***Module: Customer Registration***

**Step 1: Identify Input Boundaries**

For each field, we identify the boundary values based on the requirements:

1. Name:
   * Minimum: 2 characters
   * Maximum: 50 characters
   * BVA test values: 1, 2, 3, 49, 50, 51 characters
   * Invalid: Empty (0 characters), 1 character, 51+ characters
   * Valid: 2 characters, 50 characters
2. Email:
   * Must be a valid email format (e.g., contains "@" and a domain like ".com")
   * Must be unique (not already registered)
   * BVA test values:
     + Invalid format: Missing "@", missing domain, invalid characters
     + Boundary for length: Test very short valid email (e.g., [a@b.co](mailto:a@b.co)) and very long email (up to system limits, typically 254 characters per RFC 5321)
     + Uniqueness: Test an email already in the system vs. a new one
3. Password:
   * Minimum: 6 characters
   * Maximum: Not specified (assume reasonable system limit, e.g., 128 characters for BVA)
   * BVA test values: 5, 6, 7, 127, 128, 129 characters
   * Invalid: Empty, 5 characters, 129+ characters
   * Valid: 6 characters, 128 characters
4. Initial Deposit (for customers):
   * Minimum: $1
   * Maximum: $10,000
   * Must be numeric
   * BVA test values: $0, $1, $2, $9999, $10000, $10001
   * Invalid: Non-numeric, $0, negative values, $10001+
   * Valid: $1, $10000

**Step 2: Write Test Cases**

Each test case will cover boundary values and include expected outcomes. I’ll assume a customer registration scenario since the system allows new users to register as customers. Test cases will include valid, invalid, and boundary conditions for each field.

**Test Cases for Name Field**

| **Test Case ID** | **Description** | **Input Name** | **Email** | **Password** | **Initial Deposit** | **Expected Result** |
| --- | --- | --- | --- | --- | --- | --- |
| TC\_N1 | Name empty | "" | [test@ex.com](mailto:test@ex.com) | Pass123 | 100 | Error: Name is required |
| TC\_N2 | Name too short (1 char) | "A" | [test@ex.com](mailto:test@ex.com) | Pass123 | 100 | Error: Name must be at least 2 characters |
| TC\_N3 | Name at min boundary (2 chars) | "Ab" | [test@ex.com](mailto:test@ex.com) | Pass123 | 100 | Success: Registration completes |
| TC\_N4 | Name just above min (3 chars) | "Abc" | [test@ex.com](mailto:test@ex.com) | Pass123 | 100 | Success: Registration completes |
| TC\_N5 | Name at max boundary (50 chars) | "A...A" (50 chars) | [test@ex.com](mailto:test@ex.com) | Pass123 | 100 | Success: Registration completes |
| TC\_N6 | Name just above max (51 chars) | "A...A" (51 chars) | [test@ex.com](mailto:test@ex.com) | Pass123 | 100 | Error: Name cannot exceed 50 characters |

**Test Cases for Email Field**

| **Test Case ID** | **Description** | **Input Name** | **Email** | **Password** | **Initial Deposit** | **Expected Result** |
| --- | --- | --- | --- | --- | --- | --- |
| TC\_E1 | Email empty | John Doe | "" | Pass123 | 100 | Error: Email is required |
| TC\_E2 | Invalid email (no @) | John Doe | "test.ex.com" | Pass123 | 100 | Error: Invalid email format |
| TC\_E3 | Invalid email (no domain) | John Doe | "test@" | Pass123 | 100 | Error: Invalid email format |
| TC\_E4 | Valid short email | John Doe | "[a@b.co](mailto:a@b.co)" | Pass123 | 100 | Success: Registration completes (if unique) |
| TC\_E5 | Email already exists | John Doe | "[existing@ex.com](mailto:existing@ex.com)" | Pass123 | 100 | Error: Email already registered |
| TC\_E6 | Long valid email (near 254 chars) | John Doe | "[a...a@longdomain.com](mailto:a...a@longdomain.com)" (254 chars) | Pass123 | 100 | Success: Registration completes (if unique) |

