**Project 1: Inventory App**

Justin Starr

Department of STEM

CS 360 – Mobile Architect & Programming

Professor Jerome DiMarzio

November 12, 2023

**Project 1: Inventory App**

As a newly hired member of the mobile application development team for Mobile2App, a software development company, the following proposal has been written that describes a mobile application that will be written by our team. The proposal describes the mobile application, its major components, and the functionality that is necessary based on the given scenario. It continues to describe the various users of the mobile application, assumptions being made regarding their needs and preferences, the various screens and features that will be necessary to produce a user-centered UI for the app, and how the functional app requirements will be presented in the code and connected to the UI.

The application that has been chosen for development is the Inventory application. The purpose of the app is to facilitate the process of a warehouse inventory management process, which will be used to track items in a warehouse. The major components of the application include a database with at least two tables, one for storing the inventory items and one for storing usernames and passwords. The application must provide a method for users to log into the application, or to create a username and password if they have not previously used the application. After a user is logged into the application the user will be able to see a list of all the items that the warehouse has in its inventory along with the quantity of each item. From this list, the user can select any item that is in the warehouse's inventory, and they will be able to adjust the quantity for each item in the inventory. From here they can also choose to remove an item from inventory in its entirety. Users will be able to add new items to the inventory from the home screen where a button will be prominently displayed where they can click to add a new item to the warehouse’s inventory. Lastly, the application will notify the user when any item’s quantity in inventory has been depleted or reduced to zero.

A variety of potential users have been identified for the application that is going to be developed. These users include stakeholders such as the business owner(s) who would be interested in knowing how much inventory his/her company has and how much money is invested or tied up in the company's inventory. When a stakeholder wants to have this information, when they click on items that are in the inventory, it should state how much those items are worth. Another user of the application would be any ordering managers for the warehouse. They need to be able to know how much of each item they have so they know when to order more products. Also, the application is designed so that managers do not have to guess how much of an item the warehouse has at any given time, and they can be notified when an item is reduced to zero. This helps safeguard against running out of items and even over-ordering of items, which can be a significant problem for warehouse management. Another potential user of the application is warehouse employees such as order fulfillment workers. If the warehouse is large, it would be beneficial for the employee to know inventory information before attempting to retrieve an item from inventory. By utilizing the app, workers can find an item that is in inventory and determine if the item is in stock before fetching an item. If the item is out-of-stock for whatever reason, time will have been saved from not having to go to the item’s location in the warehouse to find this information. Lastly, another potential user of the application could be various authorized warehouse vendors. If vendors have expectations from warehouse stakeholders to ensure their products are always in stock, it would be beneficial for vendors to have the ability to use the application to manage their products more easily and effectively within the warehouse. This is significantly beneficial for vendors where their product's location in the warehouse may be hard to get to and/or count. The application will provide an easy method for vendors who need to have this type of information.

The first screen that is displayed to users when the application executes is the login screen. The application’s name will be displayed to the user along with a text field where users can enter their username and a text field where users can enter their password. Once the user does this, the user has two options of buttons that they can press. The user can choose to press the “Login” button or the “Create New User” button. If the user has never used the application before, they should choose the “Create New User” button which will then create the new user by storing their username and password into the database that stores this information. Once this happens, a toast message will be displayed to the user to log in with their new credentials. The user will type in their username and password, and this time select the “Login” button. Returning users will always choose the “Login” button. If a returning user tries to create a new user, a message will be displayed that the user already exists, and to please log in using their existing credentials. According to Google, “account creation and login can be a major source of confusion and app abandonment” (Google Inc., 2023). The goal of this screen is to simplify the login experience which will enhance their overall experience.

The “Home” screen is the screen users will be directed to upon authentication. This screen will display a list of all the items that are in the warehouse’s inventory. This list will be scrollable so that users can easily see all the inventory items and be displayed to the user with a grided layout design. Each inventory item is clickable which will navigate the user to a screen with specific details and functions regarding that item. Additionally, there will be a button displayed prominently that has a plus sign symbol. This button will allow users to add new items to the existing inventory.

At the top of the screen, there will also be a notification button that resembles a bell. When the notification button is pressed, the user is navigated to the notifications screen where the user can see any notifications the system has made, such as when an item in inventory has been depleted or reduced to zero. Users will know when they have new notifications by making use of material design components, such as communication components, which provide “helpful information, by alerting users with badges…,” (Google Inc., 2023). While on the notification screen, which users can navigate to by pressing the notification button on the home screen, the application will display to the user any item in inventory that has a quantity of zero. If there are no items in inventory with a quantity of zero, a message saying “No current notifications” will be displayed to the user. The user can intuitively navigate back to the home screen by pressing the back button.

