UiBank API test scenarios

**1. Banking Operations Collection (Session-Based Services)**

These services require users to log in first.

* **1.1 User Login Service**
  1. **Scenario:** Successful login with valid credentials
     1. **Pre-requisite details:** Valid username and password of a registered user.
     2. **Step:** Enter the valid username and password in the login form and submit.
     3. **Expected Result:** The user should be successfully logged in, and a session token should be generated.
     4. **TC Category:** End-to-end
  2. **Scenario:** Failed login with invalid username
     1. **Pre-requisite details:** Invalid username and valid password.
     2. **Step:** Enter the invalid username and valid password in the login form and submit.
     3. **Expected Result:** The system should reject the login attempt and display an error message indicating an invalid username.
     4. **TC Category:** End-to-end
  3. **Scenario:** Failed login with invalid password
     1. **Pre-requisite details:** Valid username and invalid password.
     2. **Step:** Enter the valid username and invalid password in the login form and submit.
     3. **Expected Result:** The system should reject the login attempt and display an error message indicating an invalid password.
     4. **TC Category:** End-to-end
  4. **Scenario:** Failed login with missing information
     1. **Pre-requisite details:** Missing username or password.
     2. **Step:** Attempt to log in without entering a username or password.
     3. **Expected Result:** The system should display an error message prompting the user to enter all required information.
     4. **TC Category:** End-to-end
  5. **Scenario:** Session token generation and validation
     1. **Pre-requisite details:** Successful login.
     2. **Step:** After successful login, capture the session token and use it for subsequent API calls.
     3. **Expected Result:** The session token should be successfully generated and validated for accessing other banking services.
     4. **TC Category:** End-to-end
* **1.2 Get Profile Service**
  1. **Scenario:** Retrieve profile with valid session
     1. **Pre-requisite details:** Valid session token.
     2. **Step:** Use the valid session token to request the user's profile information.
     3. **Expected Result:** The system should return the correct customer information associated with the logged-in user.
     4. **TC Category:** End-to-end
  2. **Scenario:** Reject access with invalid session
     1. **Pre-requisite details:** Invalid or expired session token.
     2. **Step:** Use an invalid or expired session token to request the user's profile information.
     3. **Expected Result:** The system should reject the request and return an error message indicating an invalid session.
     4. **TC Category:** End-to-end
  3. **Scenario:** Return correct customer information
     1. **Pre-requisite details:** Valid session token.
     2. **Step:** Use the valid session token to request the user's profile information.
     3. **Expected Result:** The system should return accurate and complete customer details.
     4. **TC Category:** End-to-end
  4. **Scenario:** Protect customer privacy
     1. **Pre-requisite details:** Valid session token.
     2. **Step:** Attempt to access another user's profile information using the current user's session token.
     3. **Expected Result:** The system should prevent access to other users' information and return an appropriate error message.
     4. **TC Category:** End-to-end
* **1.3 Create Account Service**

1. **Scenario:** Create savings account successfully
   * 1. **Pre-requisite details:** Valid session token.
     2. **Step:** Use the API to create a new savings account.
     3. **Expected Result:** A new savings account should be created successfully and linked to the customer's profile.
     4. **TC Category:** End-to-end
2. **Scenario:** Create checking account successfully
   * 1. **Pre-requisite details:** Valid session token.
     2. **Step:** Use the API to create a new checking account.
     3. **Expected Result:** A new checking account should be created successfully and linked to the customer's profile.
     4. **TC Category:** End-to-end
3. **Scenario:** Create account with initial balance
   * 1. **Pre-requisite details:** Valid session token.
     2. **Step:** Use the API to create a new account with an initial balance.
     3. **Expected Result:** The new account should be created with the specified initial balance.
     4. **TC Category:** End-to-end
4. **Scenario:** Reject creation with invalid session
   * 1. **Pre-requisite details:** Invalid session token.
     2. **Step:** Attempt to create a new account using an invalid session token.
     3. **Expected Result:** The system should reject the request and return an error message indicating an invalid session.
     4. **TC Category:** End-to-end
5. **Scenario:** Handle duplicate account creation attempts
   * 1. **Pre-requisite details:** Valid session token.
     2. **Step:** Attempt to create an account with the same details as an existing account.
     3. **Expected Result:** The system should handle the duplicate account creation attempt and return an appropriate error message.
     4. **TC Category:** End-to-end

