Group Meetings

Date	Attendees	Details
17/11/20	Daniel, Maxime, Mahsum, Fareese, Amartya	Began by going over the project overview – this consisted of looking at both the functional and non-functional requirements for the project. This allowed us to see which parts of the project to prioritize and how we would be able to split the work to allow for the most efficient team development process.
25/11/20	Daniel, Maxime, Mahsum, Fareese, Amartya	We looked at different use cases for the project so that we can make sure our final application can meet a specification of fulfilling bank users' needs. Individual tasks were assigned based on this, including the development of a home page which would branch onto the other pages required.
29/11/20	Daniel, Maxime, Mahsum, Fareese	Start of developing a home page. Started to create dashboard and went over the structure of all the elements in the dash, as well as other pages that would need to be created. This structure was drawn out and we figured out how this can be done in HTML and Ruby. Bootstrap will be used for some of the design elements of the application. Also mentioned the start of working on our individual parts once the base of the home page is created.
30/11/20	Daniel, Mahsum, Fareese, Amartya	Developed the home page as well as the main structure of the site. User login page created with functionality to create new users and login with the newly created details. Basic account management functionality also created, including changing user details and a 'forgot password' feature to send the user an email containing a link to change password. Allocated some more work to be done on our separate branches which we will then discuss at our next meeting before merging to the main branch.
17/12/20	Daniel, Maxime, Mahsum, Fareese, Amartya	Added additional validation, merged all branches into main. Code Inspection of admin and transaction branch. Improved user interface.
18/12/20	Daniel, Maxime, Mahsum, Fareese, Amartya	Deployed the project online with Heroku. Added to the project report and ER Diagram. Fixed bugs with seeding data on production environment.

Project Overview

Must create a fake bank system to trick a scammer into believing they are accessing a victim's true bank account.

Requirements

The project would include the following pages:

- Login
- Account overview
- Transaction history
- Payments/transfers

Functional requirements

- Enable users to create, edit and delete one or more user profiles.
- Add/manage/remove accounts to user profiles.
- Generate a realistic recent transaction history and history that reflects the account balances.
- Admin area that can be to configure the access credentials of fake-users, the numbers and names of the accounts of each user and the amount of money in each account.

Non-functional requirements

- The login process requires only a single password.
- Application supports multiple currencies (GBP, USD and a third currency)
- Admin superusers can add, edit and delete user accounts.
- The system can generate a random recent transaction history for an end-user.
- Some of the peripheral screens, not essential to the refund scam, have realistic content and a realistic look and feel.

Notes

- Must **not** use the name, logo, look and feel (e.g. bespoke fonts and colours) of a genuine online banking service.
- Must **not** use copyright or trademarked materials of a real organisation.

Deployment

The deployed version should be preconfigured with consumer and administrator accounts. The consumer should have multiple bank accounts with transaction history. Correct access credential to user and admin account should be provided.