**Test Cases for Password Field**

| **Test Case ID** | **Description** | **Input Name** | **Email** | **Password** | **Initial Deposit** | **Expected Result** |
| --- | --- | --- | --- | --- | --- | --- |
| TC\_P1 | Password empty | John Doe | [test@ex.com](mailto:test@ex.com) | "" | 100 | Error: Password is required |
| TC\_P2 | Password too short (5 chars) | John Doe | [test@ex.com](mailto:test@ex.com) | "Pass1" | 100 | Error: Password must be at least 6 characters |
| TC\_P3 | Password at min boundary (6 chars) | John Doe | [test@ex.com](mailto:test@ex.com) | "Pass12" | 100 | Success: Registration completes |
| TC\_P4 | Password just above min (7 chars) | John Doe | [test@ex.com](mailto:test@ex.com) | "Pass123" | 100 | Success: Registration completes |
| TC\_P5 | Password at max boundary (128 chars) | John Doe | [test@ex.com](mailto:test@ex.com) | "P...3" (128 chars) | 100 | Success: Registration completes |
| TC\_P6 | Password above max (129 chars) | John Doe | [test@ex.com](mailto:test@ex.com) | "P...3" (129 chars) | 100 | Error: Password exceeds maximum length |

**Test Cases for Initial Deposit Field**

| **Test Case ID** | **Description** | **Input Name** | **Email** | **Password** | **Initial Deposit** | **Expected Result** |
| --- | --- | --- | --- | --- | --- | --- |
| TC\_D1 | Deposit empty | John Doe | [test@ex.com](mailto:test@ex.com) | Pass123 | "" | Error: Initial deposit is required |
| TC\_D2 | Deposit non-numeric | John Doe | [test@ex.com](mailto:test@ex.com) | Pass123 | "abc" | Error: Deposit must be numeric |
| TC\_D3 | Deposit below min ($0) | John Doe | [test@ex.com](mailto:test@ex.com) | Pass123 | 0 | Error: Deposit must be at least $1 |
| TC\_D4 | Deposit negative | John Doe | [test@ex.com](mailto:test@ex.com) | Pass123 | -1 | Error: Deposit must be at least $1 |
| TC\_D5 | Deposit at min boundary ($1) | John Doe | [test@ex.com](mailto:test@ex.com) | Pass123 | 1 | Success: Registration completes |
| TC\_D6 | Deposit just above min ($2) | John Doe | [test@ex.com](mailto:test@ex.com) | Pass123 | 2 | Success: Registration completes |
| TC\_D7 | Deposit at max boundary ($10000) | John Doe | [test@ex.com](mailto:test@ex.com) | Pass123 | 10000 | Success: Registration completes |
| TC\_D8 | Deposit above max ($10001) | John Doe | [test@ex.com](mailto:test@ex.com) | Pass123 | 10001 | Error: Deposit cannot exceed $10000 |

***Module: Login***

**Step 1: Identify Input Boundaries**

Based on the requirements, the inputs for the login functionality are:

1. **Email**:
   * Must be in a valid email format (e.g., contains "@" and a domain like ".com").
   * Must match a registered user’s email in the system.
   * Required field.
   * BVA test values:
     + Invalid: Empty, invalid format (e.g., missing "@", missing domain), non-existent email.
     + Valid: Shortest valid email (e.g., "[a@b.co](mailto:a@b.co)") of a registered user, long valid email (up to 254 characters per RFC 5321).
     + Boundary: Test with a registered vs. non-registered email.
2. **Password**:
   * Must be present (non-empty).
   * Must match the stored password for the user.
   * Required field.
   * BVA test values:
     + Invalid: Empty, incorrect password.
     + Valid: Correct password (exact match).
     + Boundary: Test with empty string, single character, and correct password (assuming previous registration rules: minimum 6 characters).
3. **User Type Selection**:
   * Must be selected (required field).
   * Assumed to be a dropdown or similar input with predefined options (e.g., "Customer" as per previous requirements).
   * BVA test values:
     + Invalid: No selection (empty or null).
     + Valid: A valid user type (e.g., "Customer").
     + Boundary: Test with no selection vs. valid selection.
4. **Credential Matching**:
   * Email and password must match a registered user’s stored credentials.
   * Error message: "Invalid email or password" for non-matching credentials.

