# Design Documentation

*Feedback system for Pharmacy Order Management System*

## Introduction

Pharmacy order management system (POMS) that allows pharmaceutical products to consumers buy them according to their preferences through our system. The Company owners want to achieve their goals more efficiently by giving honest feedbacks regarding the product quality, packaging and other facilities provided.

## Scenario

I am developing this online Pharmacy Feedback platform to great service to customers. Due to the Current pandemic situation of COVID-19 the customers couldn't get their prescription and medication equipment's. The new solution should provide company with a set of features to place new products, packaging and delivery methods in the online platform, monitor the number and state of all current orders and improve customer satisfaction. Being aware that provides Java and MySQL development services.

## Solution

The system has substantial expertise in providing custom software development services which allows to create high-quality order management systems for the customers. The management system transfers all received orders to an operator to be processed. The owner can view pharmaceutical items, it helps to control the number of goods in stock, and enables customers' feedback processing.

This software allows users to receive notifications and the goal is to get a feedback from the user. The management system has some pre-defined reporting templates or users can write a Feedbacks manually. This Feedback System is to extract data about customers' experiences and find out what they expect from you. Pharmacy feedback system that serves to bridge the gap between you and your customer's perceived value of your service.

Then the customer can give us feedback by rating us by giving stars and, write some note about quality of product, if there are any problems with the order, what about our packaging and the user is satisfied with our delivery methods are etc.

## Overview of the Design

The system will be designed using Java as the main language. Furthermore, MySQL will be used as the database platform. Since we are designing a desktop application, Eclipse is the recommended IDE. RMI server will be used as the framework to interact with the client GUIs.

## The Scope of the Job

Nexas Pharmacy Feedback system will be designed to attract more consumers. To develop the business, the most important factor is to maintain quality control. The main functionality of this system is a feedback framework to obtain realistic opinions from consumers. This will help the business reduce unsatisfied consumers and increase sales on the products they prefer the most.

## System Design

The proposed system of Nexas Pharmacy Feedback System is an application written in JAVA and SQL. This system is consisting of APIs like RMI, QuickChart.io and Javamail. The RMI (Remote Method Invocation) is an API that provides a mechanism to create a distributed application in java. The RMI allows an object to invoke methods on an object running in another JVM. QuickChart.io is an API that provides a mechanism to create a programming language Java, which allows the creation of a wide variety of both interactive and non-interactive charts NET. Java Mail is a Jakarta EE API used to send and receive email via SMTP, POP3, and IMAP. Java Mail is built into the Java EE platform, but also provides an optional package for use in Java SE.

