contingency plans will be put in place to ensure that the project is completed within the allocated budget.

## CHAPTER 5: SCHEDULE AND BUDGET

### 5.1 SCHEDULE

The pharmaceutical flutter app project has a detailed schedule that includes several phases of development and testing. The schedule is designed to ensure that the project is completed within the given timeline and that all deliverables are met.

The schedule includes the following phases:

1. Requirements gathering and analysis: This phase includes gathering and analyzing the requirements for the app. It involves consultations with stakeholders and subject matter experts to identify the key features and functionalities required for the app. This phase is expected to take 2 weeks.
2. Planning and design: This phase includes the planning and design of the app. It involves developing a detailed project plan, creating wireframes and prototypes, and designing the user interface. This phase is expected to take 3 weeks.
3. Development: This phase includes the development of the app. It involves coding, testing, and debugging the app to ensure that it meets all the required standards. This phase is expected to take 8 weeks.
4. Testing: This phase includes the testing of the app to ensure that it is stable and reliable. It involves unit testing, integration testing, and acceptance testing. This phase is expected to take 3 weeks.
5. Deployment: This phase includes the deployment of the app to the target audience. It involves releasing the app to the app store and ensuring that it is available for download. This phase is expected to take 1 week.

The total duration of the project is expected to be 17 weeks, with a buffer period of 2 weeks for any unforeseen delays or issues.

|  |  |
| --- | --- |
| Phase | Duration |
| Requirements gathering | 2 weeks |
| Planning and design | 3 weeks |
| Development | 8 weeks |
| Testing | 3 weeks |
| Deployment | 1 week |
| Total (excluding buffer) | 17 weeks |
| Buffer period | 2 weeks |
| Total (including buffer) | 19 weeks |

### 5.2 BUDGET

Here is a revised budget for the pharmaceutical Flutter app project:

1. Salaries: Free, as it will be a student project.
2. Software and tools: Free, as Firebase software and Figma will be used.
3. Hardware: Kshs 60,000 will be allocated for the purchase of laptops or desktops needed for the project.
4. Data collection: Kshs 1,000 has been allocated for the collection of data.
5. Storage: Kshs 500 has been allocated for the purchase of flash disks.
6. Stationery and printing: Kshs 400 has been allocated for stationery and printing costs.
7. Hosting charges: Kshs 20,000 has been allocated for hosting charges. This will cover the cost of hosting the app on a cloud server and maintaining it to ensure its stability and security for the duration of the project.
8. Miscellaneous costs: Kshs 4,000 has been allocated for miscellaneous expenses that may arise during the project.

Overall, the budget has been carefully planned to ensure that all the necessary expenses required to develop a high-quality pharmaceutical Flutter app are covered, while considering the constraints of a student project budget. The allocation of funds to the different areas has been based on careful consideration of the requirements and needs of the project. The total estimated budget for the project is Kshs 85,900.

|  |  |
| --- | --- |
| Expense Category | Amount (Kshs) |
| Salaries | Free |
| Software and tools | Free |
| Hardware | 60,000 |
| Data collection | 1,000 |
| Storage | 500 |
| Stationery and printing | 400 |
| Hosting charges | 20,000 |
| Miscellaneous costs | 4,000 |
| Total | 85,900 |

## CHAPTER 6: CONCLUSIONS AND FUTURE WORKS

### 6.1 CONCLUSIONS

In conclusion, the proposed pharmaceutical Flutter app for ordering medicine is expected to provide an efficient and convenient way for users to order and track their medications. The app will provide features such as medication search and filter, medication ordering, delivery tracking, payment gateway integration, user registration and login, user profile management, and pharmacy inventory management. These features will enable users to order their medications from the comfort of their homes and have them delivered to their doorstep.

The project team aims to deliver a high-quality, user-friendly, and reliable application that meets all the project requirements and user needs. The development process will follow a structured approach, including requirements analysis, planning, architectural design, software development, testing, and deployment. The project budget is estimated at Kshs 85,900 which is expected to cover all the expenses required to develop and deploy the application.

### 6.2 FUTURE WORKS

There are several potential areas for future work on this project. The app can be enhanced to provide additional features, such as medication reminders, drug interaction warnings, and health tips. In addition, the app can be integrated with electronic health record systems to provide healthcare providers with real-time access to patients' medication history, allergies, and medical conditions.

Furthermore, the app can be expanded to cover more pharmacies and healthcare providers to provide users with a wider range of medication options and healthcare services. Additionally, the app can be localized to support multiple languages and currencies to cater to users in different countries.

In conclusion, the proposed pharmaceutical Flutter app for ordering medicine has great potential to improve access to medication and healthcare services. There are several avenues for future work, and the project team is committed to continuously improving and expanding the application to meet the evolving needs of users and healthcare providers.

### 6.3 REFERENCES

"A systematic review of mobile health applications for chronic disease management" by L. Free, G. Phillips, F. Galli, et al.

"Design and Implementation of a Mobile Electronic Health Record Application for Antenatal Care in Rural Kenya" by P. Braitstein, J. Einterz, P. Sidle, et al.

"mHealth: A global vision for the use of mobiles in healthcare" by World Health Organization

"Mobile Health Technology for Improving Symptom Management in Low Income and Underserved Patients with Breast Cancer: A Systematic Review and Meta-Analysis" by S. Sajeed, S. Cooper, A. Verma, et al.

"The use of mobile devices for medical applications" by S. Lee, M. Lee, and Y. Koo.

Flutter: A Framework for Mobile App Development. (n.d.). Retrieved from <https://flutter.dev/>

Payment Gateway Integration. (n.d.). Retrieved from <https://www.paypal.com/us/business/accept-payments/online-checkout>

Pharmaceutical Industry Overview. (n.d.). Retrieved from <https://www.ibisworld.com/united-states/pharmaceutical-manufacturing-industry/market-research-report/>

Pharmacy Inventory Management. (n.d.). Retrieved from <https://www.omnicell.com/solutions/pharmacy-inventory-management>

# PART 2: SOFTWARE REQUIREMENTS SPECIFICATION

## 1. INTRODUCTION

### 1.1. Purpose

The purpose of this document is to present a detailed description of the pharmacy app system. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli.

### 1.2. Scope

This software system will be a pharmacy app for users to order and purchase medicine from local pharmacies. The app will be designed to maximize efficiency and be of great convenience for all the user groups. The system contains a relational database with the following entities: Users, Pharmacy, Products, Orders, Payment, and Delivery.

### 1.3. Intended Audience and Reading Suggestions

The intended audience for this project are:

* Product Owners
* Pharmacy owners/managers
* Pharmacy cooperatives
* Pharmacy customers
* Investors
* Business analysts
* Product developers
* Front-end developers
* Back-end python developers
* Database designers and engineers
* Product testers
* Project supervisors

The intended use for this document is as follows:

* Designing and brainstorming new features
* Planning project duration, sprints, estimating costs
* Evaluating risks
* Monitoring and measuring the team’s success
* Conflicting situations when involved parties have different visions of a well-executed product.

### 1.4. Definitions and Acronyms

|  |  |
| --- | --- |
| **Term** | **Definition** |
| User | A general term for anybody who interacted with the system, whichever the level (e.g. Customer, Admin, Admin) |
| Sudo Admin/ Superuser | System administrator who manages and controls the system. The superuser has access to all the CRUD operations (e.g. system admin) |
| CRUD | Abbreviation for CREATE, READ/RETRIEVE, UPDATE, DELETE operations |
| API | Abbreviation for Applications Programming Interface, that acts as a middleman in the communication between the server and the app/dashboard |

## 2. OVERALL DESCRIPTION

### 2.1. Product Perspective

The product is not new as there are already some others in the market with the same aim but with some limitations that the product is looking to overcome.

### 2.2. Product Functions

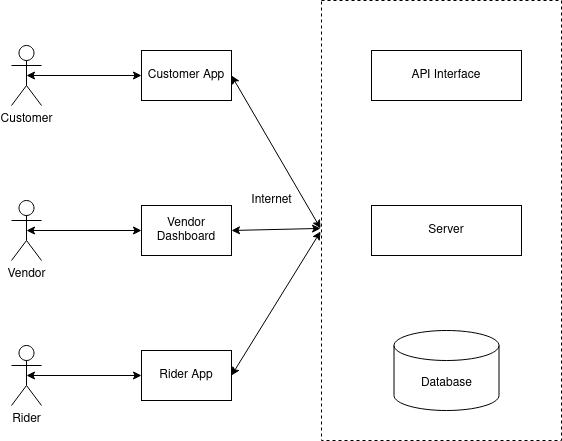
● Create and manage user accounts

● Create, update, delete products

● Bill on checkout

● Notify users of important actions

The following diagram summarizes the flow of data:



### 2.3. User classes and objectives

Customers

● Create account

● Browse available products

● Add items to cart

● Checkout

● Update delivery locations

● Update profile

Admin

● View sales summary

● Manage orders

● Dispatch orders

Rider

● Receive shipment request

● Confirm delivery

Admin

● Perform super user roles

### 2.4. Operating environment

The server will be deployed on a Linux operating system. The web apps can be accessed through any browser on a PC, Mac, and mobile devices. The Android app will be accessible via Android phones and can also be compiled to work on iOS devices as the development technology used is platform-independent.

### 2.5. User documentation

The customer accesses the system through the customer app. The app utilizes the interface that enables the user to perform their specific roles from account creation to order checkout. The pharmaceutical company manager or chemist accesses the system through the web interface. Here they have the system overview of all the roles and actions they can perform. The rider will have a separate app to enable them to view placement requests involving them and the delivery details as well. The admin also has their own separate dashboard for their specific roles. The users will have a manual for use on their specific applications.