**Step 2: Write Test Cases**

Each test case will cover boundary values and include expected outcomes for the login process. I’ll assume the system has a registered user with email "[user@ex.com](mailto:user@ex.com)", password "Pass123", and user type "Customer" for testing. The test cases will validate email format, password presence, user type selection, and credential matching.

**Test Cases for Email Field**

| **Test Case ID** | **Description** | **Email** | **Password** | **User Type** | **Expected Result** |
| --- | --- | --- | --- | --- | --- |
| TC\_E1 | Email empty | "" | Pass123 | Customer | Error: Email is required |
| TC\_E2 | Invalid email format (no @) | "user.ex.com" | Pass123 | Customer | Error: Invalid email format |
| TC\_E3 | Invalid email format (no domain) | "user@" | Pass123 | Customer | Error: Invalid email format |
| TC\_E4 | Non-existent email | "[nonexistent@ex.com](mailto:nonexistent@ex.com)" | Pass123 | Customer | Error: "Invalid email or password" |
| TC\_E5 | Valid registered email | "[user@ex.com](mailto:user@ex.com)" | Pass123 | Customer | Success: User logged in |
| TC\_E6 | Valid short email (registered) | "[a@b.co](mailto:a@b.co)" | Pass123 | Customer | Success: User logged in (if registered) |
| TC\_E7 | Long valid email (near 254 chars, registered) | "[a...a@longdomain.com](mailto:a...a@longdomain.com)" (254 chars) | Pass123 | Customer | Success: User logged in (if registered) |

**Test Cases for Password Field**

| **Test Case ID** | **Description** | **Email** | **Password** | **User Type** | **Expected Result** |
| --- | --- | --- | --- | --- | --- |
| TC\_P1 | Password empty | "[user@ex.com](mailto:user@ex.com)" | "" | Customer | Error: Password is required |
| TC\_P2 | Incorrect password | "[user@ex.com](mailto:user@ex.com)" | "Wrong123" | Customer | Error: "Invalid email or password" |
| TC\_P3 | Correct password | "[user@ex.com](mailto:user@ex.com)" | "Pass123" | Customer | Success: User logged in |
| TC\_P4 | Password single character | "[user@ex.com](mailto:user@ex.com)" | "P" | Customer | Error: "Invalid email or password" (assuming min 6 chars from registration) |

**Test Cases for User Type Selection**

| **Test Case ID** | **Description** | **Email** | **Password** | **User Type** | **Expected Result** |
| --- | --- | --- | --- | --- | --- |
| TC\_U1 | User type not selected | "[user@ex.com](mailto:user@ex.com)" | "Pass123" | None | Error: User type is required |
| TC\_U2 | Valid user type | "[user@ex.com](mailto:user@ex.com)" | "Pass123" | Customer | Success: User logged in |
| TC\_U3 | Invalid user type (not in system) | "[user@ex.com](mailto:user@ex.com)" | "Pass123" | Manager | Error: "Invalid email or password" (assuming user not registered as Manager) |

**Test Cases for Credential Matching**

| **Test Case ID** | **Description** | **Email** | **Password** | **User Type** | **Expected Result** |
| --- | --- | --- | --- | --- | --- |
| TC\_C1 | Valid credentials | "[user@ex.com](mailto:user@ex.com)" | "Pass123" | Customer | Success: User logged in |
| TC\_C2 | Invalid email, valid password | "[wrong@ex.com](mailto:wrong@ex.com)" | "Pass123" | Customer | Error: "Invalid email or password" |
| TC\_C3 | Valid email, invalid password | "[user@ex.com](mailto:user@ex.com)" | "Wrong123" | Customer | Error: "Invalid email or password" |
| TC\_C4 | Both email and password invalid | "[wrong@ex.com](mailto:wrong@ex.com)" | "Wrong123" | Customer | Error: "Invalid email or password" |

