Nick Larsen

Paul Mpanga

Team 6

SEIS 635

Final Project

**Project Description**:

Our proposal for a project is to create a money transfer application. The application will allow users to create a login with password, deposit money, withdraw money, check the balance of their account, and transfer money to another person's account. We plan to have the application be command based, unless we have time to learn about GUIs. We will use Java as our language, Eclipse as our IDE, and Github as our repository.

**Use Case UC1:** Transfer Money

**Scope:** Cash Transfer System

**Level:** user goal

**Primary Actor:** User

**Stakeholders and Interests:**

* User: Wants fast, easy way to transfer money. Wants to be confident that info will not be lost.
* Receiving user (recipient): Wants money transferred to him/her to be accurate and quick.

**Preconditions:** User has created an account, started system, and logged in.

**Success Guarantee:** System accurately captures and retains the money transfer.

**Main Success Scenario:**

1. System asks user to choose between transfer money, deposit money, withdraw money, and check balance.
2. User selects transfer money and system asks user to enter the amount to transfer.
3. User enters an amount and system checks that the amount does not exceed the account balance and notifies user that it does not exceed.
4. System asks user if to choose between recipient name, recipient email, or recipient account number as a method to identify the recipient.
5. User chooses recipient email and system prompts user to enter the email address of the recipient.
6. User enters email address and system checks that the email address is valid.
7. System notifies user that email address is valid.
8. System asks user to enter an optional note and user enters.
9. System displays the amount, recipient, and note (if applicable) to the user and asks if it is correct and if the user would like to transfer.
10. User chooses yes and the funds are transferred from user’s account to recipient’s account.

**Extensions:**

\*a. At any time, system fails.

1. System restarts.
2. System looks for most recent saved state and returns to it.

3. Amount user entered exceeds the available balance for his/her account.

1. System notifies user that the amount exceeds the available balance, displays the user’s balance.
2. System asks user if he/she would like to enter a different amount or deposit money and user selects different amount.
3. User enters an amount and system checks that the amount does not exceed the account balance and notifies user that it does not exceed.
4. Main flow is resumed.