### 2.6. Assumptions & dependencies

Some of the system's functions need other requirements to perform. These include:

The users will have a stable internet connection.

The customers will have a functional Android phone.

The customer order location is the delivery location.

The admins have an operational computer or laptop for managing their dashboard.

## 3. SYSTEM FEATURES

This section describes specific product functionality and its execution criteria. It gives a more in-depth description with detailed explanations.

### 3.1. Functional requirements

#### 1. User Accounts

1.1. The system shall enable the user to create an account.

1.2. The system shall authenticate users using their login credentials.

1.3. The system shall enable users to sign out and clear saved details.

1.4. The system shall warn users in case of input errors.

#### 2. Products Listing

2.1. The system shall display all available products.

2.2. The system shall filter admin-specific products.

2.3. The system shall allow users to view product details.

#### 3. Product Filtration

3.1. The system shall allow the user to filter products by admin.

3.2. The system shall allow the user to filter products by product name.

3.3. The system shall display relevant information when filters are not found.

#### 4. Shopping cart

4.1. The system shall enable users to add products to the cart.

4.2. The system shall allow users to edit cart items.

4.3. The system shall enable users to clear cart items.

4.4. The system shall enable users to checkout.

4.5. The system shall enable the user to process payments via M-Pesa.

#### 5. Location update

5.1. The system shall enable users to pick their default delivery location.

5.2. The system shall enable users to search their location by name.

#### 6. Sales overview

6.1. The system shall summarize admin sales and provide graphs.

6.2. The system shall provide sales listing status such as pending, on transit, complete or canceled.

6.3. The system shall enable order items modification.

#### 7.Email Notification

7.1. The system shall notify users via email when an important action is taken.

#### 8. User support

8.1. The system shall enable users to contact for help.

#### 9. Order Shipment

9.1. The system shall avail shipment info to the rider.

9.2. The system shall enable riders to update shipment status.

9.3. The system shall enable riders to confirm product delivery.

#### 10. User feedback

10.1. The system shall enable users to give their feedback.

10.2. The system shall enable the user to rate the service.

#### 11. Reports

11.1. The system shall provide admins with sales report.

11.2. The system shall provide admin with operations details.

### 3.2. Non-functional requirements

#### 1. Usability

1.1. The system shall be easy to use and navigate for all types of users.

1.2. The system shall provide clear and concise instructions on how to use the different features.

#### 2. Performance

2.1. The system shall have a fast response time when loading and processing data.

2.2. The system shall be able to handle a large number of users at the same time without any performance issues.

2.3. The system shall have an uptime of at least 99%.

#### 3. Security

3.1. The system shall use encryption to protect user data.

3.2. The system shall have user authentication to prevent unauthorized access.

3.3. The system shall have backup and recovery procedures to ensure data is not lost in case of a security breach or system failure.

#### 4. Compatibility

4.1. The system shall be compatible with different browsers such as Chrome, Safari, and Firefox.

4.2. The system shall be compatible with different operating systems such as Windows, Linux, and Mac OS.

4.3. The system shall be compatible with different mobile devices such as Android and iOS.

#### 5. Scalability

5.1. The system shall be scalable to accommodate an increasing number of users and transactions.

5.2. The system shall be able to handle a large amount of data without any performance issues.

#### 6. Maintenance

6.1. The system shall have a user-friendly interface for system maintenance and upgrades.

6.2. The system shall have documentation and user manuals for easy maintenance.

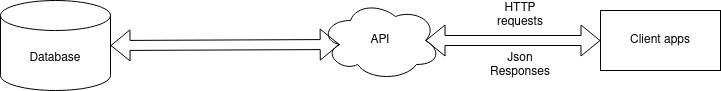
6.3. The system shall have a dedicated support team to help users in case of any issues or problems.

## 4. EXTERNAL INTERFACE REQUIREMENTS

### 4.1. User Interfaces

The pharmacy app shall have a user-friendly interface for all the users, with easy-to-use and intuitive features. The app shall use appropriate icons to support text requirements and improve user experience. The app shall have a navigation menu and toolbars to assist the user move to different pages and take actions. The app shall provide appropriate feedback to the user upon successful or failed actions.

### 4.2. Hardware Interfaces

The pharmacy app will use API to communicate with the server for processing transactions and retrieving data. The API shall be compatible with different hardware devices such as smartphones and tablets running on different operating systems such as Android and iOS as shown below:  


### 4.3. Software Interfaces

The mobile application will work on both Android and iOS devices, while the web interface will be accessible through any web browser on any device.

### 4.4. Communications Interfaces

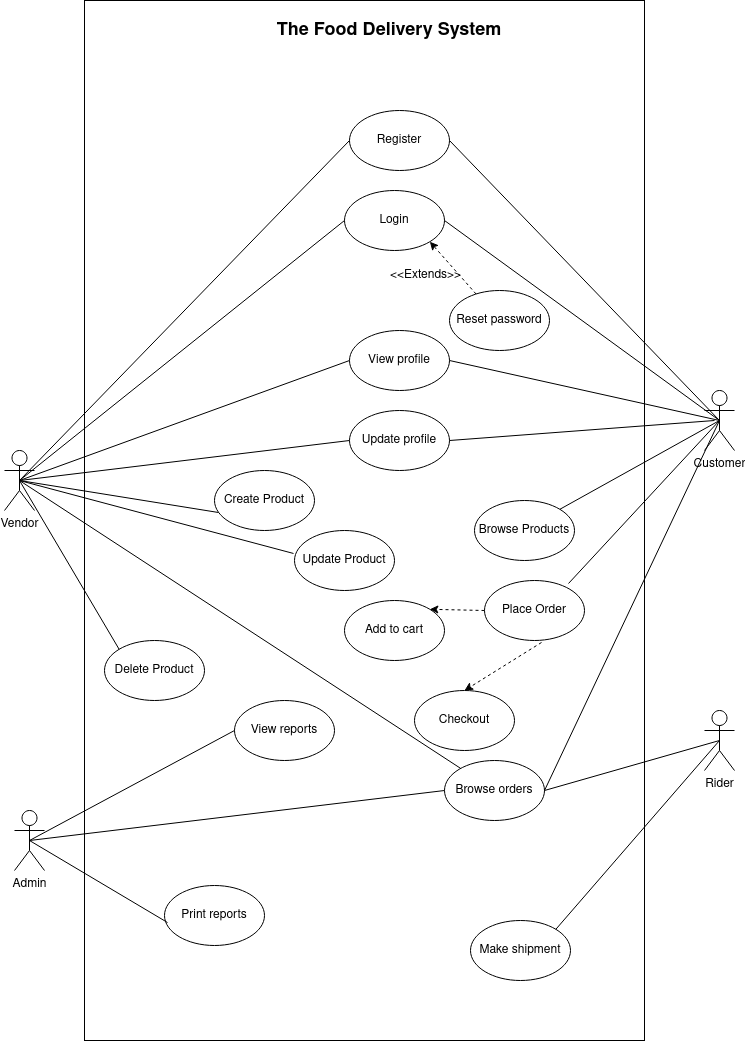
The product will use HTTPS and FTPS protocols to ensure secure data transfer between the server and the client. All requests and responses will be in JSON format.

The Email service to be used in this system is SMTPS to ensure secure transmission of emails.

## 5. PRELIMINARY OBJECT-ORIENTED DOMAIN ANALYSIS

### 5.1. Use case diagram

The use case diagram for the pharmacy app is as follows:



**DRUG STORE MOBILE APP DELIVERY SYSTEM**

## 5.2. Use case template

1. Sign Up Use Case Template

|  |  |
| --- | --- |
| **Use Case Name** | **Sign Up** |
| Actors | User |
| Description | The user creates a new account in the system |
| Pre-Conditions | The user is not already registered in the system |
| Post-Conditions | The user has a new account in the system |
| Normal Flow | 1. User clicks on "Sign Up" button.  2. User enters their personal details (name, email, password).  3. User clicks on "Submit" button.  4. System validates the information entered by the user.  5. System creates a new account for the user and sends a confirmation email.  6. User receives the confirmation email and clicks on the link to activate their account. |
| Alternate Flow | 4a. If the email entered by the user is already in use, the system displays an error message and prompts the user to enter a different email address. |
| Exceptions | None |

1. Login Use Case Template

|  |  |
| --- | --- |
| **Use Case Name** | **Login** |
| Actors | User |
| Description | The user logs in to their account in the system |
| Pre-Conditions | The user has a registered account in the system |
| Post-Conditions | The user is logged in to their account in the system |
| Normal Flow | 1. User clicks on "Login" button.  2. User enters their email and password.  3. User clicks on "Submit" button.  4. System validates the email and password.  5. System redirects the user to their dashboard if the email and password are correct. |
| Alternate Flow | 4a. If the email or password entered by the user is incorrect, the system displays an error message and prompts the user to enter the correct information. |
| Exceptions | None |

1. Password Reset Use Case Template

|  |  |
| --- | --- |
| **Use Case Name** | **Password Reset** |
| Actors | User |
| Description | The user resets their password |
| Pre-Conditions | The user has a registered account in the system |
| Post-Conditions | The user's password is reset |
| Normal Flow | 1. User clicks on "Forgot Password" button.  2. User enters their email.  3. User clicks on "Submit" button.  4. System sends a password reset link to the user's email.  5. User clicks on the password reset link.  6. User enters a new password.  7. User clicks on "Submit" button.  8. System validates the new password.  9. System updates the user's password. |
| Alternate Flow | 4a. If the email entered by the user is not found in the system, the system displays an error message and prompts the user to enter a different email address. |
| Exceptions | None |

