CS-360-R4875 Mobile Architect & Programming 24EW4

3-3 Project One | DEONNE LUDWIG 03/23/2024

**App Development Proposal | OPTION1: Inventory App**

**WIM | WAREHOUSE INVENTORY MANAGEMENT**

**GOALS:**

The overarching purpose of the Inventory App is to provide users with a tool for efficiently managing and tracking items in a warehouse. Key components necessary to support the app's goals include:

**Database:** A database with at least two tables, one to store the inventory items and one to store user logins and passwords.

**User Authentication Screen:** A screen for logging into the app to implement security. Note that this should also be used to create a login if the user has never logged in before.

**Inventory Display Screen:** A screen, with a grid, that displays all items in the inventory.

**Item Management Screen** A mechanism by which the user can add and remove items from inventory. The user can increase or decrease the number of a specific item in the inventory.

**Inventory Alerts:** A mechanism by which the application will notify the user when the amount of any item in the inventory has been reduced to 0 (zero)

**USERS / NEEDS / PREFERENCES:**

**Warehouse Inventory Control Manager:** to monitor existing inventory levels, manage needed inventory orders and respond to alerts.

**Warehouse Inventory Accountant:** to easily view and evaluate the existing inventory and its worth and cost to replenish.

**Warehouse Inventory Analyst:** to analyze trends, generate reports, and optimize inventory workflows.

**UI / SCREENS / FEATURES / USER MOVEMENTS / EXAMPLE CODE:**

|  |  |  |
| --- | --- | --- |
|  |  | **User Authentication Screen:**  HIGH-LEVEL DESCRIPTION:  A screen for logging into the app to implement security. Note that this should also be used to create a login if the user has never logged in before.  USER MOVEMENTS:  The user is able to enter their username and password to log into the app or click on “Create an account” to enter a username, email and password for the new account. They can also go back to login.  EXAMPLE CODE:  <TextView  android:textSize="26sp"  android:layout\_columnSpan="1"  android:layout\_gravity="center"  android:text="LOGIN" /> <TextView  android:text="USER NAME" />  <EditText android:layout\_gravity="fill\_horizontal" />  <TextView  android:text="PASSWORD" />  <EditText android:layout\_gravity="fill\_horizontal"  android:inputType="textPassword" />  <Button  android:layout\_column="1"  android:layout\_gravity="center"  android:text="NEW HERE? CREATE AN ACCOUNT" />  NOTE: THE USER WILL BE ABLE TO INPUT INFORMATION INTO THE WHITE BOX AND THIS INFORMATION WOULD BE CALLED AND VERIFIED (FOR EXISTING USERS OR SAVED TO A TABLE WITH THE USER LOGIN INFORMATION FOR NEW USERS. |
| **A screenshot of a computer screen  Description automatically generated** |  | **Inventory Display Screen:**  HIGH-LEVEL DESCRIPTION:  A screen, with a grid, that displays all items in the inventory.  USER MOVEMENTS:  The user can click on one of the categories for inventory to view the details of the category or search. On the category detail screen, the user can click on the row to edit or delete. They can also add a new item. Or go back to the previous screen.  EXAMPLE CODE:  CATEGORY MAIN  <Button  android:layout\_column="1"  android:layout\_gravity="center"  android:text="PARENT LUMBER" />  <Button  android:layout\_column="1"  android:layout\_gravity="center"  android:text="CTL LUMBER" />  (…ETC)  CATEGORY DETAIL  <TextView  android:textSize="26sp"  android:layout\_columnSpan="1"  android:layout\_gravity="center"  android:text="PARENT LUMBER" />  (READ AND DISPLAY ITEMS FROM  TABLE OF INVENTORY)  <Button  android:layout\_column="4"  android:layout\_gravity="left"  android:text="BACK" />  (…ETC)  NOTE: IN THE CATEGORY MAIN SCREEN THE USER WILL BE ABLE TO CLICK ON THE CATEGORY THAT THEY WOULD LIKE TO VIEW. THE CATEGORY DETAIL SCREEN WILL DISPLAY INVENTORY ITEM INFORMATION PULLED FROM THE SUPPORTING TABLE. THEY WILL BE ABLE TO EDIT OR DELETE EXISTING ITEMS OR ADD A NEW ITEM TO THE TABLE. |
|  |  | **Item Management Screen**  HIGH-LEVEL DESCRIPTION:  A mechanism by which the user can add and remove items from inventory. The user can increase or decrease the number of a specific item in the inventory.  USER MOVEMENTS:  On the category detail screen, the user can click on the row to edit or delete. They can also add a new item. Or go back to the previous screen. On the item detail screen (edit) the user can change the values for the item and save to update.  EXAMPLE CODE:  CATEGORY DETAIL (SEE ABOVE)  ITEM DETAIL  <TextView  android:textSize="26sp"  android:layout\_columnSpan="1"  android:layout\_gravity="center"  android:text="PARENT LUMBER" />  <TextView  android:text="TAG" />  <EditText android:layout\_gravity="fill\_horizontal"  android:inputType="textTag" />  (…ETC)  <Button  android:layout\_column="4"  android:layout\_gravity="left"  android:text="BACK" />  (…ETC)  NOTE: CATEGORY DETAIL (SEE ABOVE)  IN THE ITEM EDIT SCREEN THE USER WILL BE ABLE TO EDIT THE VALUES FOR THE ITEM AND SAVE THE CHANGE BACK TO THE TABLE. |
|  |  | **Inventory Alerts:**  HIGH-LEVEL DESCRIPTION:  A mechanism by which the application will notify the user when the amount of any item in the inventory has been reduced to 0 (zero)  USER MOVEMENTS:  On the category main screen (not shown but would have “!” by the user icon), category detail screen, and the item detail screen (edit) an alert can notify the user by putting a red “!” by the user account and marking the low item with red text color in each screen.  NOTE: AN ALERT CAN BE CREATED BY IF INVENTORY = 0 THEN.. PLACE “! NEXT TO ACCOUT ICON AND CHANGE STYLE OF ITEM IN CATEGORY DETAIL AND ITEM DETAIL.  ALSO A POPUP ALERT DIALOG COULD BE ADDED AFTER THE USER LOGS IN. |