Deliverables

Deliverable 1: Create Accounts table and setup

Sub-Task 1: Add a screenshot from the db of the system user (Users table)

(Clearly show/caption which is the system user (should not be a real user, password should not be a valid password/hash))

Sub-Task 2: Add a screenshot from the db of the world account (Accounts table)

(Clearly show/caption which is the world account (should have account number 000000000000, should be of type world, should be associated with the system user))

Sub-Task 3: Explain the purpose and usage of these two entries and how they relate

(explain why they're needed for transactions)

Deliverable 2: Dashboard

Sub-Task 1: Add a screenshot showing the requested links/navigation

Links in the navbar don't count (it's ok if some are duplicated here and navbar). Ensure the screenshot shows the heroku dev url

Deliverable 3: Create a checking Account

Sub-Task 1: Add a screenshot showing the Create Account Page. (Have valid date filled in) (Ensure the screenshot shows the heroku dev URL)

Sub-Task 2: Add screenshots showing validation errors and success message. (Show the minimum funding validation) (Show the success message from task 1's data) (Ensure the screenshot shows the heroku dev URL)

Sub-Task 3: Add a screenshot showing the transaction generated from the initial deposit (from the DB) (Clearly highlight or mention in the caption which to look at. There should be two records showing the positive and negative movement of funds between accounts)

Sub-Task 4: Explain which account number generation you used and the account creation process including the transaction logic. (Clearly mention the option number and the text from the proposal).( Clearly explain the full steps of creating an account including the association to the user and the transactions being generated (at a high level as they're discussed in detail later))

Deliverable 4: User will be able to list their accountsSub-Task 1: Add a screenshot showing the user's account list view (show 5 accounts) (There should be at least 5) (Show account number, account type, modified, and balance)(Ensure the screenshot shows the heroku dev URL)

Sub-Task 2: Briefly explain how the page is displayed and the data lookup occurs (Explain the database call and the data used to get the data and how it's displayed on the webpage)

Deliverable 5: Account Transaction Details

Sub-Task 1: Add a screenshot of an account's transaction history.(should have at least a few samples)(Account: Show account number, account type, balance, opened/created date of the selected account (from Accounts table))(Transactions: Show the src/dest account numbers (not account id), the transaction type, the change in balance, when it occurred, expected total, and the memo) (Ensure the screenshot shows the heroku dev URL)

Sub-Task 2: Explain how the lookup and display occurs (Describe the db query and the data used to fetch the necessary results and anything else relevant)

Deliverable 6: Deposit/Withdraw

Sub-Task 1: Show a Screenshot of the Deposit Page (Show valid data filled in) (Ensure the screenshot shows the heroku dev URL)

Sub-Task 2: Show a Screenshot of the Withdraw Page (this potentially can be the same page with different views)(Ensure the screenshot shows the heroku dev URL) (Page should include a dropdown of logged in user's accounts and a text field to add desired amount to)

Sub-Task 3: Show validation error for negative numbers (Ensure the screenshot shows the heroku dev URL)

Sub-Task 4: Show validation error for withdrawing more than the account contains(Ensure the screenshot shows the heroku dev URL)

Sub-Task 5: Show a success message for deposit and withdraw (2 screenshots) (Shows deposit success)(Shows withdraw success)

Sub-Task 6: Show a screenshot of the transaction pairs in the DB for the above tests (Expected total should be accurate for both accounts)(Highlight or mention in caption which records to look at)

Sub-Task 7: Briefly explain how transactions work (List each step and give some detail on it: Should include account lookup, expected total calculation, two transaction records, and updating the account balances based on the history (not basic math))