1. User dashboard use case template

|  |  |
| --- | --- |
| **Use Case Name** | **User Dashboard** |
| Actors | User |
| Description | The user can access their dashboard to view and manage their profile, view order history, track current orders, and access customer support. |
| Preconditions | The user must be logged in to their account. |
| Postconditions | The user is able to view and manage their account information and order history. |
| Normal Flow | 1. User logs in to their account.  2. User is directed to the dashboard.  3. User can view and manage their profile information.  4. User can view their order history.  5. User can track current orders.  6. User can access customer support. |
| Alternate Flows | None |
| Exception Flows | None |

1. Place order use case template

|  |  |
| --- | --- |
| **Use Case Name** | **Place Order** |
| Actors | User, Pharmacy |
| Description | The user can select and order the desired medication from the pharmacy. |
| Preconditions | The user must be logged in to their account and have access to the pharmacy's available inventory. |
| Postconditions | The order is placed and sent to the pharmacy for processing. |
| Normal Flow | 1. User logs in to their account.  2. User browses the pharmacy's inventory.  3. User selects the desired medication and quantity.  4. User adds medication to the cart.  5. User reviews the cart and proceeds to checkout.  6. User confirms the order and payment information.  7. The order is sent to the pharmacy for processing. |
| Alternate Flows | None |
| Exception Flows | 1. If the user's payment information is invalid, the user is prompted to correct the information and resubmit the order. |

# PART 3: SOFTWARE DESIGN DOCUMENT

## 1. INTRODUCTION

### 1.1. Purpose

This document provides a detailed description of the software design, architecture, and functionality of a pharmacy app. Its purpose is to aid in decision-making, planning, analysis, and implementation of the software.

### 1.2. Scope

This document describes the design of the pharmacy app, including its overall system architecture, component subsystems, and modules. UML diagrams will be used to visually present the system's design and aid developers in understanding the information presented in this document.

### 1.3. Project Executive Summary

The pharmacy app is designed to provide an easy-to-use interface for customers to purchase pharmaceutical products and for pharmacists to manage their inventory and customer orders. Customers can browse products, add them to their cart, and checkout using a secure payment system. Pharmacists can manage their inventory, view and process customer orders, and receive notifications when new orders are placed. The app will improve the efficiency of the pharmacy and make purchasing pharmaceutical products more convenient for customers.

### 1.4. Points of Contact

The following are the contact details for the document author and the project developer.

* Name: Dennis Mbatia
* Email: [mathengedennis115@gmail.com](mailto:mathengedennis115@gmail.com)

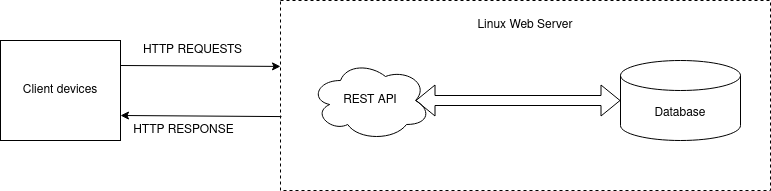
## 2. SYSTEM ARCHITECTURE

### 2.1. Hardware

The hardware component comprises of three major layers. These are:

* The database storage
* The web server
* The client devices

Below illustrates the hardware architecture:



### 2.2.Software

The software layer also comprises of three major sections mainly:

* Front-end: Flutter
* Back-end: React JavaScript Framework
* Database: Firebase

The software modules used in this system application is as illustrated below:

