**CS 360 Project One Option 1: Inventory App Development Proposal**

1. ***Articulate the goals of the project.***
2. ***Describe the application you have chosen, its major components, and what functionality will be necessary based on the scenario.***

The app that I have chosen is the inventory tracking app, which will be designed to help users efficiently manage the tracking and monitoring of various items. The major components that will be included will be the ability for users to login with credentials, a display that users can view the given inventory and available quantity, a screen that allows users to manage item by updating quantity, and the ability for users to receive a notification when stock of specific items are low.

**Describe the users of the application and the assumptions being made regarding their needs and preferences.**

1. ***As you think about potential users, consider the following questions to guide your response:***
   1. ***How many different types of users can you identify?***
   2. ***What different goals might individual users have?***
   3. ***What will users need to achieve their goals in this app?***

The inventory tracking app is great for various users who have independent goals related to inventory management and monitoring. These include:

* Retail store manager/vendor: They will need to track and monitor items to ensure there is a consistent supply.
* Small Business Owners: They require personal updates on stock quantities and ordering.
* Entrepreneurs: They need to consistently track materials, especially for an e-commerce business with an on-hand supply of materials.

Individual goals for these each user include:

* Retail Store Manager/Vendor: They will need to ensure the business maintains a healthy supply of products.
* Small Business Owner: Need to manage the business at all levels and may oversee monitoring and reordering items as needed.
* Entrepreneur: Solely responsible for the stock of product and will require constant monitoring and tracking for business continuity.

For these users to achieve their goals, the app must include various features such as:

* A simple and secure login process.
* A user-friendly interface that can be easily navigated.
* The ability to add/update items as needed with detailed descriptions.
* Grid display that provides a clear view of all the products.
* The ability to manage items at the individual level by adding or deleting items from the database.
* A notification system that alerts the users of low stock quantities of items.

***Discuss the screens and features that will be necessary to produce a user-centered UI design for the app.***

1. ***Include a high-level description of every necessary screen and feature that will be in your app.***

The inventory app will feature five primary screens, the Login screen, the Registration screen, the Inventory Display screen, the Inventory Management screen, and the Notification screen. The Login screen will provide the user with the ability to login using a username and password, or the ability to sign up and register. If a user chooses to register, they will be directed to the Registration screen to input credential details like a username and password. Afterwards, they will be redirected back to the Login screen.

Once the logs in they will be brought to the Inventory Display screen, where this will act as the main home screen that displays a list of the items included in the database. The next available screen will be the Inventory Management screen, where the user can gain more detail about a specific item and edit the quantity they wish to update to the database. Lastly, the Notification screen is also available, where the user can view a list of the items that are included for notification when item quantity is running low.

The features that the app will support will follow a logical progression across screens. The app will start with the Login screen, where there will be input text fields for both the username and password, and a button where a user can sign up and register. The Inventory Display screen will feature a grid or list of all the inventory items, a search bar where the user can search for any item, and an add item button. If a user searches for an item using the search bar, a dropdown will display with a list of items relevant to the search. Once the user finds the item they searched for, a notification window will appear asking them to add the item to the inventory.

The Inventory Management screen will include various features such text fields for the name, description, and quantity of an item. A save button to save any instance of the item that was edited where it will be updated in the Inventory Display screen. The management screen will also feature the ability to increase or decrease the quantity of the item. As well as the ability to delete the item altogether from the database.

In the Notification screen, the screen will feature a list of all the items that are added for notification to the user, with the ability to add or delete new items as needed. It will also feature the ability for the user to clear all notifications. The user can also set the amount at which the system will notify when an item reaches a certain quantity.

B. ***Explain how a user might move from one screen to another.***

* 1. ***You may choose to support your ideas with a simple diagram or illustrations to better represent the different components.***

When the user initially opens the Inventory Tracking app, they will be greeted by the Login screen with text fields for a username and password, and a sign-up link. If a user needs to create a new account, they can click the sign-up link and are redirected to the Registration screen. After entering their credentials, the user is navigated to the Inventory Display screen.

On the Inventory Display screen, users are shown a grid display of all the inventory items and can search for any specific items. They can also click the add item button to input information for a new item, which brings them to the Item Management screen.

In the Inventory Management screen, users can modify details for newly created items or existing items. Item details such as manually adjusting quantities using the increase/decrease button or even deleting an item. After saving a modified item, the user is returned to the Inventory Display screen. Lastly, users can select a button at the top of the screen that allows access to the Notification screen, where users can manage notification alerts related to item quantity.

