

### ### Business Requirement Document (BRD)

#### Document Title: **\*\*Money Transfer Web Application Feature\*\***

**\*\*Document Version:\*\*** 1.0

**\*\*Date:\*\*** October 16, 2024

**\*\*Author:\*\*** ABC

**\*\*Reviewed by:\*\*** A

**\*\*Approved by:\*\*** B

### ### 1. **\*\*Project Overview\*\***

The objective of this project is to develop a web-based form for a money transfer feature that allows users to select their account, choose a recipient account, enter an amount to transfer, and proceed with the transaction after entering a valid transaction password. The system will validate the inputs, handle potential errors, and notify the user if the transaction is successful or failed. A refresh option will also be provided to reset the form fields.

### ### 2. **\*\*Purpose\*\***

This document outlines the business and technical requirements for implementing the **\*\*Money Transfer\*\*** feature in a web-based interface. This feature is aimed at enhancing user convenience by providing an easy-to-use money transfer interface. Validation and security checks are implemented to ensure smooth and secure transactions.

### ### 3. **\*\*Scope\*\***

The system should enable users to perform the following:

- Select a source account from which they wish to transfer money.

- Select a recipient account to transfer the money.
- Enter an amount between 1 and 1,000,000.
- Enter a valid transaction password to complete the transaction.
- Handle successful or failed transactions based on password validation.
- Provide a button to reset the form fields if needed.

### ### 4. \*\*Business Requirements\*\*

#### #### 4.1 User Story

As a **bank customer**, I want to transfer money between accounts using a web-based form, with clear validation and feedback for security, so that I can complete transactions easily and securely.

---

### ### 5. \*\*Functional Requirements\*\*

#### #### 5.1 Account Selection

- The user must be able to select the account from which they wish to transfer money.
- A dropdown list will display available account numbers for the user to choose from.

#### #### 5.2 Recipient Account Selection

- The user must be able to select the recipient account.
- A second dropdown list will display available recipient account numbers for the user to choose from.

#### #### 5.3 Entering Transfer Amount

- The user must enter an amount they wish to transfer.
- The input field must allow a range between 1 and 1,000,000.

- If the user enters an amount outside this range, an error message should appear in the form of a popup, stating: "Your amount is not in range."

#### #### 5.4 Validating Transfer Amount

- The system must ensure that the amount entered is numeric.
- If the amount is non-numeric or out of the valid range, the system should prevent the user from proceeding and display an appropriate error message.

#### #### 5.5 Transaction Password Entry

- After the user clicks the **\*\*Send Money\*\*** button, a section should appear for the user to enter their transaction password.
- The transaction password must be validated against the system's predefined password (`123456`).
- If the user enters the correct password, a success message will appear: `Transaction Successful!`
- If the user enters the wrong password, a failure message will appear: `Transaction Failed. Incorrect Password.`

#### #### 5.6 Reset/Refresh Functionality

- A **\*\*Refresh\*\*** button must be provided to reset all form fields.
- Upon clicking the **\*\*Refresh\*\*** button, the form should be cleared, and the transaction password section (if visible) should be hidden.

---

### ### 6. **\*\*Non-Functional Requirements\*\***

#### #### 6.1 Usability

- The system should be intuitive and easy for users to navigate.
- Error messages must be clear and help users understand how to correct issues (e.g., amount out of range or incorrect password).

#### #### 6.2 Performance

- The form should load quickly and respond to user inputs within 1-2 seconds.

#### #### 6.3 Security

- Only valid transaction passwords should allow the user to complete the transaction.
- The system should ensure basic input validation (e.g., amount range validation, password validation) to prevent erroneous or malicious inputs.

#### #### 6.4 Compatibility

- The system must work across modern browsers, including Chrome, Firefox, Safari, and Edge.
- The application must be responsive and accessible on both desktop and mobile browsers.

---

### ### 7. **\*\*User Interface Requirements\*\***

#### #### 7.1 Account Dropdowns

- Two dropdown lists must be present:
  - **\*\*Source Account\*\***: Lists the available accounts from which the user can transfer money.
  - **\*\*Recipient Account\*\***: Lists the available accounts to which the user can transfer money.

#### #### 7.2 Amount Input Field

- A text input field should allow the user to enter the transfer amount.
- The system should validate that the entered amount is numeric and within the range of 1 to 1,000,000.

#### #### 7.3 Send Money Button

- A **Send Money** button should trigger the display of the transaction password section.

#### #### 7.4 Transaction Password Section

- Upon clicking the **Send Money** button, the **Transaction Password** section should appear below.
- The password input should be a masked field to protect the password entry.

#### #### 7.5 Message Display

- A message should appear below the password entry to notify the user whether the transaction was successful or failed.

#### #### 7.6 Refresh Button

- A **Refresh** button should be provided that resets the form fields to their default state and hides any messages or password sections.

---

### ### 8. **Validation and Error Handling**

#### #### 8.1 Amount Validation

- If the user enters an amount less than 1 or greater than 1,000,000, the system will display a popup message: **"Your amount is not in range."**
- The system will not proceed to the password section until the amount is valid.

#### #### 8.2 Password Validation

- If the user enters a valid transaction password (`123456`), a success message will appear.

- If the user enters an invalid password, a failure message will appear.

---

#### ### 9. **\*\*Assumptions\*\***

- The user is already logged into their account.
- The user has access to their account numbers for transferring and receiving funds.
- The valid transaction password is preset to `123456` for the purpose of testing the functionality.

---

#### ### 10. **\*\*Dependencies\*\***

- This feature is dependent on the integration of the front-end with the backend validation of accounts and passwords (beyond this form).
- Browser compatibility should be ensured before release.

---

#### ### 11. **\*\*Acceptance Criteria\*\***

- The form must display all fields as per the design.
- The form must validate the amount and display an error if it's out of range.
- The password section must appear only after valid input has been entered in all prior fields.
- Transaction success or failure messages should appear based on password validation.

- The refresh button must reset all fields and hide the password section.

---

### ### 12. \*\*Sign-Off\*\*

- \*\*Business Stakeholder\*\* : \_\_The Kiran Academy\_\_\_\_\_
- \*\*Project Manager\*\* : \_\_AB\_\_\_\_\_
- \*\*Date\*\* : \_\_\_\_\_16 Oct 2024\_\_\_\_\_

---

This Business Requirement Document (BRD) provides a clear understanding of the feature functionality for the money transfer system. The project team should use this document as a baseline for development, testing, and delivery.

---

Let me know if you'd like to add or adjust anything!