Android App

iOS App

Firebase Firestore API

FIREBASE

Data layer

Service layer

FIREBASE functions

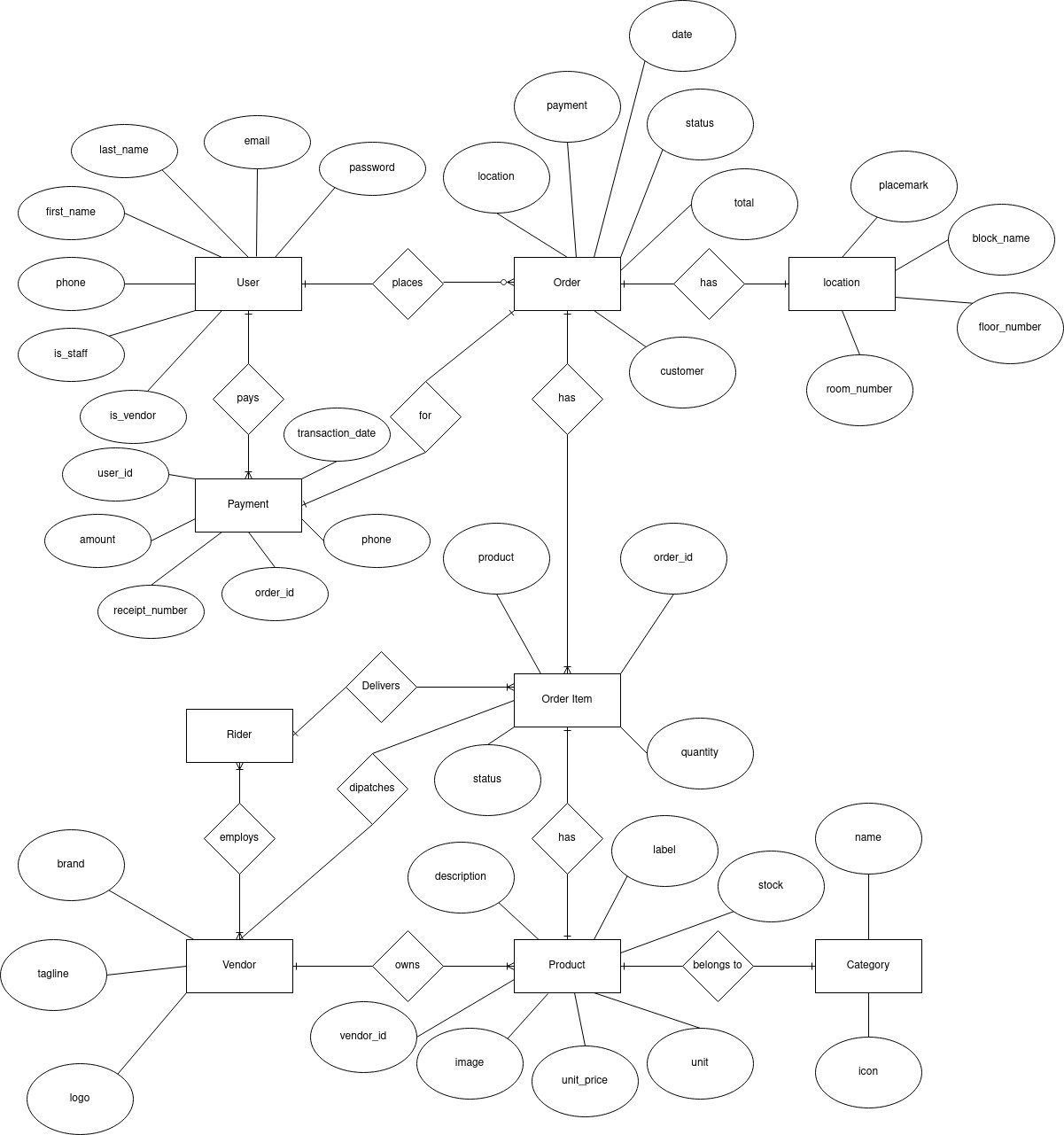
HTTP

Protocols

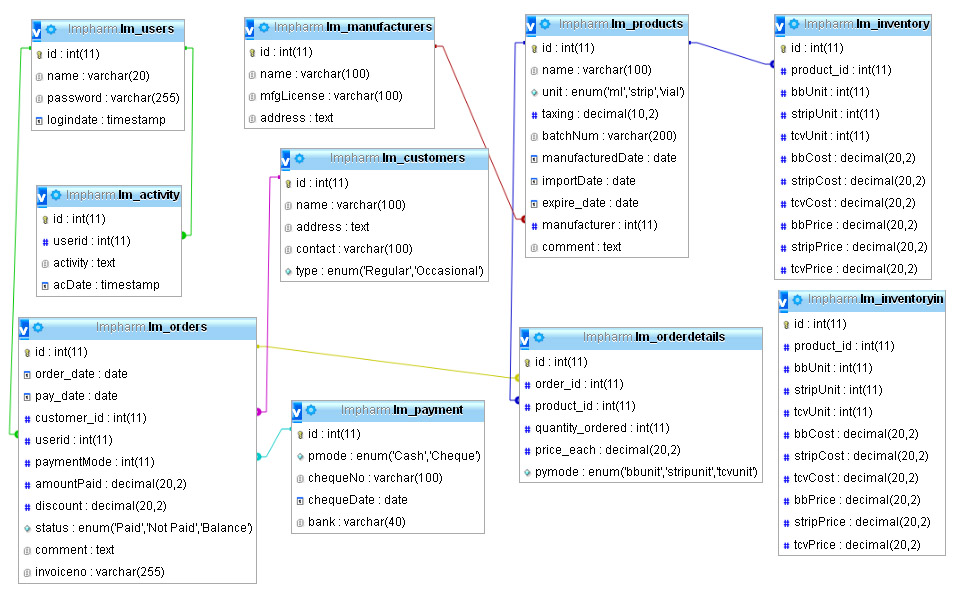
Client Layer

## 3. FILE AND DATABASE DESIGN

### Entity relations diagram



### Data schema



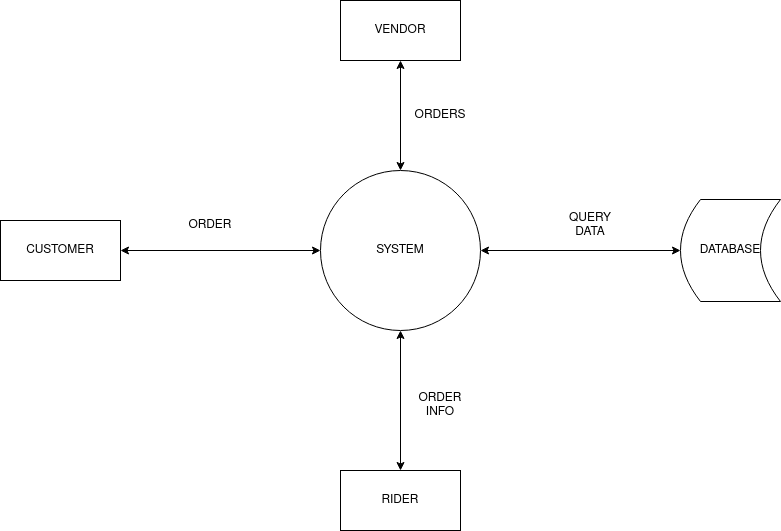
## 4. HUMAN MACHINE INTERFACE

### 4.1. Data flow diagrams

Data flow diagrams show data flow through the systems generated by entities and processed.

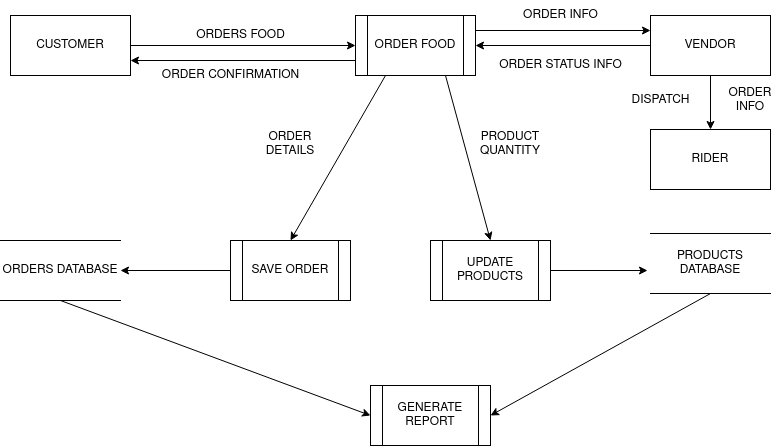
#### Level 0 DFD

This generally illustrates the inputs and outputs of data through the system.



#### Level 1 DFD

This delves a little deeper to show the actual processes involved, their inputs and the outputs they return.



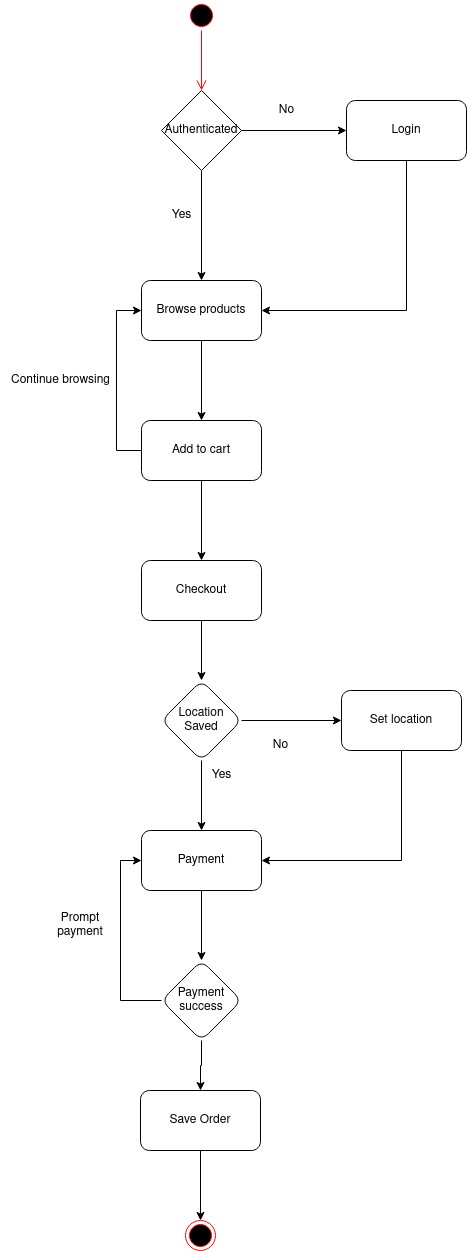
### 4.2. Activity diagram

#### 4.2.1. Authentication

This shows the flow of actions taken by the user when logging into the pharmacy app and the system's response to those actions.

#### 4.2.2. Medication ordering

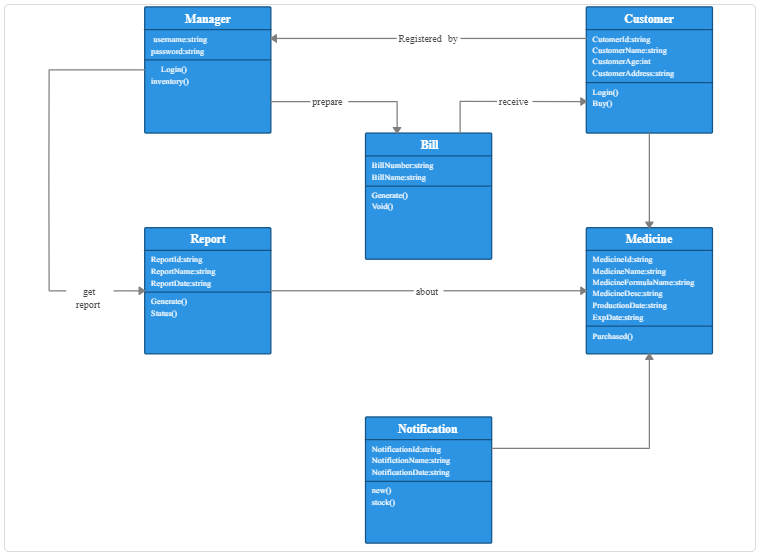
This diagram shows the flow of actions taken by the user when ordering medication through the pharmacy app.



## 5. DETAILED DESIGN

### 5.1.Class diagram

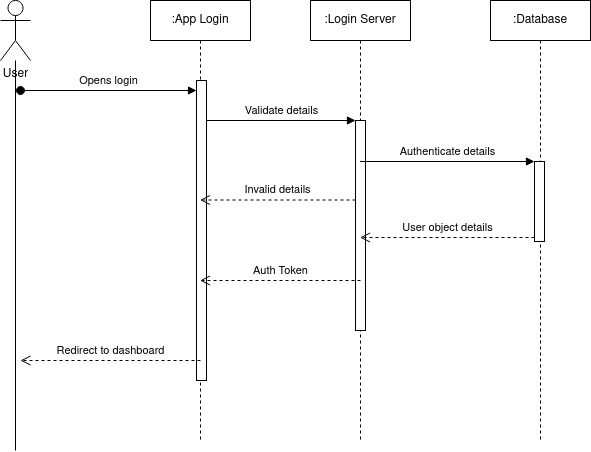
The class diagram demonstrates the system classes, their properties, functions, and connections between objects to define the structure of a project.



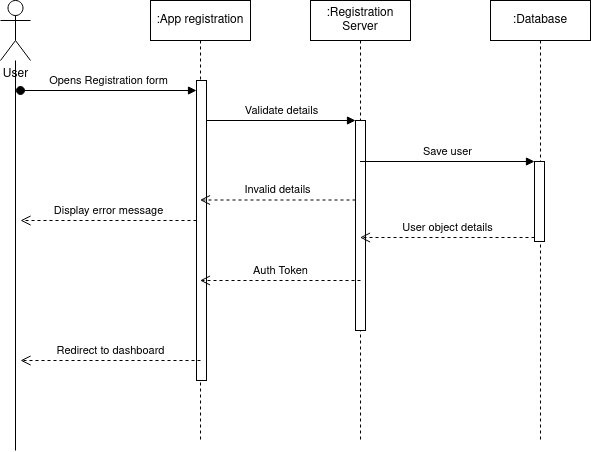
### 5.2. Sequence diagram

The sequence diagram is an interaction diagram that illustrates how actions are taken, including when and how messages are transmitted.

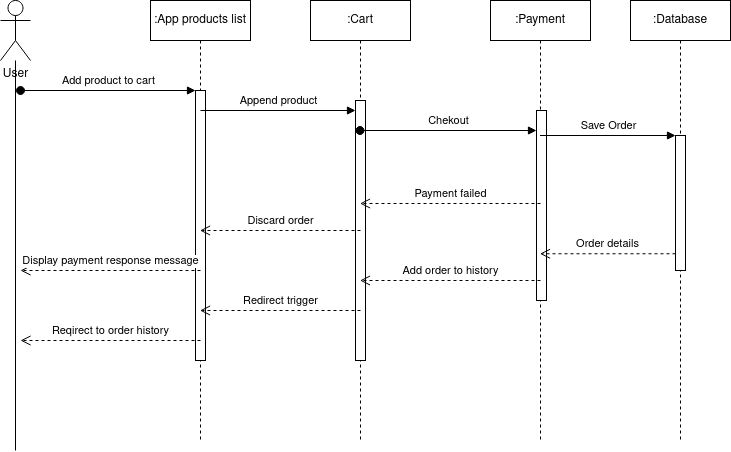
#### 5.2.1 Login sequence



#### 5.2.2 Account Creation sequence



#### 5.2.3 Order sequence



### 5.3 Deployment diagram

The deployment view of a system is represented by a deployment diagram.