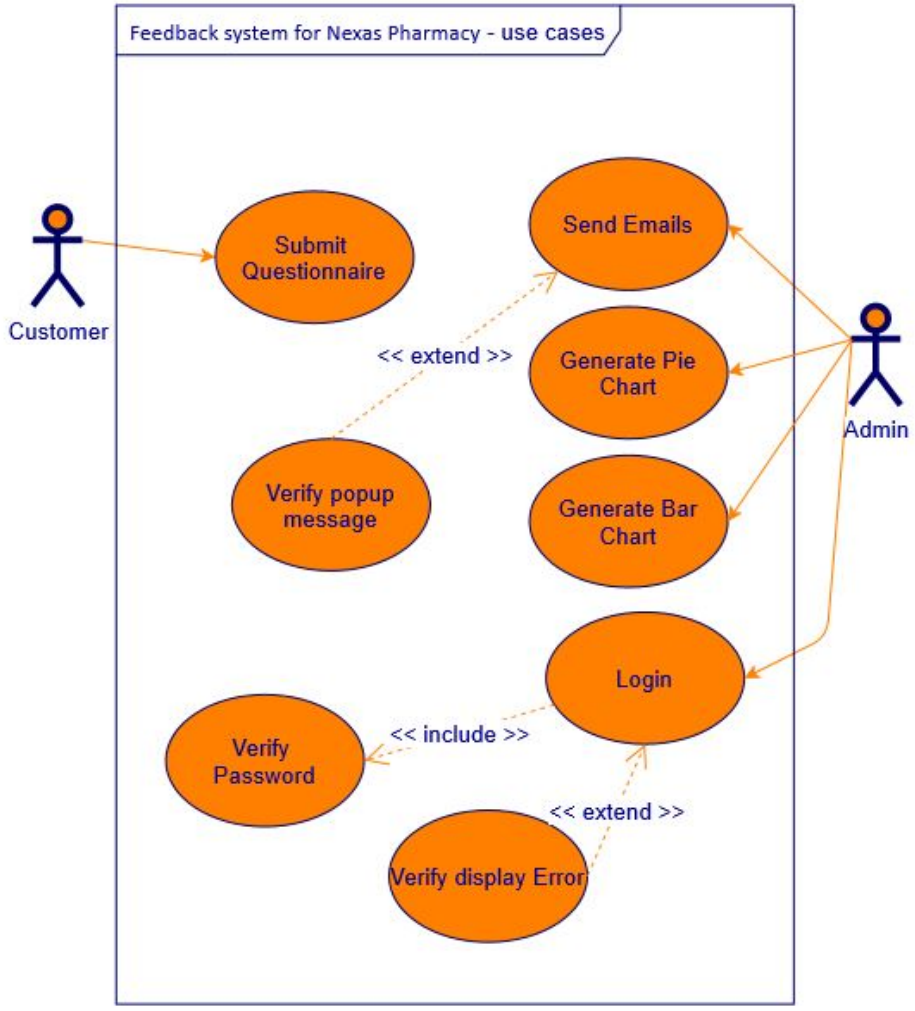


*Figure1: Use Case Diagram*

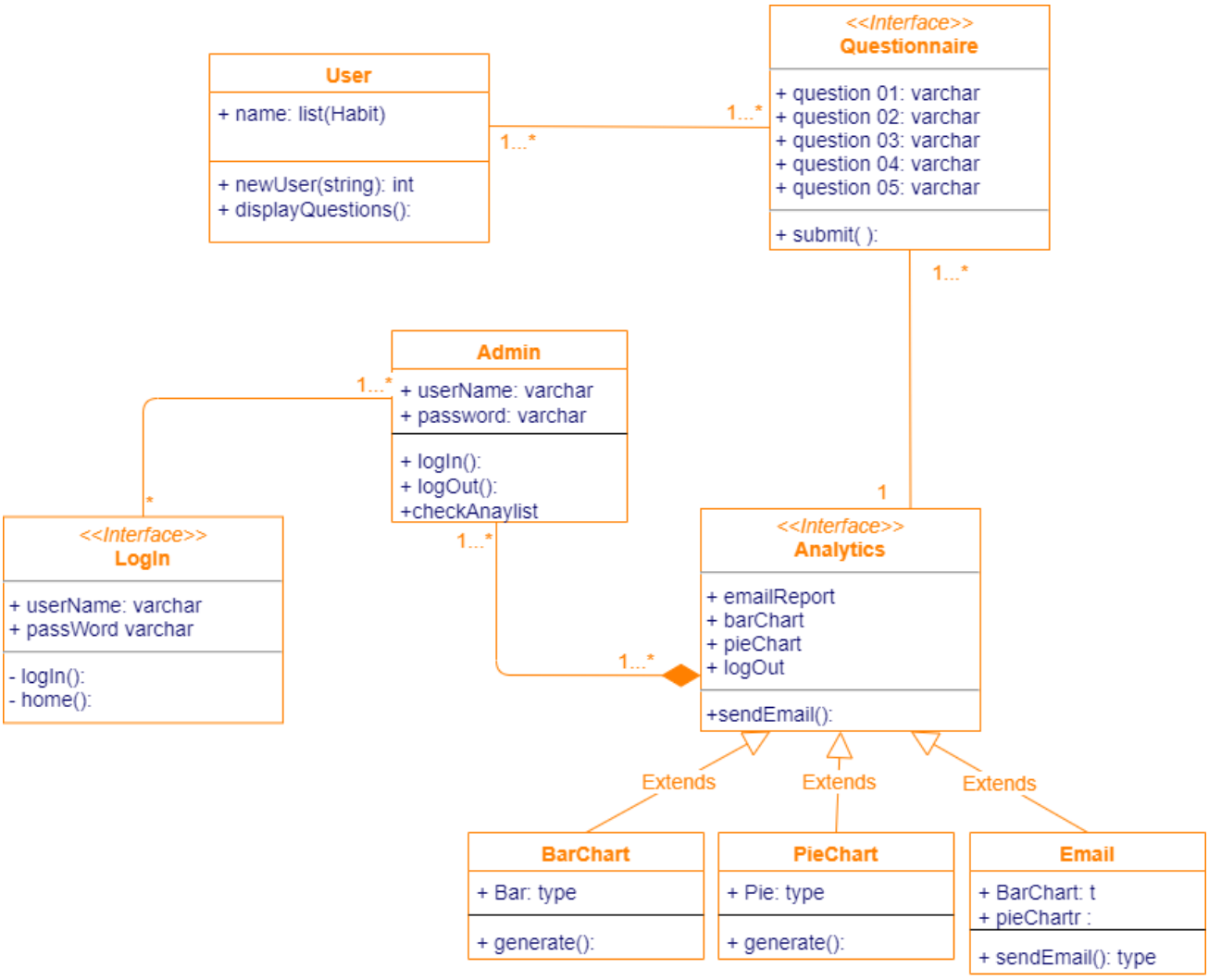


*Figure 2: Class Diagram*

# Activity Diagran - Feedback system for Nexas Pharmacy

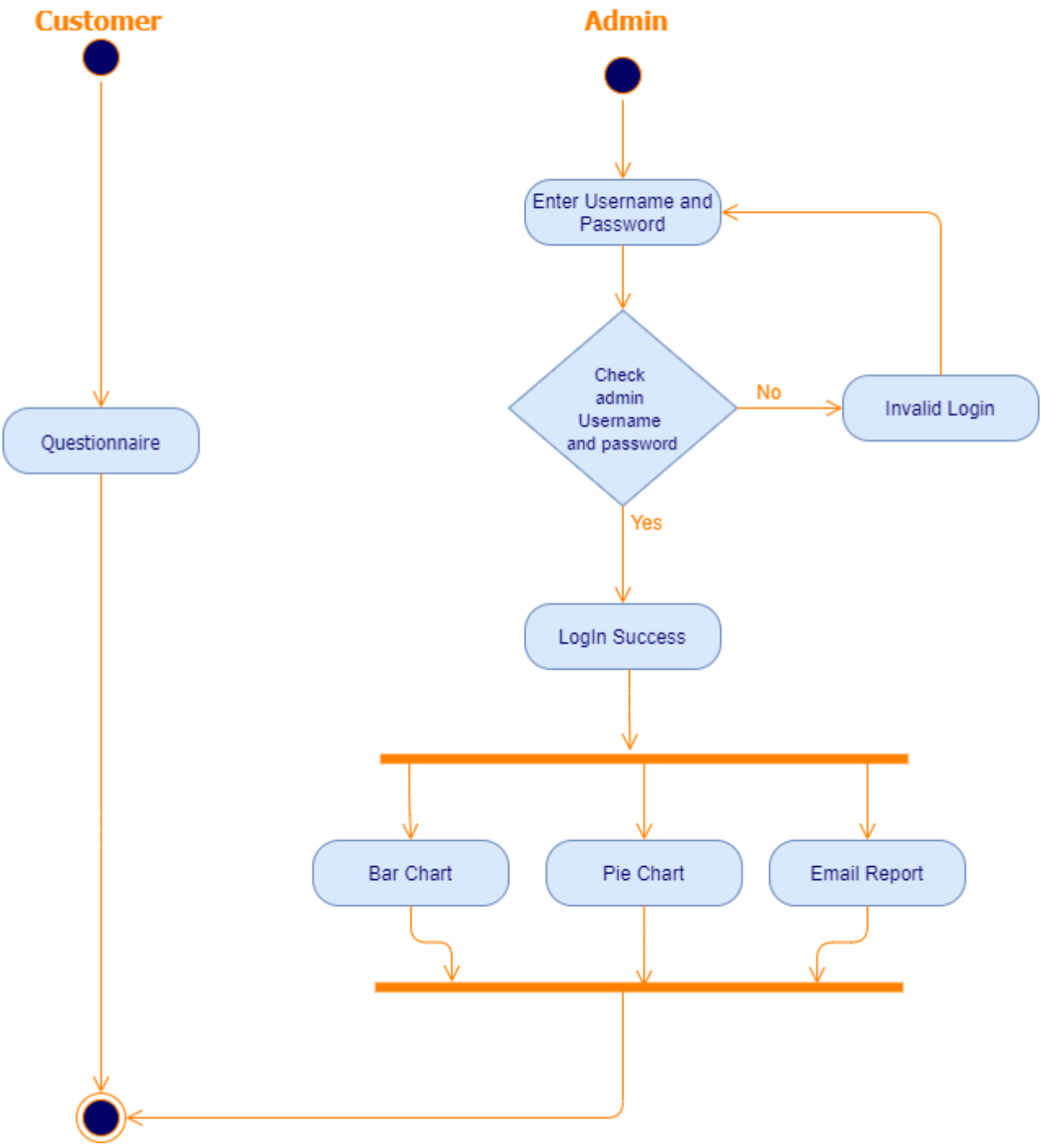


*Figure 3: Activity Diagram*