From the home screen, when a user presses an item that is in inventory, they are navigated to the item details screen. This screen will display, to the user, specific information regarding that item, such as the item's quantity, the unit value of the item, and the total value for the quantity of that item in inventory. On this screen, there will be a button with a plus sign that is different than the plus sign on the home screen, but similar, where the user can press it and it will increase the number for that item in inventory. Like the plus sign button, there will be a minus sign button on the right side of the plus sign button that when pressed will reduce the quantity of that item in inventory by one. If the item is reduced to zero, the application will alert the user on the home screen, with a badge, that there is a new notification. The notification is to remind the user that an item in inventory has been reduced to zero. Also, there will be a button below the item’s information that is presented to the user with a trash can symbol, and when pressed, the user can completely remove the item from inventory. The user can intuitively navigate back to the home screen by pressing the back button.

In addition to incorporating the mentioned design features, the users will be kept in mind and at the forefront of the development process by ensuring industry standards and best practices are followed. It is essential that apps incorporate and follow specific guidelines such as accessibility guidelines, which help to guide developers in creating applications that ensure the app’s content is visible or “as visible as possible by checking color contrast and text sizing, and that components are visually comprehensible and easy to discern from each other,” (Goole inc., 2023). Displaying text that is too small to read or text that is dark on dark backgrounds are examples of poor design which results in poor user experiences. If users cannot read text, there is a high probability that users will not enjoy using the app and avoid using the application. Similarly, buttons should be of appropriate size and properly distanced from other buttons so that users do not make mistakes when taking an action.

When considering how the functional app requirements will be represented in the code and connected to the UI, we begin by looking at each screen, their components, and what those components do, whether it be to display or accept data as input, and where this data might come from. To begin with, on the login screen, Basic information is displayed next to EditText boxes with TextViews that inform the user of what type of information the EditText boxes are expecting to receive as input. Two EditText boxes are used to enable the user to type in their username and password as strings. Once the “Login” button or “Create New User” button is pressed, the appropriate action is taken. If the “Create New User” button is pressed, the string information is then passed into the database where it is added to the table of users and passwords. If the login button is pressed, the string information is taken and compared to the database that contains the table of usernames and passwords. If the user is authenticated, no further information is passed back to the user on the login screen and the user is navigated to the home screen. If the user is not authenticated, meaning the user credentials did not match any existing username or password, a message is sent back to the user on the login screen that the username or password was incorrect.

Once a user is authenticated, the application automatically retrieves and displays the inventory items from the database table that contains the inventory information. The user does not have to request this information; as a feature of the app, the application automatically retrieves the information for the user to view. The only input the application accepts as input on the home screen is button presses, which occur by creating event listeners within the code. Once a button is pressed, the user is navigated to the appropriate screen based on what the onClick event listener for each button is told to do in the code. On the home screen, the user can choose to press the notification button, the plus sign button, or any item displayed on the screen, which are each of the items that are in the warehouse’s inventory. When a user navigates to the notification screen, the application automatically will collect data from the inventory database and display to the user any item in inventory that has a quantity of zero. This screen does not accept any input from the user except that the displayed list can be scrolled through, and the user can intuitively move back to the home screen by pressing the back button. Again, this is accomplished through event listeners.

The last screen that users can navigate to from the home screen is when a user selects a specific item in inventory to view. When an item is clicked, the application, again, goes to the database that contains all of the inventory items and their associated data. That data is passed back to be displayed to the user in a gridded layout. This screen does not accept any type of text-related data from the user. Instead, it also makes use of onClick event listeners for the buttons on this screen. There is a button, a plus sign, that when pressed, the quantity for that item is increased by one. Then, the application sends that number back to the application's database to store the new quantity for that particular item in inventory. The same applies to the minus button except instead of incrementing the item’s quantity, it decreases the item’s quantity by one unit. This value is then sent to the database where the item’s new quantity is stored. Similar to the plus sign and the minus sign, there is a delete button that has a trash can symbol on it that when pressed, the application will remove the item completely from the database and all of its associated data. Once this process has been completed, the user will automatically be redirected to the home screen.

**References**

Google Inc. (2023). *Design for Mobile*. <https://developer.android.com/design/ui/mobile>