This and the component diagram are connected because deployment diagrams are used to deploy the components.

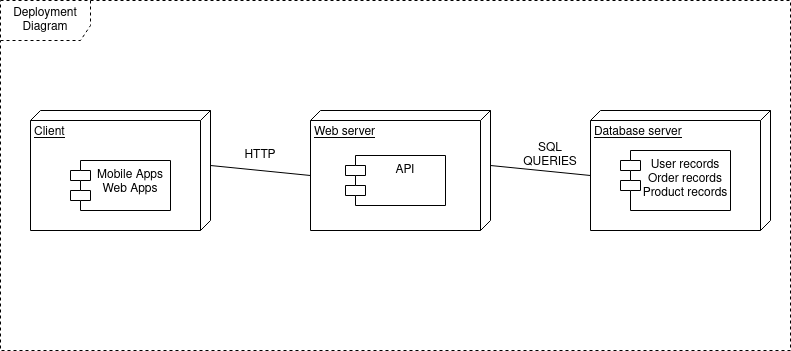
In a deployment diagram, nodes are present.

Nodes are merely the actual pieces of hardware that are used to deliver the program.

System engineers can benefit from deployment diagrams.

Because it regulates the following variables, an effective deployment diagram is crucial.

* Function
* Scalability
* Reliable Maintenance



# PART 4: TEST PLAN

## 1. INTRODUCTION

The goal of this test plan is to ensure that the pharmacy app meets the desired quality standards and is free of defects. The objectives of the test plan include:

* Identifying and eliminating defects in the software
* Ensuring the functionality of the software meets the requirements specified
* Validating the usability and user-friendliness of the software
* Ensuring that the software is compatible with the target environment
* The scope of this test plan is limited to testing the software application on various devices, browsers and operating systems.

## 2. TEST PLAN

The testing strategy for the pharmacy app will include:

* **Unit testing**: Testing individual modules or components of the software in isolation.
* **Integration testing**: Testing the integration between different modules or components.
* **System testing**: Testing the software as a whole to ensure all features and functionalities are working as expected. Interfaces that will be tested include:
* **User interface**: The interface presented to the user for interacting with the software.
* **Database interface**: The interface for managing and storing data.
* **API interface**: The interface for communicating with other systems.

Bug triage process:

* Bugs will be prioritized based on their severity and impact on the system.
* Bugs will be tracked and managed using a bug tracking tool.
* Bugs will be triaged by a designated team member who will assign them to the relevant team members for resolution.

## 3. TEST PROCEDURE

### 3.1. Test case specification

#### Login Test

**Table 3.1.1** Login positive test case

|  |  |
| --- | --- |
| ID | LOGIN\_SUCCESS |
| PRIORITY | High |
| DESCRIPTION | To test is user is able to sign in. |
| REFERENCE | Functional requirements |
| USERS | Customer, Admin and Admin |
| PREREQUISITES | System is online  User had already registered |
| INPUT | Valid email address and password |
| STEPS | Move to login screen  Enter email  Enter password  Click on login button |
| EXPECTED RESULTS | Login success snack message, reroute to dashboard. |
| STATUS | PASSED |

**Table 3.1.2** Login negative test case

|  |  |
| --- | --- |
| ID | LOGIN\_FAILURE |
| PRIORITY | High |
| DESCRIPTION | To test authenticity of the system |
| REFERENCE | Functional requirements |
| USERS | Customer,Admin and Admin |
| PREREQUISITES | System is online  User had already registered |
| INPUT | Incorrect email address or password |
| STEPS | Move to login screen  Enter email  Enter password  Click on login button |
| EXPECTED RESULTS | The system displays error messages |
| STATUS | FAILED |

#### Registration test

**Table 3.1.3** Registration positive test case

|  |  |
| --- | --- |
| ID | REGISTRATION\_SUCCESS |
| PRIORITY | High |
| DESCRIPTION | To test if the user is able to create an account. |
| REFERENCE | Functional requirements |
| USERS | Customer, Admin |
| PREREQUISITES | System is online |
| INPUT | Valid email address, Phone number, password and password confirmation |
| STEPS | Move to registrations screen  Enter email  Enter phone number  Enter password  Click on register button |
| EXPECTED RESULTS | The system successfully registers and authenticates users for that session.  The system mails user as per the email provided.  The system reroutes the user to the dashboard. |
| STATUS | PASSED |

**Table 3.1.3** Registration negative test case

|  |  |
| --- | --- |
| ID | REGISTRATION\_FAILURE |
| PRIORITY | High |
| DESCRIPTION | To test correctness of system forms |
| REFERENCE | Functional requirements |
| USERS | Customer, Admin |
| PREREQUISITES | System is online. |
| INPUT | Enter incorrect input in the input fields. |
| STEPS | Move to register screen  Enter incorrect email  Enter incorrect password  Click on register button |
| EXPECTED RESULTS | The system displays error messages |
| STATUS | FAILED |

#### Product filter test

**Table 3.1.4** Search positive test case

|  |  |
| --- | --- |
| ID | FILTER\_SUCCESS |
| PRIORITY | High |
| DESCRIPTION | To test if user can filter products |
| REFERENCE | Functional requirements |
| USERS | Customer |
| PREREQUISITES | System is online  User authenticated into the system |
| INPUT | Product name |
| STEPS | While on the dashboard  Click on the admin  OR  Go to all medicine list section  Enter medicine name in the search bar |
| EXPECTED RESULTS | The system displays a list of medicine items matching the name or none. |
| STATUS | PASSED |

#### Order placement test

**Table 3.1.5** Order positive test case

|  |  |
| --- | --- |
| ID | ORDER\_SUCCESS |
| PRIORITY | High |
| DESCRIPTION | To test if user can place a medicine order |
| REFERENCE | Functional requirements |
| USERS | Customer |
| PREREQUISITES | System is online  User is authenticated into the system.  User has added items to the cart. |
| INPUT | User delivery address  User payment phone number. |
| STEPS | Navigate to the cart section  Validate order details  Click on the make payment button |
| EXPECTED RESULTS | The system prompts user for MPESA pin through STK push.  Order is saved. |
| STATUS | PASSED |

**Table 3.1.6** Order negative test case

|  |  |
| --- | --- |
| ID | ORDER\_FAILURE |
| PRIORITY | High |
| DESCRIPTION | To test if system if user checkout details are correct |
| REFERENCE | Functional requirements |
| USERS | Customer |
| PREREQUISITES | System is online  User is authenticated |
| INPUT | User delivery address  User payment phone number. |
| STEPS | Go to cart screen  Try checkout without items in cart  Try checkout with no payment number  Try checkout without setting delivery address |
| EXPECTED RESULTS | The system displays error messages |
| STATUS | FAILED |

#### Create product test

**Table 3.1.7** Product create positive test case

|  |  |
| --- | --- |
| ID | PRODUCT\_CREATE\_SUCCESS |
| PRIORITY | High |
| DESCRIPTION | To test if admins can add products to their inventory. |
| REFERENCE | Functional requirements |
| USERS | Admin |
| PREREQUISITES | System is online  The admin is authenticated.  The admin has a shop. |
| INPUT | Product name  Price per unit  Unit quantity  Product Description  Product Category  Product Image |
| STEPS | Click on the add product route  Fill the form fields as labelled  Click on Save product |
| EXPECTED RESULTS | Product is saved and rerouted to the inventory list. |
| STATUS | PASSED |

**Table 3.1.8** Product create negative test case

|  |  |
| --- | --- |
| ID | PRODUCT\_CREATE\_FAILURE |
| PRIORITY | High |
| DESCRIPTION | To test validity of form inputs |
| REFERENCE | Functional requirements |
| USERS | Admin |
| PREREQUISITES | System is online  User had already registered |
| INPUT | Leave one or more fields empty |
| STEPS | Click on the add product route  Fill the form partially  Click on Save product |
| EXPECTED RESULTS | The system displays error messages |
| STATUS | FAILED |

#### Order update

**Table 3.1.9** Product positive test case

|  |  |
| --- | --- |
| ID | ORDER\_UPDATE\_SUCCESS |
| PRIORITY | High |
| DESCRIPTION | Test if admin can modify order status |
| REFERENCE | Functional requirements |
| USERS | Admin |
| PREREQUISITES | System is online  The users are authenticated.  The admin has assigned order to rider |
| INPUT | Admin |
| STEPS | Admin:  Click on order to view details Assign a rider if status is pending Rider:  Click on order to view details Confirm order on delivery. |
| EXPECTED RESULTS | The order status is updated as on Transit and delivered respectively |
| STATUS | PASSED |

#### Report generation

**Table 3.1.10** Product positive test case

|  |  |  |  |
| --- | --- | --- | --- |
| ID | | REPORT\_SUCCESS | |
| PRIORITY | | High | |
| DESCRIPTION | | To test report generation for admins | |
| REFERENCE | | Functional requirements | |
| USERS | | Admin | |
| PREREQUISITES | | System is online  The users are authenticated. | |
| INPUT | | None | |
| STEPS | | On the dashboard analytics  Click on the print report to save the pdf of the sales report. | |
| EXPECTED RESULTS | | Report document downloads | |
| STATUS | | PASSED | |

## 4. TESTING RESOURCE PLANNING

The following resources will be required for testing:

* Hardware: Devices (e.g. smartphones, tablets, laptops) and servers required for testing.
* Software: Test management tool, bug tracking tool, automation tools.
* Human resources: Testers, developers and project managers.