## ER Diagram - Feedback system for Nexas Pharmacy

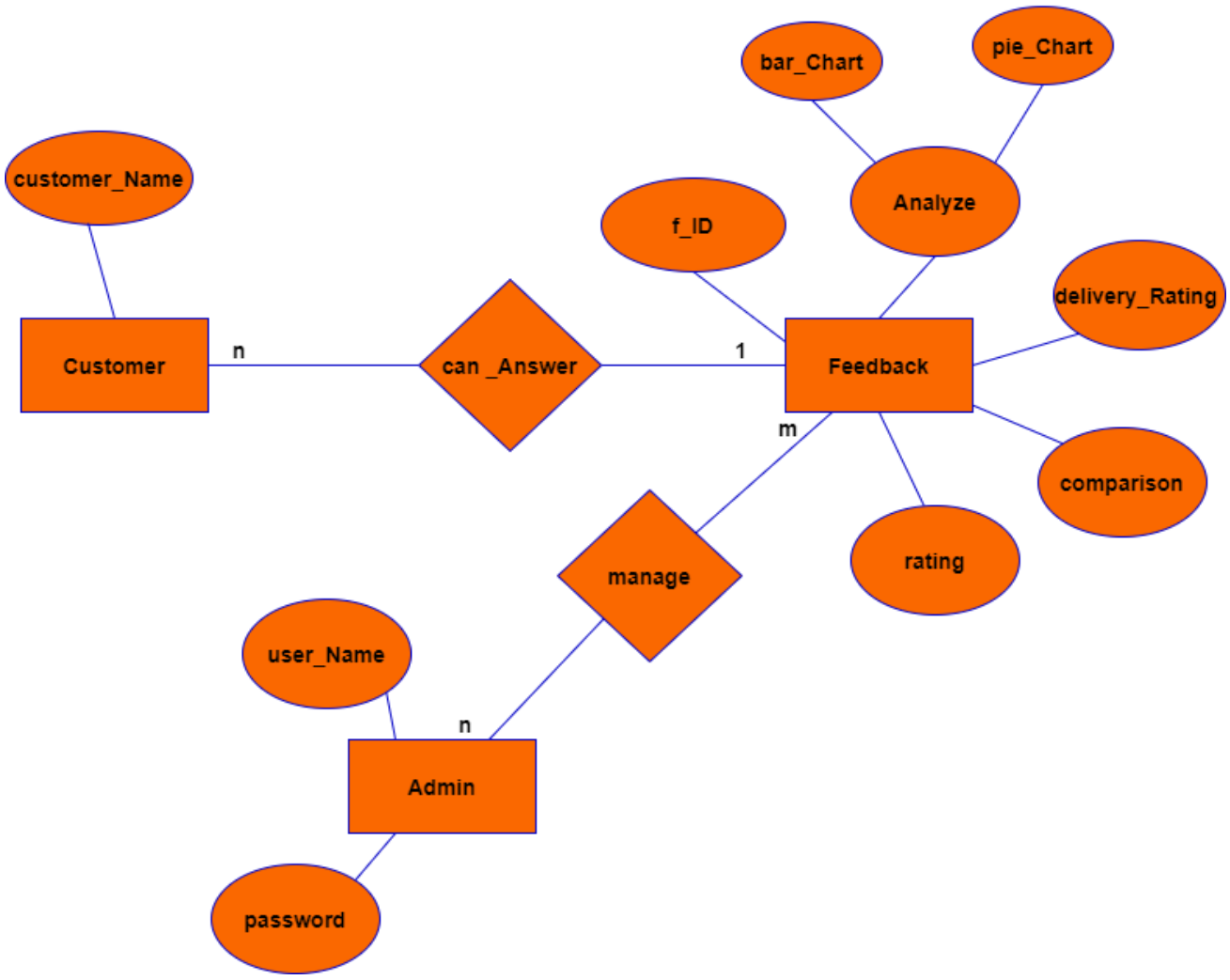


*Figure 4: ER Diagram*

## Design Details and Interface

The interfaces of the system are attached below to provide the picture and the scope of the system.