***Module: Withdraw***

**Step 1: Identify Input Boundaries**

Based on the requirements, the inputs for the withdrawal functionality are:

1. Amount:
   * Must be numeric, positive, and greater than 0.
   * Must not exceed the current account balance.
   * BVA test values (assuming a sample balance for testing, e.g., $1000):
     + Invalid: 0, negative values (e.g., -1), non-numeric (e.g., "abc").
     + Boundary: $0, $0.01, $1, $999.99, $1000, $1000.01 (relative to a $1000 balance).
     + Valid: $0.01 (smallest positive amount), $1000 (equal to balance).
   * Note: Since no maximum limit is specified (other than balance), I’ll use the balance as the upper boundary.
2. Description:
   * Optional, maximum 100 characters.
   * BVA test values:
     + Invalid: 101 characters.
     + Boundary: 0 characters (empty), 1 character, 99 characters, 100 characters.
     + Valid: Empty, 100 characters.
3. Sufficient Funds Validation:
   * Amount must be ≤ current balance.
   * BVA test values (assuming $1000 balance):
     + Boundary: $1000 (equal to balance), $1000.01 (just above balance).
     + Invalid: Any amount > $1000.
     + Error message: "Insufficient funds for this withdrawal" for amounts exceeding balance.
4. Transaction Recording:
   * Must record: Date, type (withdrawal), amount, description, new balance.
   * BVA not directly applicable, but we’ll verify correct recording in test cases.

**Step 2: Write Test Cases**

Each test case will cover boundary values and include expected outcomes for the withdrawal process. I’ll assume a customer account with a balance of $1000 for testing purposes, as no specific balance is provided. The test cases will validate the amount, description, sufficient funds, and transaction recording.

**Test Cases for Amount Field**

| **Test Case ID** | **Description** | **Input Amount** | **Description** | **Current Balance** | **Expected Result** |
| --- | --- | --- | --- | --- | --- |
| TC\_A1 | Amount empty | "" | "Grocery" | 1000 | Error: Amount is required |
| TC\_A2 | Amount non-numeric | "abc" | "Grocery" | 1000 | Error: Amount must be numeric |
| TC\_A3 | Amount negative | -1 | "Grocery" | 1000 | Error: Amount must be greater than 0 |
| TC\_A4 | Amount zero | 0 | "Grocery" | 1000 | Error: Amount must be greater than 0 |
| TC\_A5 | Amount just above zero | 0.01 | "Grocery" | 1000 | Success: Withdrawal completes, new balance = $999.99, transaction recorded (Date, Withdrawal, $0.01, "Grocery", $999.99) |
| TC\_A6 | Amount at balance boundary | 1000 | "Grocery" | 1000 | Success: Withdrawal completes, new balance = $0, transaction recorded (Date, Withdrawal, $1000, "Grocery", $0) |
| TC\_A7 | Amount just above balance | 1000.01 | "Grocery" | 1000 | Error: "Insufficient funds for this withdrawal" |

**Test Cases for Description Field**

| **Test Case ID** | **Description** | **Input Amount** | **Description** | **Current Balance** | **Expected Result** |
| --- | --- | --- | --- | --- | --- |
| TC\_D1 | Description empty | 100 | "" | 1000 | Success: Withdrawal completes, new balance = $900, transaction recorded (Date, Withdrawal, $100, "", $900) |
| TC\_D2 | Description 1 character | 100 | "A" | 1000 | Success: Withdrawal completes, new balance = $900, transaction recorded (Date, Withdrawal, $100, "A", $900) |
| TC\_D3 | Description at max boundary (100 chars) | 100 | "A...A" (100 chars) | 1000 | Success: Withdrawal completes, new balance = $900, transaction recorded (Date, Withdrawal, $100, "A...A", $900) |
| TC\_D4 | Description above max (101 chars) | 100 | "A...A" (101 chars) | 1000 | Error: Description cannot exceed 100 characters |