## 5. TEST ENVIRONMENT

The test environment will be set up to mimic the production environment as closely as possible. The following environments will be tested:

* Mobile devices: Android and iOS devices
* Desktop browsers: Chrome, Firefox, Safari and Edge
* Operating systems: Windows, macOS and Linux

## 6. TEST SCHEDULE AND ESTIMATION

The estimated duration for testing the pharmacy app is 4 weeks. The testing schedule will be divided into the following phases:

* Week 1: Unit testing
* Week 2: Integration testing
* Week 3: System testing
* Week 4: Regression testing and bug fixing

During each testing phase, progress and results will be reported to the project manager and development team

# PART5: USER MANUAL

## 1. CLIENT USER MANUAL

This is a user manual for the client's mobile application.

### 1.1. Registration

1. Open the app and click on the "Get Started" button.
2. You will be redirected to the login screen.
3. Click on Registration Link if you don’t have an account
4. Enter your email address in the provided field.
5. Enter your phone number in the provided field.
6. Create a password for your account and confirm it in the provided fields.
7. Click on the "Register" button to complete the registration process.
8. You will be redirected to the Login Screen where you can log into the app and access its features.

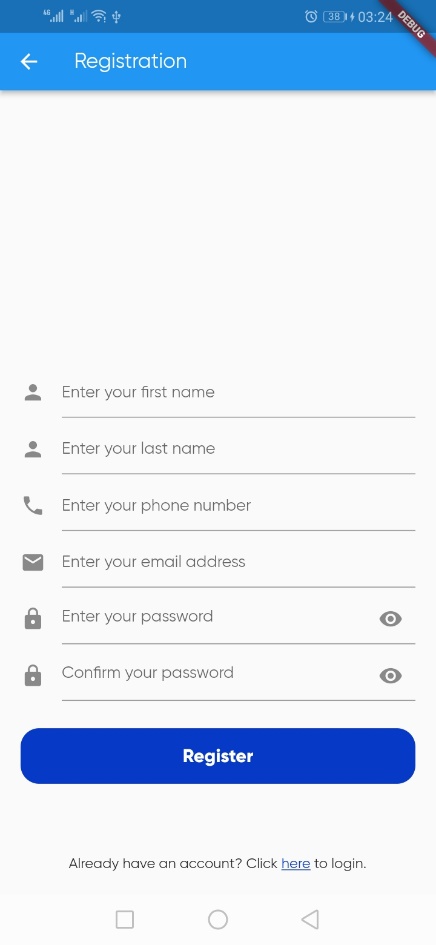
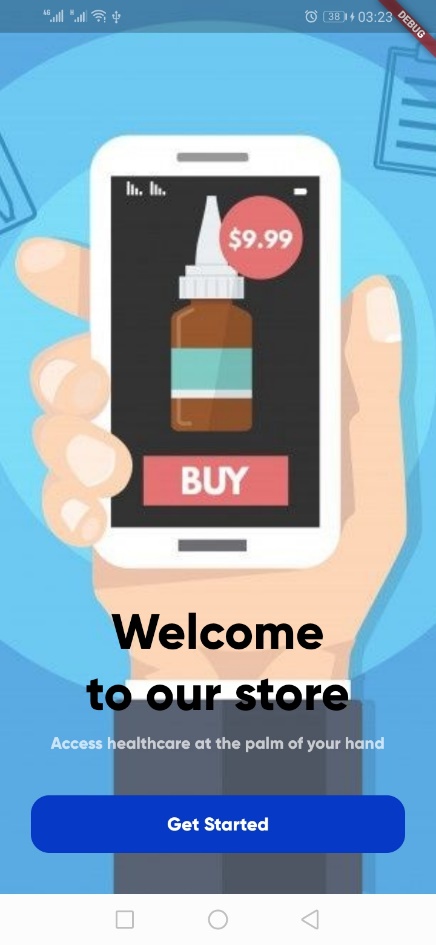


Fig 1.1 Welcome and Register Screen Screenshot

The form will display error messages in case there is any encountered. The user is required to enter the correct email address, Phone number and Strong password.

The password must not be empty and must have at least 8 characters.

### 1.2. Login

1. Open the app and click on the "Get Started" button.
2. You will be redirected to the login screen.
3. Enter your email address in the provided field.
4. Enter password for your account.
5. Click on the "Login" button to complete the login process.
6. You will be logged into the app and can access its features.

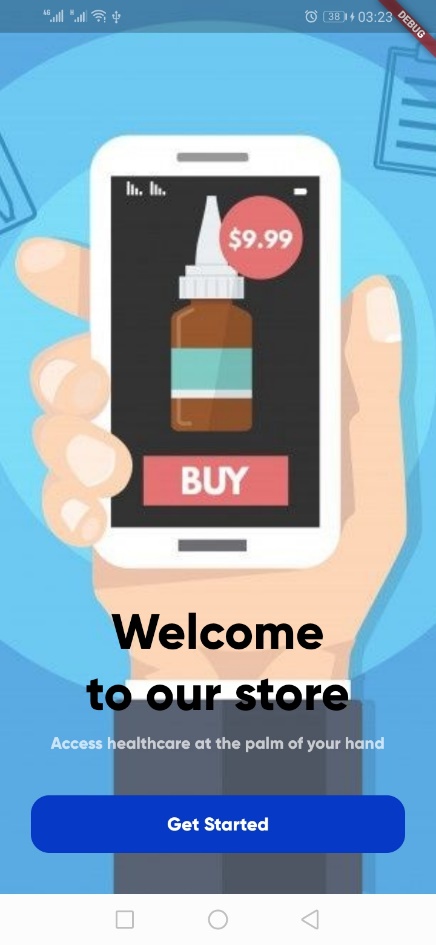


Fig 1.2 Welcome and Login Screen Screenshot

The app will display error or success messages. The error message will be displayed when the form fields are incorrect or empty.

Success messages will be displayed when the form is successfully validated and authenticated.

### 1.3. Add items to cart

1. Once authenticated, the user will be redirected to the dashboard screen.
2. Filter the products by categories
3. Filter products by the list
4. Click on the product card to view product details and add products to cart.

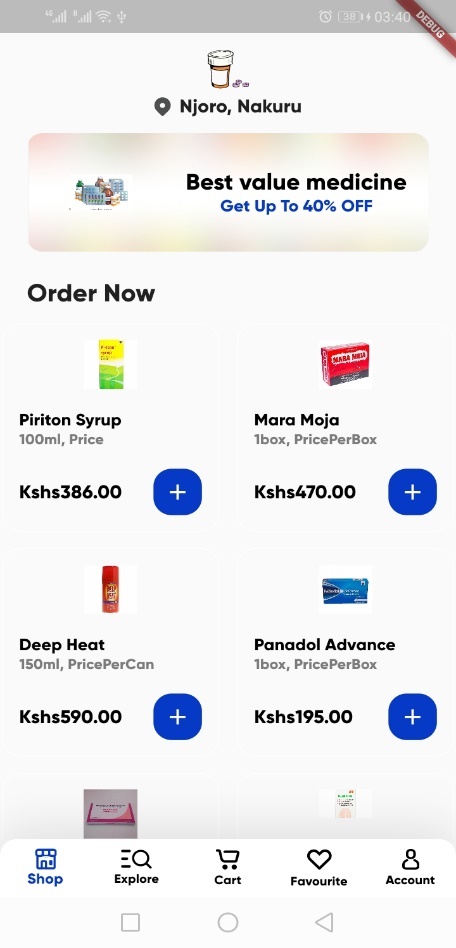


Fig 1.3 Dashboard Screen Screenshot

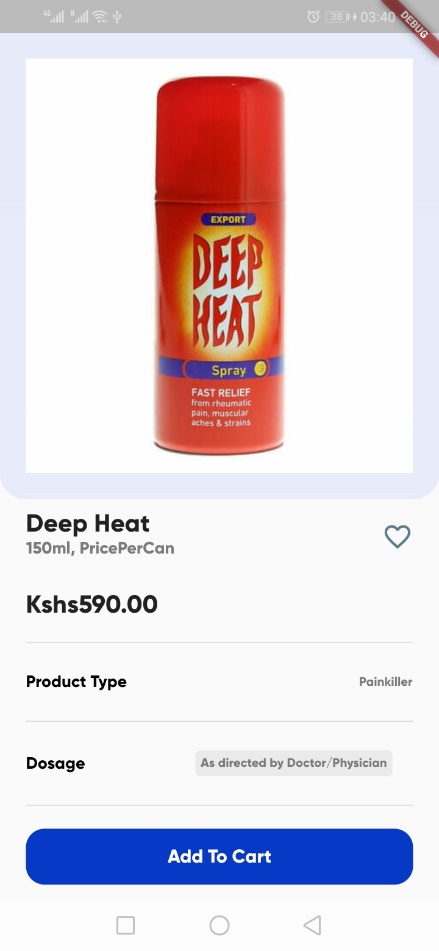


Fig 1.4 Product Details Screen Screenshot

### 1.4. Checkout

1. Once items have been added to cart, you will see an indicator in the bottom cart nav item
2. Click on the Bottom Cart Nav item to go to the cart page.
3. Click on + or - buttons on the cart items to increment or decrement the quantity respectively.
4. Click on the Checkout button update your delivery location and payment phone number
5. Click on the Proceed to Payment button to confirm the purchase and pay via MPESA.