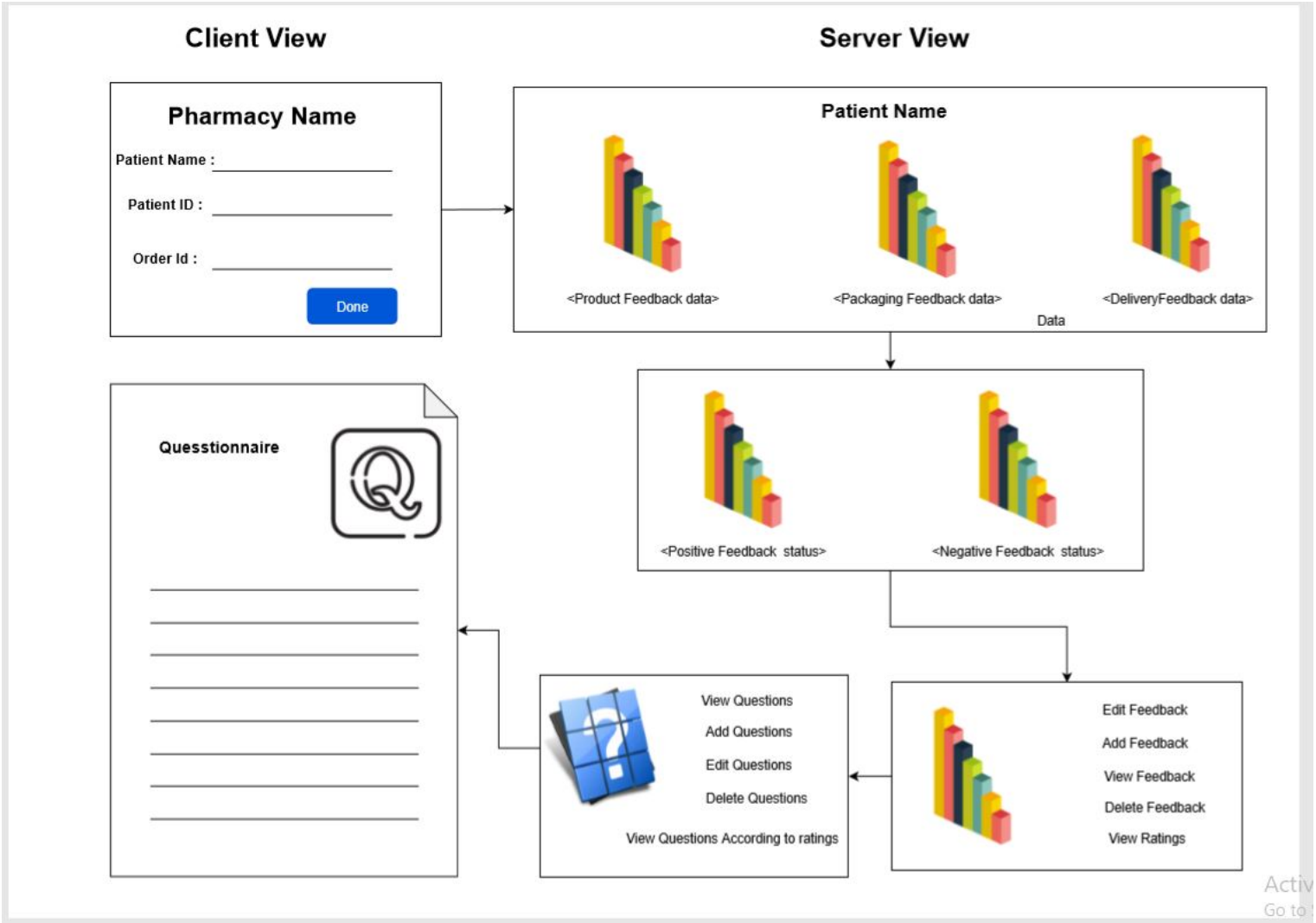


*Figure 5: GUI Mock Ups*

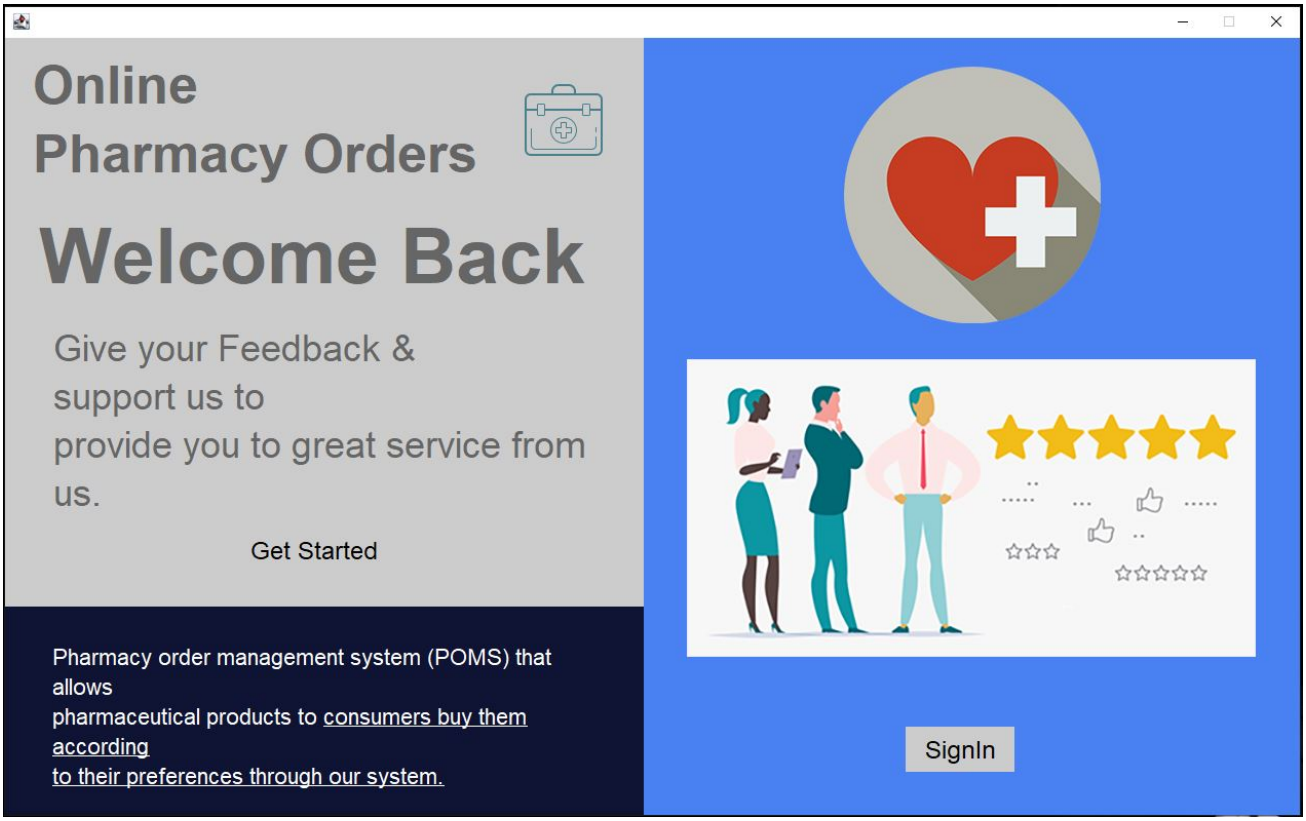


*Figure 6: Dashboard Screen*

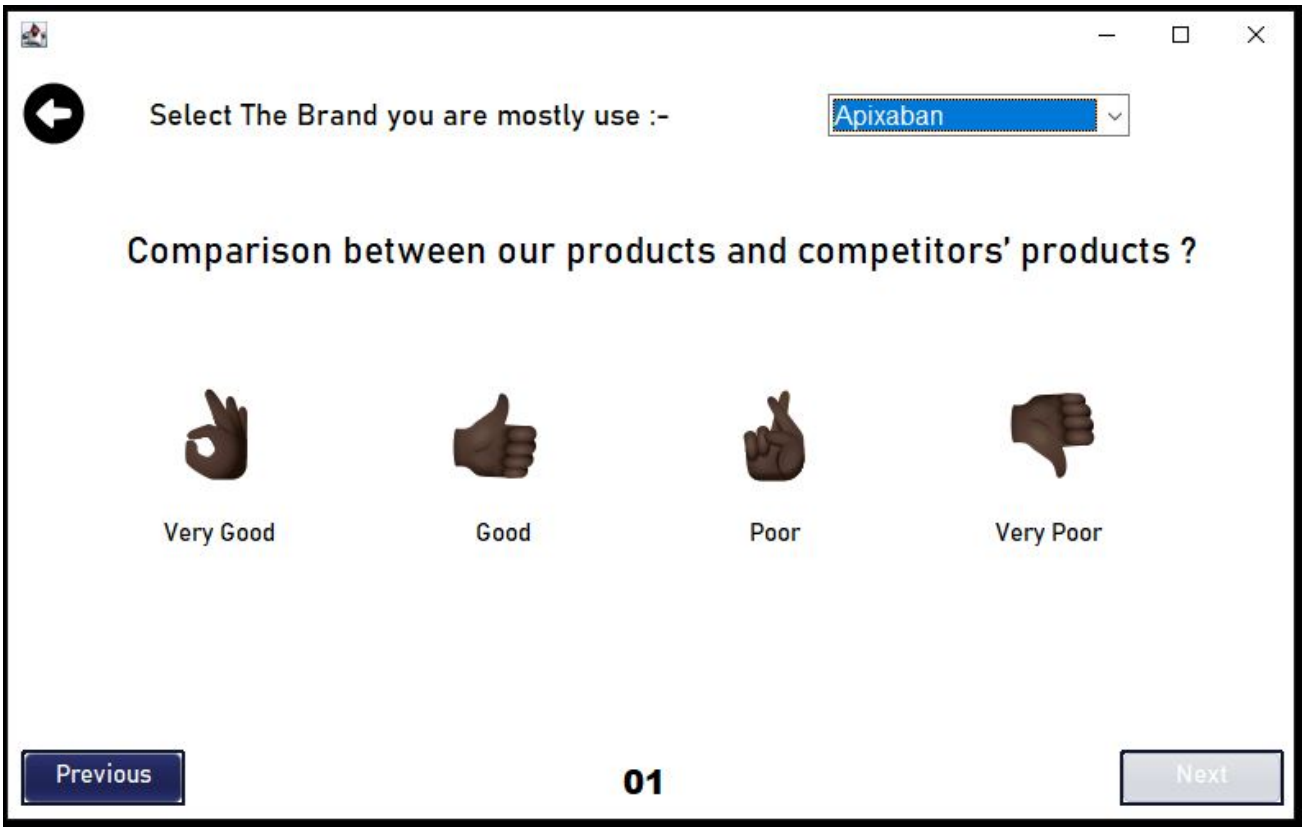


*Figure 7: Question 01*

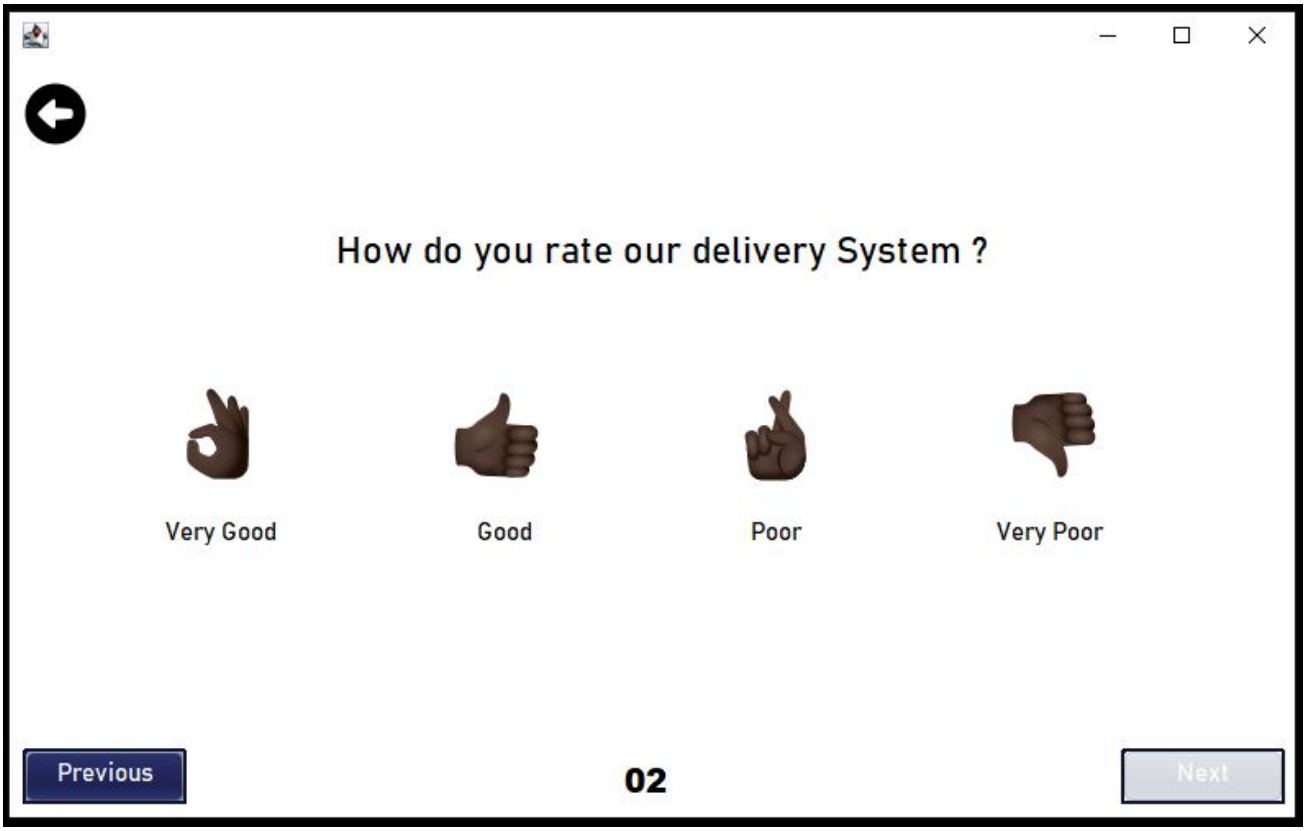


*Figure 8: Question 02*


*Figure 9: Question 03*


*Figure 10: Question 04*


*Figure 11: Question 05*

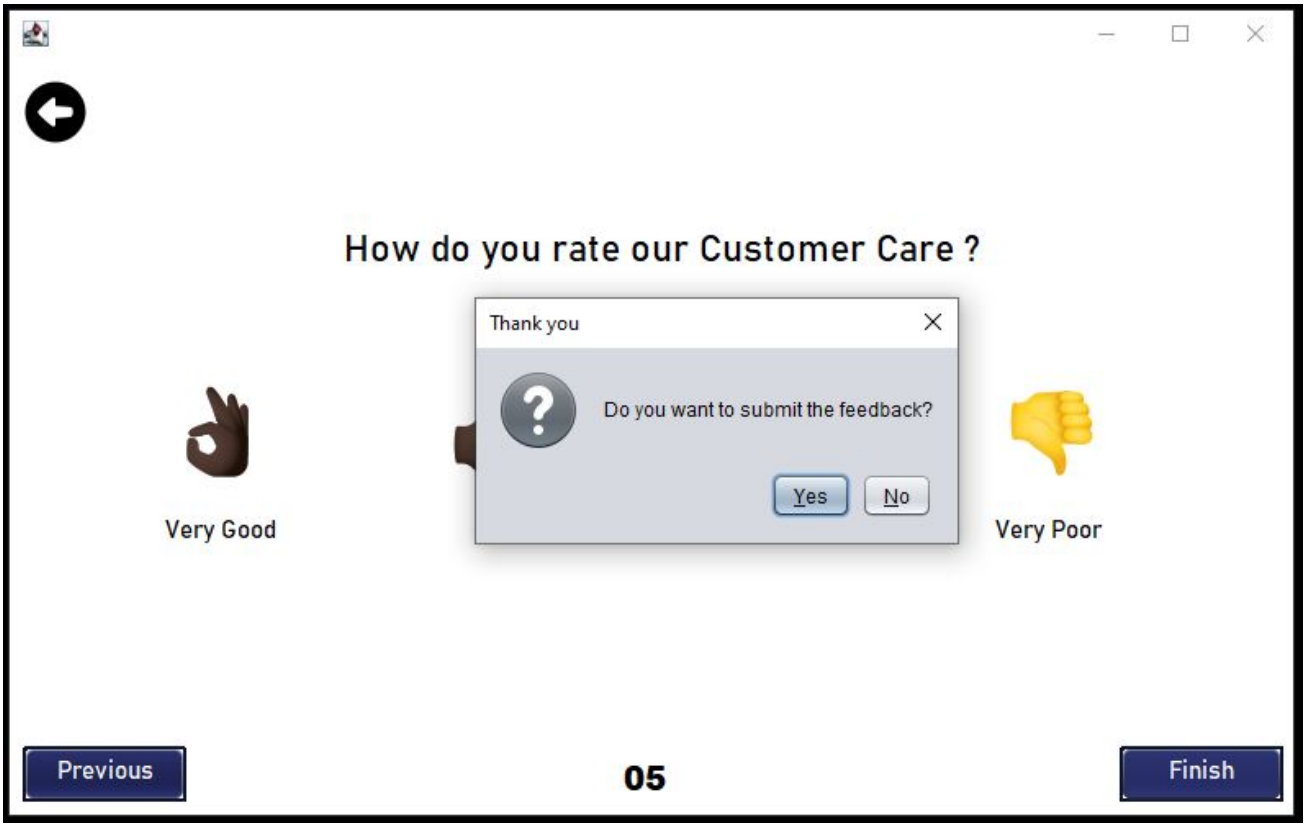


*Figure 12: Validation*

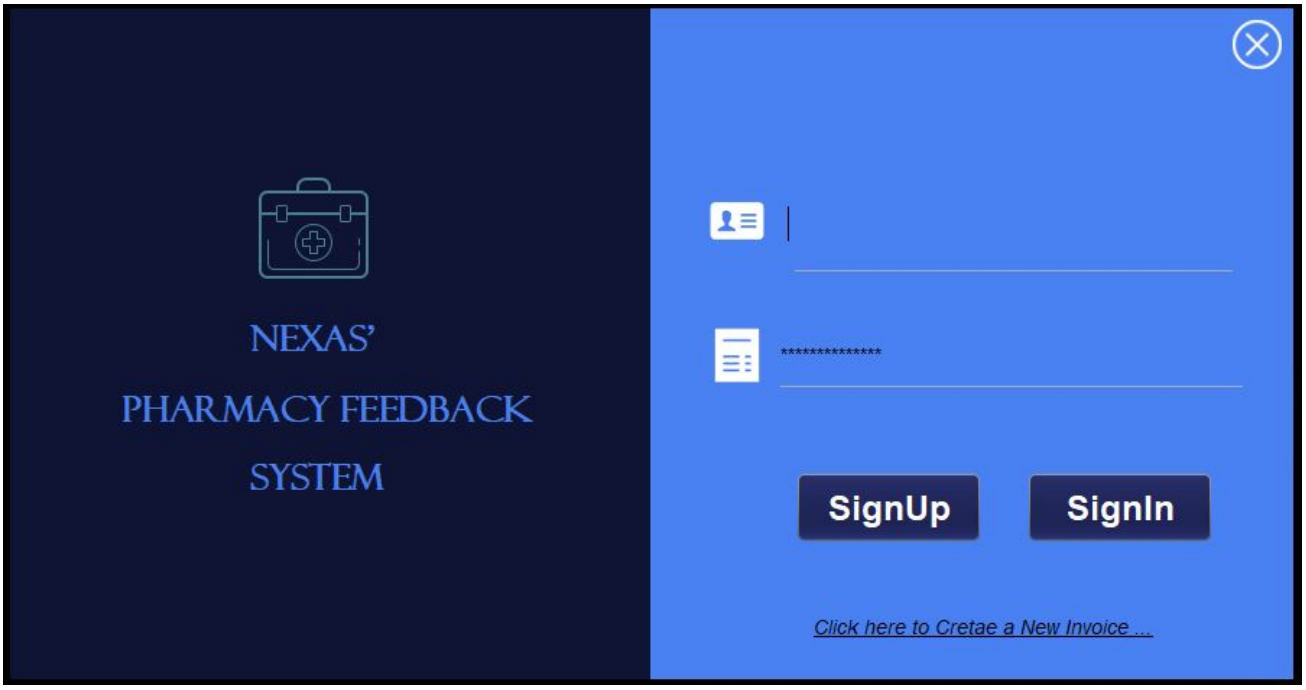


*Figure 13: Admin login Screen*


*Figure 14: Analytics Bar Chart View*

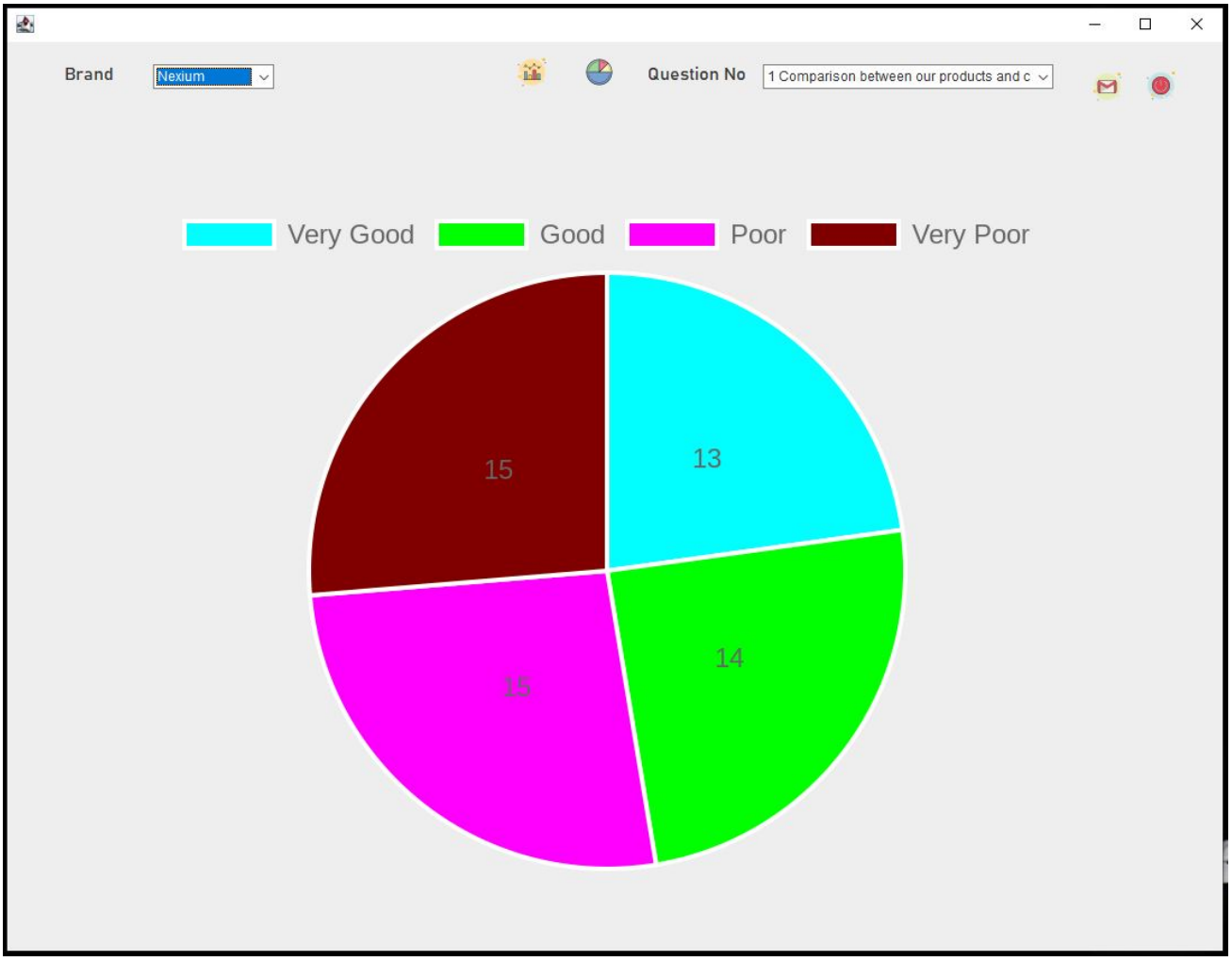


*Figure 15: Analytics Pie Chart View*

