|  |  |
| --- | --- |
| Name: | Launch App |
| Summary: | Open the personal monthly budget app |
| Version: | 1.0 |
| Preconditions: | System displays the personal monthly budget app icon |
| Triggers: | User selects the personal monthly budget app icon |
| Main Success Scenario: | 1. System builds the user interface 2. System displays the user interface |
| Alternative Success Scenarios: |  |
| Post Conditions: | 1. System displays the name of the app 2. System displays the logo 3. System displays the email edit text field 4. System displays the password edit text field 5. System displays the login button 6. System displays the “Sign-Up” link |
| “Used” Use Cases: | 1. Logout |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Login |
| Summary: | Login to the user database |
| Version: | 1.0 |
| Preconditions: | 1. System displays the name of the app 2. System displays the logo 3. System displays the email edit text field 4. System displays the password edit text field 5. System displays the login button 6. System displays the “Sign-Up” link |
| Triggers: | App loads after the user selects the app icon |
| Main Success Scenario: | 1. System logs into the database with an existing user account |
| Alternative Success Scenarios: | 1. System creates a new account with an email and password from the user input |
| Post Conditions: | 1. System displays the name of the app 2. System displays the logo 3. System displays the email edit text field 4. System displays the password edit text field 5. System displays the login button 6. System displays the “Sign-Up” link |
| Alternative Post Conditions: | 1. System displays the email edit text field 2. System displays the password edit text field 3. System displays the “Sign Up” button |
| “Used” Use Cases: | 1. Open Report 2. Create Report 3. Exit Report |
| Extension Points: |  |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Logout |
| Summary: | Log the user account out of the database |
| Version: | 1.0 |
| Preconditions: | 1. System displays the “Open budget” link 2. System displays the “Create new budget” link 3. System displays the “Logout” link |
| Triggers: | User selects the “Logout” link |
| Main Success Scenario: | 1. System logs out of the database |
| Alternative Success Scenarios: |  |
| Post Conditions: | 1. System displays the name of the app 2. System displays the logo 3. System displays the email edit text field 4. System displays the password edit text field 5. System displays the login button 6. System displays the “Sign-Up” link |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Open Report |
| Summary: | Open the monthly budget report created before |
| Version: | 1.0 |
| Preconditions: | 1. System displays the “Open budget” link 2. System displays the “Create new budget” link 3. System displays the “Logout” link |
| Triggers: | User selects the “Create new budget” option |
| Main Success Scenario: | 1. The existing monthly budget that was previously created is pulled from the database and displayed so that the user can update and store any new changes into the database. |
| Alternative Success Scenarios: |  |
| Post Conditions: | 1. System displays the monthly budget table that contains text views and edit text fields for user input 2. System displays the “Save” button 3. System displays the back button |
| “Used” Use Cases: | 1. Update Expense 2. Delete Expense 3. Add Expense |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Create Report |
| Summary: | Create a new monthly budget report |
| Version: | 1.0 |
| Preconditions: | 1. System displays the “Open budget” link 2. System displays the “Create new budget” link 3. System displays the “Logout” link |
| Triggers: | User selects the “Create a new budget” link |
| Main Success Scenario: | 1. System creates a new monthly budget report |
| Alternative Success Scenarios: |  |
| Post Conditions: | 1. System displays the monthly budget table that contains edit text fields 2. System displays the “Save” button 3. System displays the back button |
| “Used” Use Cases: | 1. Add Income 2. Add Expense |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Update Expense |
| Summary: | Update an expense amount |
| Version: | 1.0 |
| Preconditions: | 1. System displays the monthly budget table that contains edit text fields 2. System displays the total expenses 3. System displays the total savings 4. System displays the “Save” button 5. System displays the back button |
| Triggers: | User selects the “Open budget” link |
| Main Success Scenario: | 1. System re-calculates the total expenses 2. System re-calculates the total savings 3. System stores the new changes into the database |
| Alternative Success Scenarios: |  |
| Post Conditions: | 1. System displays the monthly budget table that contains edit text fields 2. System re-calculates the total expenses 3. System re-calculates the total savings 4. System displays the total expenses 5. System displays the total savings 6. System displays the “Save” button 7. System displays the back button |