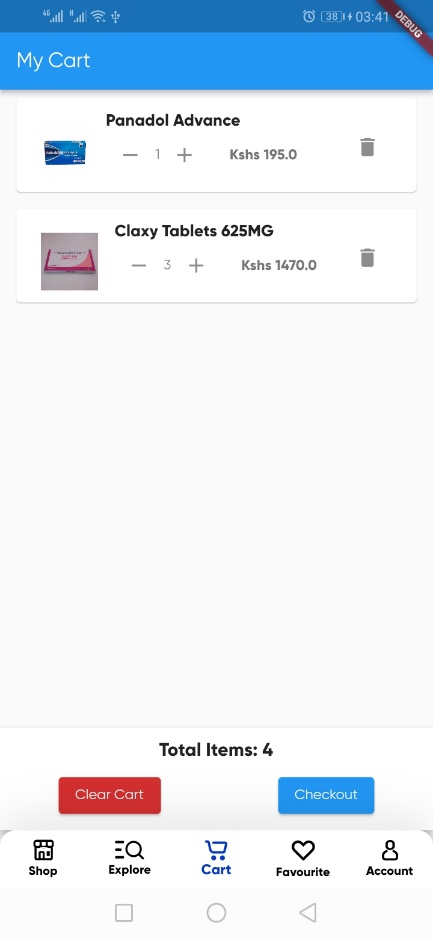


Fig 1.5 Product Details to Cart navigation pages



Fig 1.6 Location Search page

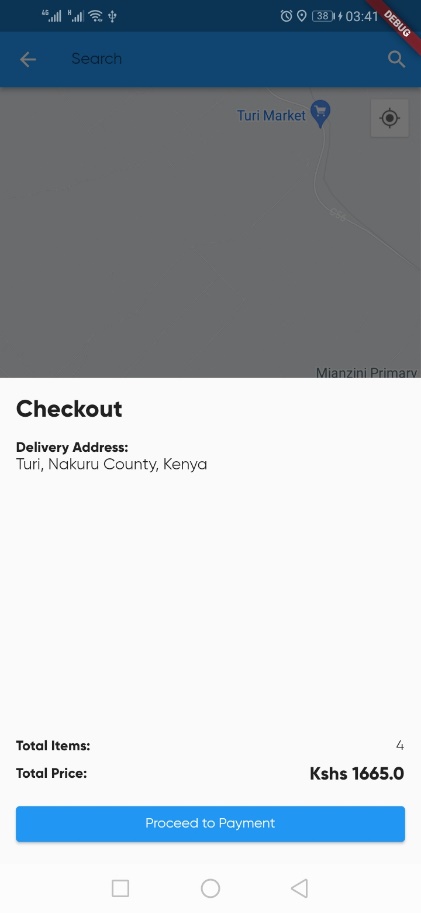


Fig 1.7 Checkout Bottom Sheet with delivery address and Cart Item Details



Fig 1.8 MPESA payment prompt

### 1.5. Profile and history

1. Click on the Profile navigation button at the bottom
2. Click on orders card to view and manage the orders
3. Click on help card to view contact and help details
4. Click on Delivery Access to update default delivery location
5. Click on logout to log out your details.

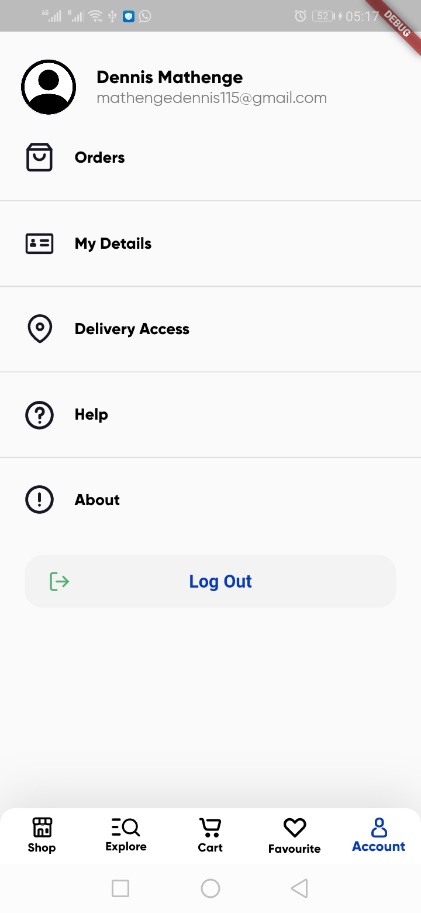


Fig 1.9 Profile page screenshot

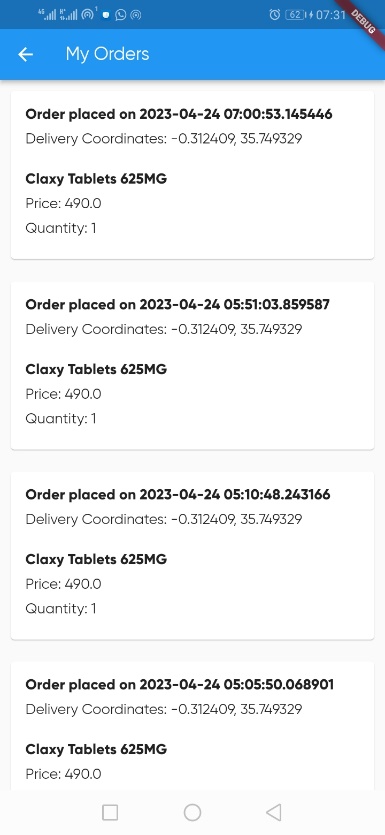


Fig 1.10 Orders Screenshot

It is important to note that new order can be viewed by simply scrolling down on the list page to refresh.

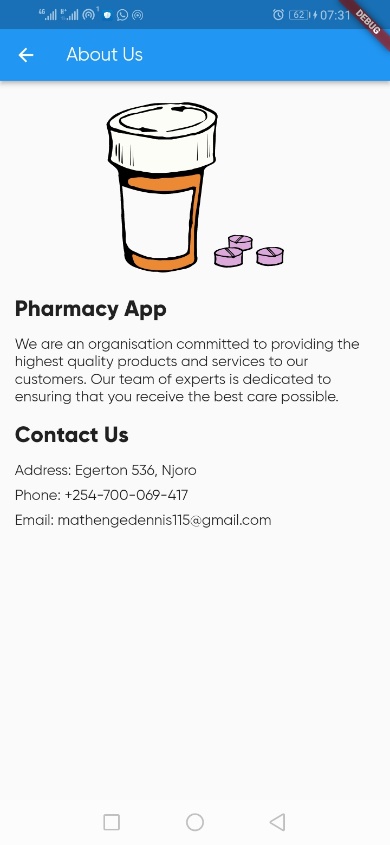
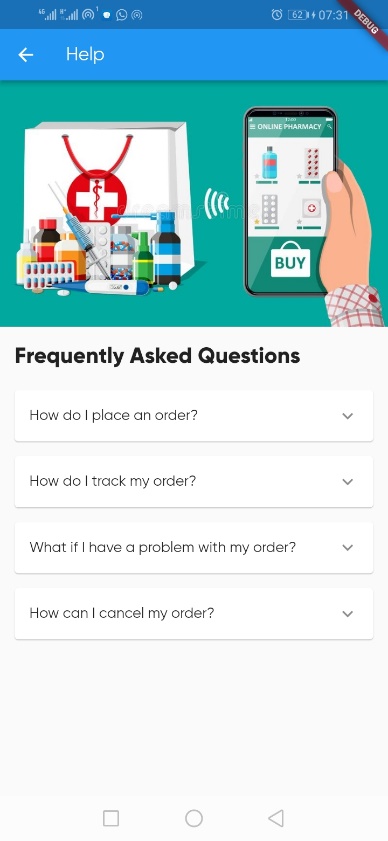


Fig 1.11 Help and contact Details Screen

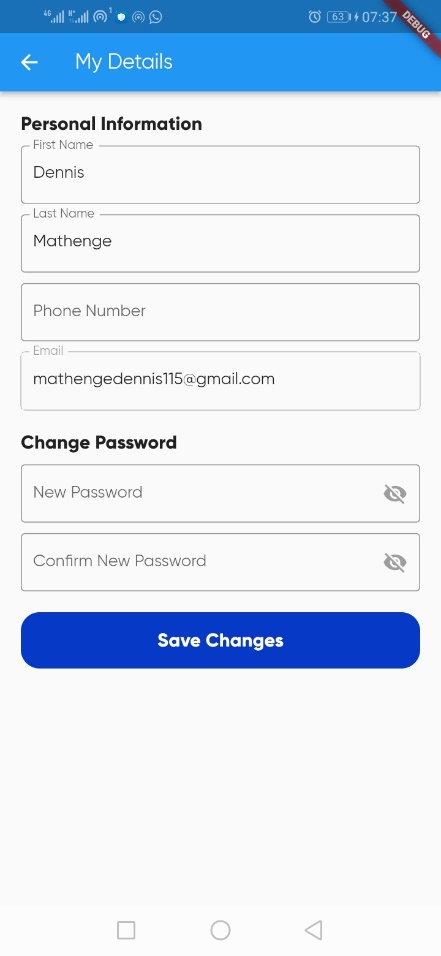


Fig 1.12 Profile Update Screenshot

## 2. ADMIN USER MANUAL

This is a user manual for the admin’s web application.

### 2.1. Registration and login

1. Open the web link and click on the "sign up" text link to navigate to the sign up page
2. You will be redirected to the sign up page you can switch back from the sign in text button to go back and login with your account details.
3. Enter your email address in the provided field.
4. Create a password for your account and confirm it in the provided fields.
5. Click on the "Register" button to complete the registration process.