**Test Cases for Sufficient Funds and Transaction Recording**

| **Test Case ID** | **Description** | **Input Amount** | **Description** | **Current Balance** | **Expected Result** |
| --- | --- | --- | --- | --- | --- |
| TC\_S1 | Insufficient funds (just above balance) | 1000.01 | "Grocery" | 1000 | Error: "Insufficient funds for this withdrawal" |
| TC\_S2 | Exact balance withdrawal | 1000 | "Full withdrawal" | 1000 | Success: Withdrawal completes, new balance = $0, transaction recorded (Date, Withdrawal, $1000, "Full withdrawal", $0) |
| TC\_S3 | Partial withdrawal | 500 | "Partial withdrawal" | 1000 | Success: Withdrawal completes, new balance = $500, transaction recorded (Date, Withdrawal, $500, "Partial withdrawal", $500) |
| TC\_S4 | Zero balance withdrawal | 1 | "Grocery" | 0 | Error: "Insufficient funds for this withdrawal" |

***Module: Transfer Funds***

**Step 1: Identify Input Boundaries**

Based on the requirements, the inputs for the transfer functionality are:

1. **Recipient's Email Address**:
   * Must be a valid email format (e.g., contains "@" and a domain like ".com").
   * Must belong to an existing customer (not the sender).
   * Must not be the sender’s own email (self-transfer prevention).
   * BVA test values:
     + Invalid: Empty, invalid format (e.g., missing "@", missing domain), non-existent email, sender’s own email.
     + Valid: Shortest valid email (e.g., "[a@b.co](mailto:a@b.co)") of an existing customer, long valid email (up to 254 characters per RFC 5321).
     + Boundary: Test with an email just registered vs. non-existent.
2. **Amount**:
   * Must be numeric, positive, and greater than 0.
   * Must not exceed the current account balance.
   * BVA test values (assuming a sample balance of $1000):
     + Invalid: 0, negative values (e.g., -1), non-numeric (e.g., "abc").
     + Boundary: $0, $0.01, $1, $999.99, $1000, $1000.01 (relative to a $1000 balance).
     + Valid: $0.01 (smallest positive amount), $1000 (equal to balance).
3. **Description**:
   * Optional, maximum 100 characters.
   * BVA test values:
     + Invalid: 101 characters.
     + Boundary: 0 characters (empty), 1 character, 99 characters, 100 characters.
     + Valid: Empty, 100 characters.
4. **Sufficient Funds Validation**:
   * Amount must be ≤ current balance.
   * BVA test values (assuming $1000 balance):
     + Boundary: $1000 (equal to balance), $1000.01 (just above balance).
     + Invalid: Any amount > $1000.
     + Error message: "Insufficient funds for this transfer" for amounts exceeding balance.
5. **Self-Transfer Validation**:
   * Prevent transfers to the sender’s own account.
   * Test: Sender’s email vs. another customer’s email.
   * Error message: "You cannot transfer money to yourself" for self-transfer.

**Step 2: Write Test Cases**

Each test case will cover boundary values and include expected outcomes for the transfer process. I’ll assume the sender’s account has a balance of $1000 and their email is "[sender@ex.com](mailto:sender@ex.com)". The test cases will validate the recipient’s email, amount, description, sufficient funds, and self-transfer prevention. Transaction recording (e.g., date, type, amount, description, new balance) is implied but not explicitly tested unless specified.

