Detailed Use Case Descriptions

**Name:** Login.

**Description:** A user is logged onto the system through their own account.

**Actor:** User.

**Pre-Conditions:** Has the user provided valid credentials.

**Flow:**

1. User is given option to login or create an account.

**Alternative Flow:**

1. User has provided incorrect information, an error message is shown that asks the user to try logon again.

**Name**: Logged in User

**Description:** A logged on user can logout of their account or they can perform transactions using their account.

**Actor:** Logged in User.

**Pre-Conditions:**

1. When the user is making a withdrawal the must have enough money in their account for the withdrawal or the withdrawal must be in the acceptable limits if the account can have an overdraft.
2. When the user is doing a transfer they must include the correct bank details of the account the money is being transferred to. Only international accounts can do international transfers.

**Flow:**

1. Withdraw money from account and debit the account accordingly.
2. Make a deposit and credit the account accordingly.
3. Transfer money so that the users account is debited and the receiving account is credited.

**Post Conditions:**

**Post-Conditions:**

1. Debit an account after a withdrawal.
2. Credit an account after a deposit
3. Debit and credit an account accordingly when a transfer is completed.

**Name:** Banker

**Description:** Approves or disapproves “unusual” account activity.

**Flow:**

1. Receive notification about unusual account activity detailing activity.
2. Approve or disapprove questionable transactions.
3. Forward notification to Bank Manager

**Name:** Bank Manager

**Description:** Pulls account reports and Approves or disapproves “unusual” account activity.

**Flow:**

1. Receive notification about unusual account activity detailing activity.
2. Approve or disapprove questionable transactions.
3. Pull reports on different accounts and account types.

**Name**: ATM User

**Description:** A user enters their pin and is then able to complete transactions.

**Actor:** ATM User.

**Pre-Conditions:**

1. The user must enter a valid pin before being able to complete transactions.
2. When the user is making a withdrawal the must have enough money in their account for the withdrawal or the withdrawal must be in the acceptable limits if the account can have an overdraft.

**Flow:**

1. Withdraw money from account and debit the account accordingly.
2. Make a deposit and credit the account accordingly.
3. Show the user the balance on their account.
4. Print a receipt of any actions completed.

**Post-Conditions:**

1. Debit an account after a withdrawal.

2. Credit an account after a deposit

**Name:** Session Manager

**Description:** A Class that records all the actions that were completed by a user when they are logged into the system.

**Flow:**

1. Creates a SessionID when a new session starts.
2. Each time an action is completed this is recorded in a session log.
3. Once the user logs out the session ends and all completed actions are stored to a database.