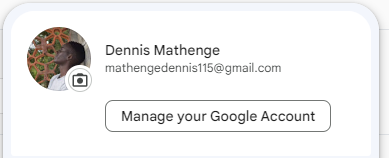


Fig 2.1 Admin sign up screen and login

### 2.2. Admin analytics

1. When successfully authenticated, the admin will be redirected to the analytics dashboard with a side menu on different types of actions.
2. On the top right corner will be the admin profile picture and print report button
3. On the left side panel, there are different navigation links to different screens.
4. The bar chart shows the daily sales totals for the last ten days with the current date being the latest.
5. There are top cards showing summary info as well such as number of new customers, pending orders total, and order stream count.

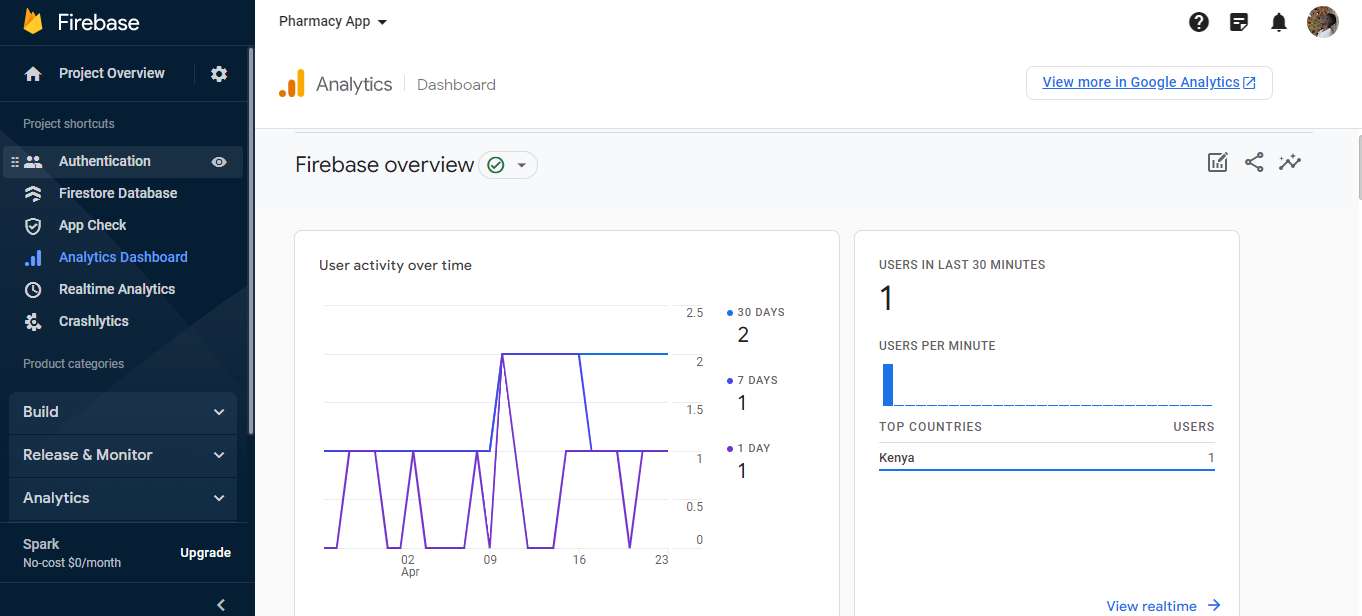


Fig 2.2 Admin analytics dashboard

### 2.3. Admin products and add new product

1. Click on the products navigation link to view your products
2. Each product can be updated by clicking on the pen icon in the product card
3. Click on new product text link to add a new product
4. On the add product form and fill in the required info.
5. Click on save to save your product.

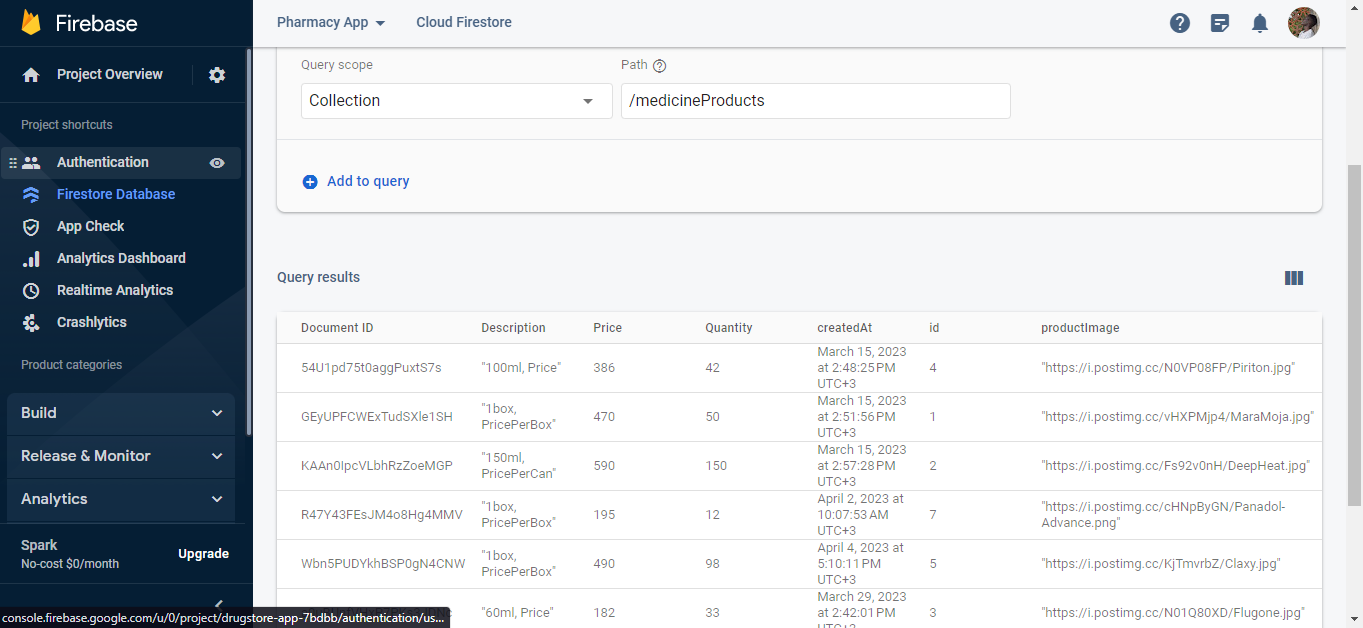


Fig 2.3 Admin Products list

### 2.4. Admin Orders

1. Click on the orders navigation link to view your orders
2. As listed, you can filter orders as per the navigation as pending, on transit, cancelled, and completed.
3. The order action view button reveals more details about the order.
4. The admin can assign a rider to a pending order and dispatch it.

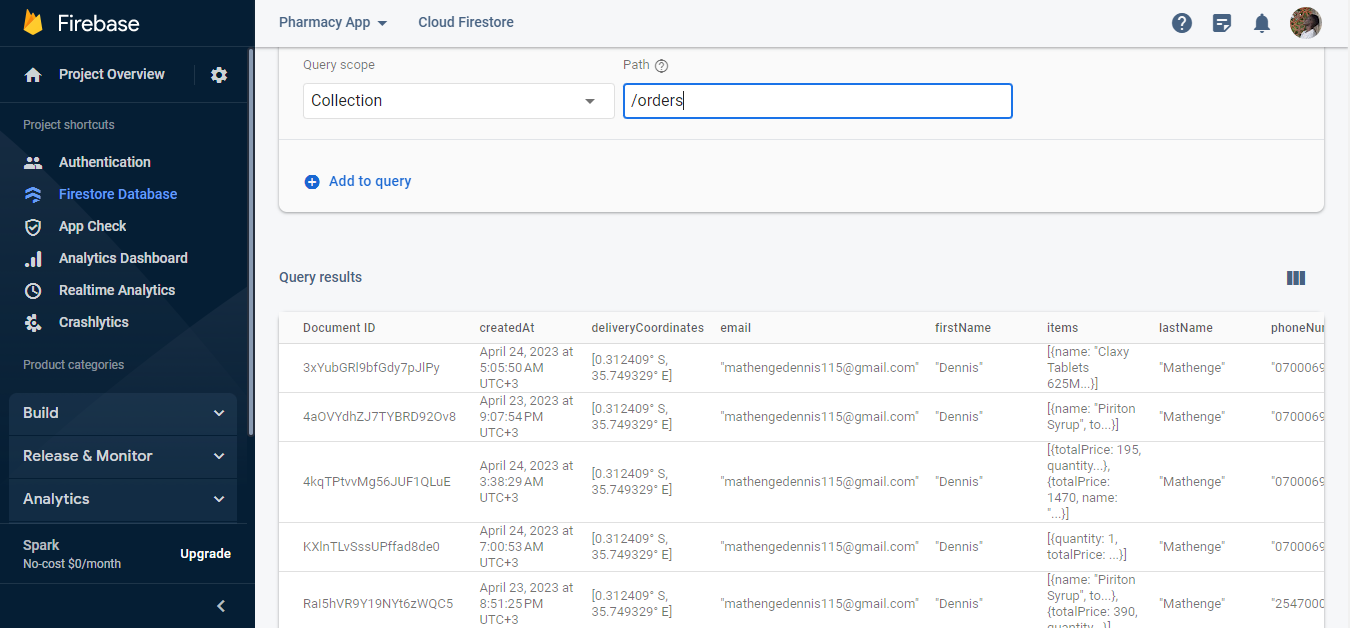


Fig 3.8 Admin orders