**Test Cases for Recipient's Email Address**

| **Test Case ID** | **Description** | **Sender Email** | **Recipient Email** | **Amount** | **Description** | **Sender Balance** | **Expected Result** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TC\_R1 | Email empty | [sender@ex.com](mailto:sender@ex.com) | "" | 100 | "Transfer" | 1000 | Error: Recipient email is required |
| TC\_R2 | Invalid email format (no @) | [sender@ex.com](mailto:sender@ex.com) | "invalid.ex.com" | 100 | "Transfer" | 1000 | Error: "Recipient not found" |
| TC\_R3 | Invalid email format (no domain) | [sender@ex.com](mailto:sender@ex.com) | "invalid@" | 100 | "Transfer" | 1000 | Error: "Recipient not found" |
| TC\_R4 | Non-existent email | [sender@ex.com](mailto:sender@ex.com) | "[nonexistent@ex.com](mailto:nonexistent@ex.com)" | 100 | "Transfer" | 1000 | Error: "Recipient not found" |
| TC\_R5 | Self-transfer | [sender@ex.com](mailto:sender@ex.com) | "[sender@ex.com](mailto:sender@ex.com)" | 100 | "Transfer" | 1000 | Error: "You cannot transfer money to yourself" |
| TC\_R6 | Valid short email (existing customer) | [sender@ex.com](mailto:sender@ex.com) | "[a@b.co](mailto:a@b.co)" | 100 | "Transfer" | 1000 | Success: Transfer completes, sender balance = $900, recipient balance updated |
| TC\_R7 | Long valid email (existing customer, near 254 chars) | [sender@ex.com](mailto:sender@ex.com) | "[a...a@longdomain.com](mailto:a...a@longdomain.com)" (254 chars) | 100 | "Transfer" | 1000 | Success: Transfer completes, sender balance = $900, recipient balance updated |

**Test Cases for Amount Field**

| **Test Case ID** | **Description** | **Sender Email** | **Recipient Email** | **Amount** | **Description** | **Sender Balance** | **Expected Result** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TC\_A1 | Amount empty | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | "" | "Transfer" | 1000 | Error: Amount is required |
| TC\_A2 | Amount non-numeric | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | "abc" | "Transfer" | 1000 | Error: Amount must be numeric |
| TC\_A3 | Amount negative | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | -1 | "Transfer" | 1000 | Error: Amount must be greater than 0 |
| TC\_A4 | Amount zero | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 0 | "Transfer" | 1000 | Error: Amount must be greater than 0 |
| TC\_A5 | Amount just above zero | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 0.01 | "Transfer" | 1000 | Success: Transfer completes, sender balance = $999.99, recipient balance updated |
| TC\_A6 | Amount at balance boundary | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 1000 | "Transfer" | 1000 | Success: Transfer completes, sender balance = $0, recipient balance updated |
| TC\_A7 | Amount just above balance | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 1000.01 | "Transfer" | 1000 | Error: "Insufficient funds for this transfer" |

**Test Cases for Description Field**

| **Test Case ID** | **Description** | **Sender Email** | **Recipient Email** | **Amount** | **Description** | **Sender Balance** | **Expected Result** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TC\_D1 | Description empty | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 100 | "" | 1000 | Success: Transfer completes, sender balance = $900, recipient balance updated |
| TC\_D2 | Description 1 character | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 100 | "T" | 1000 | Success: Transfer completes, sender balance = $900, recipient balance updated |
| TC\_D3 | Description at max boundary (100 chars) | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 100 | "T...T" (100 chars) | 1000 | Success: Transfer completes, sender balance = $900, recipient balance updated |
| TC\_D4 | Description above max (101 chars) | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 100 | "T...T" (101 chars) | 1000 | Error: Description cannot exceed 100 characters |

**Test Cases for Sufficient Funds and Self-Transfer**

| **Test Case ID** | **Description** | **Sender Email** | **Recipient Email** | **Amount** | **Description** | **Sender Balance** | **Expected Result** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TC\_S1 | Insufficient funds (just above balance) | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 1000.01 | "Transfer" | 1000 | Error: "Insufficient funds for this transfer" |
| TC\_S2 | Exact balance transfer | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 1000 | "Full transfer" | 1000 | Success: Transfer completes, sender balance = $0, recipient balance updated |
| TC\_S3 | Partial transfer | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 500 | "Partial transfer" | 1000 | Success: Transfer completes, sender balance = $500, recipient balance updated |
| TC\_S4 | Zero balance transfer | [sender@ex.com](mailto:sender@ex.com) | [recipient@ex.com](mailto:recipient@ex.com) | 1 | "Transfer" | 0 | Error: "Insufficient funds for this transfer" |
| TC\_S5 | Self-transfer prevention | [sender@ex.com](mailto:sender@ex.com) | [sender@ex.com](mailto:sender@ex.com) | 100 | "Self transfer" | 1000 | Error: "You cannot transfer money to yourself" |