Node JS: Weekend Assignment

Node JS – Bank API

The following exercise contains the following subjects:

- express
- fs
- Postman
- React

Submitting instructions:

- Push the full folder hierarchy of the project to your own repository on Github.

Instructions:

- Use MongoAtlas to handle your database.
- Create express endpoints that handle the logic
- Create a front end site for your project with React
- Test your work with Postman

We are going to build a bank API.
You are a manager of a big bank.
The manager has access to the users of the bank and can do the following:

Add users
Can add users to the bank. Each user has the following:

ID, cash (default 0), credit (default 0).

Depositing
Can deposit cash to a user. (by the user's ID and amount of cash)

Update credit
Can update a user's credit (only positive numbers) Withdraw money Can withdraw money from the user (can withdraw money until the cash and credit run out)

Transferring
Can transfer money from one user to another with credit(can transfer money until the cash and credit run out)

Show details of the user
Can fetch all details of a particular user

Show details of all users
Can fetch all details of all the users

Use cases:

1. Cannot add duplicate users
2. When fetching users, make sure they exist.
3. Any other use cases? (hint: there are!)
   If the use cases are not sufficient, send an appropriate error message to the client.

Filter the users

1. Can fetch users by the amount of cash they have.
2. Think of something else to filter.

Deploy
Deploy your project to Cyclic and connect to React

Ninja:
Add a new field for a user: ‘IsActive’.
‘IsActive’ determines if the account is active or not.

1. If the user is not active, you cannot do anything with that user.
2. Fetch the users that are active and have a specified amount of cash.