## Explanation of Testing Used

The quality assurance plan aims to ensure that the system provided to the client meets their requirements without any failures or drawbacks while maintaining the standards the industry requires. To ensure total quality, the system should meet the standards of the clients.

- Feedback
    - Since the system will be designed using an agile methodology, the developers will be able to finish one part of the system before moving on to the next. As a result, it is possible to obtain input from the client on each module of the system until the entire system is completed.
- Reviews
    - An internal quality assurance test would also be implemented independently to assess whether the ongoing project complies with the requirements of the client.

Additionally, the system would also undergo many cycles of testing to ensure the quality of the product. Testing would be carried out in the following methods;

- Black Box Testing
    - This approach enables users with little or no programming experience to test the system or to test the system exclusively based on the system's outer workings.
- White Box Testing
    - This method allows developers with wide programming knowledge to test and review the outer workings of the system as well as the inner coding used to build the system
- Alpha Testing
    - This approach enables the system to be evaluated in its early stages of development, using testing values and scenarios to verify the system's functionality.
- Beta Testing
    - This method allows the system to undergo with actual data from the company, as well as probable future users of the system

In light of the above testing methods, black box and white box testing would be used to assess individual components of the system, while alpha and beta testing would cover the whole system. Since the interface is such an integral part of the system, interface assessment methods can be used to ensure that the end product is of high quality. An interactive design assessment tool is used for this.

## Deliverables

The timeline of the assignment is three months. The progress of the system was monitored weekly by the supervisor in charge up until the formal review in week 9. We were given tasks weekly to finish them by one week. The working product is completed and it is working successfully.

## References

*Quickchart.io. 2021. Open Source Chart Image API | QuickChart. [online] Available at: <https://quickchart.io/> [Accessed 3 May 2021].*

*Tutorialspoint.com. 2021. Software Testing - Methods - Tutorialspoint. [online] Available at: <https://www.tutorialspoint.com/software_testing/software_testing_methods.htm> [Accessed 3 May 2021].*

*Youtube.com. 2021. Before you continue to YouTube. [online] Available at: <https://www.youtube.com/watch?v=A7HAB5whD6I> [Accessed 3 May 2021].*

*Youtube.com. 2021. Before you continue to YouTube. [online] Available at: <https://www.youtube.com/watch?v=ILeAeFZOkMI> [Accessed 3 May 2021].*