1. ***Justify your decisions by referencing the Android Design and Quality Guidelines linked in the Supporting Materials section.***

The design and flow decisions of the Inventory Tracking App are guided by the Android Design and Quality Guidelines to ensure a high-quality user experience that is consistent, intuitive, and efficient.

The app uses standard Android UI components like text fields, buttons, and grid views that align with the guideline’s recommendations to maintain consistency throughout the app. This approach helps users to familiarize themselves with the app’s interface. They will also encounter familiar elements that are consistent with design elements from other popular apps that aim to reduce the learning curve and improve user experience (*What a Great User Experience Looks Like*, n.d.).

The app provides users with clear navigation flow and easy-to-use controls. For instance, the login screen has straightforward options for logging in or signing up. While the Inventory Display screen features the ability to easily add or modify any available items. These features aim to provide the user with the control and freedom that the guidelines recommend so users don’t feel constrained or confused when interacting with new apps.

Input validation on the login and Inventory Management screen help to prevent unnecessary errors, while also providing a safety net for users (*Security Guidelines*, n.d.). This approach follows the guideline’s principle of helping users avoid and recover from errors. This type of feature improves not only a user’s experience but also their trust and encourages reliability and reusability.

The navigation flow between screens is designed to be intuitive and logical and follows the guidelines advice for providing a coherent navigation structure (*Core App Quality*, n.d.). For instance, the users can easily move between screens as needed, with clear navigation options and consistent UI elements that guide them through the app’s different functionalities.

***Discuss how the functional app requirements will be represented in the code design and connected to the UI.***

1. ***Explain the calls that show the flow of data between code and screens.***

The app begins with the Login screen, which prompts the user to enter details such as a username and password. Once the user inputs this information and selects login, the app will initiate an API call that will authenticate the user.

On the Inventory Display screen, the app will fetch data related to the inventory from the database once it loads. This might involve an API call that retrieves the necessary data from a local or remote database. The data will then be displayed in a type of view model.

Within the Inventory Management screen, the user will be able to edit details related to the items, such as the description, or quantity. Once the user saves the edited information, the app will send the data back to the database.

Finally, with the Notification screen, the app will retrieve and display notifications about low or zero inventory on items. When a user adds or deletes items from the notification status, the app will update the status in the database. This can be managed using a ViewModel to retrieve and update the necessary notification data.

The flow of data between code and screens in the Inventory Tracking app will involve a series of API calls and database operations that will handle the necessary user authentication, data retrieval, and updates.

1. ***List the major UI components on each screen.***

* **On the Login screen:** The username field, password field, login button, and Sign-up link.
* **Registration screen:** Username field, Password field, Confirm password field, Register button, and Return to Login link.
* **Inventory Display screen:** Inventory in grid display, search bar, and add item button.
* **Inventory Management screen:** Item name field, Description field, Quantity field, Save button, Delete button, and Increase/Decrease buttons.
* **Notification screen:** Notification list, Clear notification button, and add notification button.

1. ***Determine what data each component will either display or accept as input and where the data might come from.***

In the Login screen, the username and password field will accept data from the user in the form of a username and password. The login button will trigger the authentication process, while the input data will be sent to the authentication API for validation. The Sign-up link is a button that does not require data input.

The Registration screen will accept text from the user for both the username and password. Along with the password confirmation field. The Register button will submit the registration details to the database.

The Inventory Display screen components like the Inventory grid will display the data that is retrieved from the inventory database. The search bar will accept user input to filter the displayed items. The add item button navigates to the Inventory Management screen and doesn’t directly accept input from the user.

The Inventory Management screen features the Item name field, description field, quantity field, which each accept input from the user in the form of text or a numerical quantity. The save and delete button will each send the edited info to the inventory API to update the database. Lastly, the increase/decrease buttons affect the quantity field’s value, and this data comes from the user as they adjust the quantity field, which is then saved to the database when the save button is selected.

Lastly, the Notification screen displays the list of notifications, which is composed of the data that is retrieved from the notification database via an API call and then displayed. The clear notification button updates the status of the notification database once the button is selected by the user.

What a great user experience looks like. (n.d.). Android Developers. https://developer.android.com/quality/user-experience

What great technical quality looks like. (n.d.). Android Developers. https://developer.android.com/quality/technical

Core app quality. (n.d.). Android Developers. https://developer.android.com/docs/quality-guidelines/core-app-quality

Design for safety. (n.d.). Android Developers. https://developer.android.com/quality/privacy-and-security

Security guidelines. (n.d.). Android Developers. https://developer.android.com/privacy-and-security/security-tips