| “Used” Use Cases: | 1. Calculate Total Expenses 2. Calculate Total Savings |
| Extension Points: | Open Report |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Delete Expense |
| Summary: | Delete an expense amount |
| Version: | 1.0 |
| Preconditions: | 1. System displays the monthly budget table that contains edit text fields 2. System displays the total expenses 3. System displays the total savings 4. System displays the “Save” button 5. System displays the back button |
| Triggers: | User selects the “Open budget” link |
| Main Success Scenario: | 1. System re-calculates the total expenses 2. System re-calculates the total savings 3. System replaces the expense amounts as “null” or “0” where the user put 0 in one or more edit text edit fields 4. System stores the changes into the database |
| Alternative Success Scenarios: |  |
| Post Conditions: | 1. System displays the monthly budget table that contains edit text fields 2. System re-calculates the total expenses 3. System re-calculates the total savings 4. System displays the total expenses 5. System displays the total savings 6. System displays the “Save” button 7. System displays the back button |
| “Used” Use Cases: | 1. Calculate Total Expenses 2. Calculate Total Savings |
| Extension Points: | Open Report |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Add Expense |
| Summary: | Add an expense amount |
| Version: | 1.0 |
| Preconditions: | 1. System displays the monthly budget table that contains edit text fields 2. System displays the total expenses 3. System displays the total savings 4. System displays the “Save” button 5. System displays the back button |
| Triggers: | User selects the “Open budget” link |
| Main Success Scenario: | 1. System re-calculates the total expenses 2. System re-calculates the total savings 3. System stores the changes into the database |
| Alternative Success Scenarios: |  |
| Post Conditions: | 1. System displays the monthly budget table that contains edit text fields 2. System re-calculates the total expenses 3. System re-calculates the total savings 4. System displays the total expenses 5. System displays the total savings 6. System displays the “Save” button 7. System displays the back button |
| “Used” Use Cases: | 1. Calculate Total Expenses 2. Calculate Total Savings |
| Extension Points: | 1. Open Report 2. Create Report |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Add Income |
| Summary: | Add the monthly income amount |
| Version: | 1.0 |
| Preconditions: | 1. System displays the monthly budget table that contains edit text fields 2. System displays the total expenses 3. System displays the total savings 4. System displays the “Save” button 5. System displays the back button |
| Triggers: | User selects the “Open budget” link |
| Main Success Scenario: | 1. System calculates the total expenses 2. System calculates the total savings 3. System stores the changes into the database |
| Alternative Success Scenarios: |  |
| Post Conditions: | 1. System displays the monthly budget table that contains edit text fields 2. System displays the total expenses 3. System displays the total savings 4. System displays the “Save” button 5. System displays the back button |
| “Used” Use Cases: | 1. Calculate Total Expenses 2. Calculate Total Savings |
| Extension Points: | 1. Open Report 2. Create Report |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Calculate Total Expenses |
| Summary: | Calculate sum of the total expenses |
| Version: | 1.0 |
| Preconditions: | 1. System displays the monthly budget table that contains edit text fields 2. System displays the total expenses 3. System displays the total savings 4. System displays the “Save” button 5. System displays the back button |
| Triggers: | User enters numbers in the edit text fields |
| Main Success Scenario: | 1. System calculates the total expenses |
| Alternative Success Scenarios: |  |
| Post Conditions: | 1. System displays the total expenses amount |
| “Used” Use Cases: |  |
| Extension Points: |  |
| Author: | Brian Perry |
| Date: | December 16, 2016 |

|  |  |
| --- | --- |
| Name: | Calculate Savings |
| Summary: | Calculate the total savings |
| Version: | 1.0 |
| Preconditions: | 1. System displays the monthly budget table that contains edit text fields 2. System displays the total expenses 3. System displays the total savings 4. System displays the “Save” button 5. System displays the back button |
| Triggers: | User enters numbers in the edit text fields |
| Main Success Scenario: | 1. System calculates the total savings |
| Alternative Success Scenarios: | System displays the total savings amount |
| Post Conditions: |  |
| “Used” Use Cases: |  |
| Extension Points: |  |
| Author: | Brian Perry |
| Date: | December 16, 2016 |