* **1.4 Get Account Service**
  1. **Scenario:** Get all accounts for valid customer
     1. **Pre-requisite details:** Valid session token.
     2. **Step:** Use the API to retrieve all accounts associated with the logged-in customer.
     3. **Expected Result:** The system should return a list of all accounts belonging to the customer.
     4. **TC Category:** End-to-end
  2. **Scenario:** Return empty list for customer with no accounts
     1. **Pre-requisite details:** Valid session token for a customer with no accounts.
     2. **Step:** Use the API to retrieve all accounts for a customer who has no accounts.
     3. **Expected Result:** The system should return an empty list or an appropriate message indicating that the customer has no accounts.
     4. **TC Category:** End-to-end
  3. **Scenario:** Reject access with invalid session
     1. **Pre-requisite details:** Invalid session token.
     2. **Step:** Attempt to retrieve accounts using an invalid session token.
     3. **Expected Result:** The system should reject the request and return an error message indicating an invalid session.
     4. **TC Category:** End-to-end
  4. **Scenario:** Show accurate balance information
     1. **Pre-requisite details:** Valid session token.
     2. **Step:** Use the API to retrieve account details and verify the balance information.
     3. **Expected Result:** The system should display accurate and up-to-date balance information for each account.
     4. **TC Category:** End-to-end
* **1.5 Transfer Funds Service**
  1. **Scenario:** Successful transfer between customer's accounts
     1. **Pre-requisite details:** Valid session token, source and destination accounts with sufficient balance in the source account.
     2. **Step:** Use the API to transfer funds between two accounts belonging to the same customer.
     3. **Expected Result:** The funds should be successfully transferred, and the balances of both accounts should be updated accordingly.
     4. **TC Category:** End-to-end
  2. **Scenario:** Reject transfer with insufficient funds
     1. **Pre-requisite details:** Valid session token, source and destination accounts, but the source account has insufficient funds.
     2. **Step:** Attempt to transfer funds from an account with insufficient funds.
     3. **Expected Result:** The system should reject the transfer and return an error message indicating insufficient funds.
     4. **TC Category:** End-to-end
  3. **Scenario:** Reject transfer to non-existent account
     1. **Pre-requisite details:** Valid session token, valid source account, and a non-existent destination account.
     2. **Step:** Attempt to transfer funds to a non-existent account.
     3. **Expected Result:** The system should reject the transfer and return an error message indicating that the destination account is invalid.
     4. **TC Category:** End-to-end
  4. **Scenario:** Reject transfer with invalid session
     1. **Pre-requisite details:** Invalid session token.
     2. **Step:** Attempt to transfer funds using an invalid session token.
     3. **Expected Result:** The system should reject the request and return an error message indicating an invalid session.
     4. **TC Category:** End-to-end
  5. **Scenario:** Verify balance updates after transfer
     1. **Pre-requisite details:** Valid session token, successful fund transfer.
     2. **Step:** After a successful transfer, verify the balance updates in both the source and destination accounts.
     3. **Expected Result:** The balances of both accounts should be updated correctly to reflect the transfer.
     4. **TC Category:** End-to-end

**2. Loan Services Collection (Public Services)**

These services don't require login.

* **2.1 Apply for Loan Service**
  1. **Scenario:** Approve loan for qualified applicant
     1. **Pre-requisite details:** Loan application details that meet the loan approval criteria.
     2. **Step:** Submit a loan application with qualifying details.
     3. **Expected Result:** The loan application should be approved, and a unique loan ID should be generated.
     4. **TC Category:** End-to-end
  2. **Scenario:** Deny loan for unqualified applicant
     1. **Pre-requisite details:** Loan application details that do not meet the loan approval criteria.
     2. **Step:** Submit a loan application with non-qualifying details.
     3. **Expected Result:** The loan application should be denied, and the system should provide a reason for the denial.
     4. **TC Category:** End-to-end
  3. **Scenario:** Handle missing application information
     1. **Pre-requisite details:** Loan application with missing required information.
     2. **Step:** Submit a loan application with missing details.
     3. **Expected Result:** The system should reject the application and prompt the user to provide the missing information.
     4. **TC Category:** End-to-end
  4. **Scenario:** Generate unique loan ID for approved loans
     1. **Pre-requisite details:** Approved loan application.
     2. **Step:** Verify that a unique loan ID is generated upon approval.
     3. **Expected Result:** A unique loan ID should be generated for each approved loan.
     4. **TC Category:** End-to-end
  5. **Scenario:** Validate input data ranges
     1. **Pre-requisite details:** Loan application with out-of-range input values.
     2. **Step:** Submit a loan application with invalid input values (e.g., negative loan amount).
     3. **Expected Result:** The system should reject the application and provide an error message indicating the invalid input.
     4. **TC Category:** End-to-end
* **2.2 Get Loan Details Service**
  1. **Scenario:** Retrieve loan information with valid loan ID
     1. **Pre-requisite details:** Valid loan ID.
     2. **Step:** Use the API to retrieve loan details using a valid loan ID.
     3. **Expected Result:** The system should return the complete and accurate details for the specified loan.
     4. **TC Category:** End-to-end
  2. **Scenario:** Handle requests with invalid loan ID
     1. **Pre-requisite details:** Invalid loan ID.
     2. **Step:** Attempt to retrieve loan details using an invalid loan ID.
     3. **Expected Result:** The system should reject the request and return an error message indicating an invalid loan ID.
     4. **TC Category:** End-to-end
  3. **Scenario:** Return complete and accurate loan details
     1. **Pre-requisite details:** Valid loan ID.
     2. **Step:** Use the API to retrieve loan details and verify the accuracy of the returned information.
     3. **Expected Result:** The system should return complete and accurate loan details.
     4. **TC Category:** End-to-end
  4. **Scenario:** Protect loan information privacy
     1. **Pre-requisite details:** Valid loan ID.
     2. **Step:** Attempt to access loan information without a valid loan ID or with an ID that doesn't belong to the current user.
     3. **Expected Result:** The system should prevent unauthorized access to loan information.
     4. **TC Category:** End-to-end