**Appendix E: Completed Test Plan**

| **Success Criterion** | **Method of Testing** | **Expected Outcome** | **Actual Outcome** |
| --- | --- | --- | --- |
| The application should: |  | |  |
| Run-on an iPhone 8 | Ask my client to use the product on her phone | All features function on my client’s phone, and the layout is properly adjusted for the size specifications. | Works as intended on iPhone 8 |
| Require login to view any data of purchases and totals | Try logging in, and try signing up. Also, attempt to login using incorrect details | The homepage opens if login is successful. Alerts are shown if details are false | Works as intended with authentication catching all false details |
| The homepage should: |  | |  |
| Display the current month’s totals (including category totals) up to the current date | Run the homepage | All totals are displayed for the current month | Expenditure for each category in a month and total expenditure for that month is clearly displayed |
| Have a navigation menu to see receipts (from any month and category) and items bought | Test all the buttons on the homepage which are meant to navigate to other screens | Correct screens are loaded | Works as intended with very low wait times between loading screens |
| Contain totals of previous months upon user’s request | Attempt to navigate to the menu which shows totals of previous months | Previous month totals are displayed | Works as intended with the correct totals being displayed from the database |
| The interface should: |  | |  |
| Be “simple and intuitive to use” by using buttons and intuitive symbols to improve usability | Run all pages of the product | All buttons are blue, and their icons clearly depict what they do | All buttons are blue and the icons are clear |
| Be navigable and not require many steps to view expenditure | Check if navigating to the expenditure page takes many steps | User is able to go to the totals page within 3 button clicks | Efficient navigation, can get to the totals page by just one click after logging in |
| Have consistent placing of buttons | Run all the screens in the program | “Logout” button and the bar buttons for the different screens are in the same place | All buttons are placed and spaced consistently and logically |
| The database should: |  | |  |
| Store data on purchases made of each item and related values | Add records for several receipts | The database contains the records of the added receipts | Works as intended with a unique ID being created for each receipt |
| Have unique identifiers to locate a purchase | Add a record for a receipt and check database | The receipt is identified by a new ID that hasn’t been used previously | The receipt is identified by the token from the upload to Tabscanner, which is unique |
| To manage receipts, the user should be able to: |  | |  |
| Take a photo of the receipt after purchase, or choose an old photo | Attempt to take a photo of a new receipt, and attempt to choose an old photo | Camera app or Photo app loads and allows the user to take/choose a photo | Works as intended with Camera/Photo Library connecting smoothly |
| Cancel the upload to the database of a receipt | Choose a receipt photo and try to cancel the upload, then check database | Database shows no record of the receipt | Receipt upload does not occur until user has verified receipt data |
| View and edit the receipt in an OCR format | Choose a receipt and try and change each of its data. Check database after edits are made | The receipt’s data is displayed to the user through, for example, text fields. Each value is able to be changed, and the database reflects these changes | User is able to edit the OCR data before verifying the receipts and uploading them |
| Delete unwanted receipts | Delete an old receipt and check to see if the database contains it. Also, check that the totals have been changed | The database does not contain the deleted receipt. Totals are updated to exclude the expenditure on the deleted receipt | Works as intended. As soon as the receipt is deleted the totals are recalculated. The other months’ totals are not affected |
| To review expenditure, the program should: |  | |  |
| Calculate monthly totals for all the categories: Groceries, Lifestyle, Personal, Transportation, Miscellaneous | Add receipts and check if the totals are correct | Expenditure on each receipt from each category sums up to the correct total | Works as intended, with the correct total expenditure for each category is displayed |
| Calculate the total expenditure in a specific month | Add receipts from different months and check if the total for the current month is correct | Total is the correct sum of all the receipts | The total expenditure for each month is the sum of the individual category expenditures |
| Allow the user to select who bought a receipt, and then display this data when they are looking at the receipt | Before the receipt is uploaded to the database, specify that another person bought the receipt | Database record shows who bought the receipt. When the receipt is checked on the app, the name of the buyer is shown | User can select who bought the receipt by typing the name in before verifying and uploading. After upload, the database contains this data, and the user can see it in the manage receipts page. |