LOAN APPLICATION PROJECT DOCUMENT

GROUP 6 - Team Members: Thulile Mbuthuma, Chandralekha Veerath, Sagarika Reddy Thumma, Nirmalarani Pallapu

Roadmap of the Report

INTRODUCTION:

Aims and Objectives:

The aim of our project was to create a full stack website that helps users to apply for a home loan online but along the way we decided that it should cater for small lending businesses (i.e., loan Inc), which issues short term loans. The main idea of our website is for the lender to fill a loan application form online and then submit it to Loan Inc. The owner/authorized manager at Loan Inc. will be able to review the loan applications from the users and choose to decline or grant the loan.

Our main objectives for Loan Inc Website were:

- Be user friendly - consider accessibility, readability, and inclusivity.
- Be easy for users to apply for loans and for the managers to review the applications.
- Seamless integration of information from database, back-end and front-end.

Main use and advantages of using the website:

- It helps Loan Inc. to keep track of their customer's details and they can also access their details easily from the database, and this in turn reduces documentation as the customer's details have been saved for the next time they apply for another loan.
- Users don't need to go apply for the loan at the premise of Loan Inc. They can do it at the comfort of their homes, and this saves them time and money.
- The website is available 24/7 thus the customers always have access to it, which makes it more convenient for them as they can use it whenever they need it. They don't need to wait for business hours to apply for the loan.

BACKGROUND:

When a user clicks on the Loan Inc. website, a login form will pop up and they will be able to login if they already have an account. The login form will ask for their username and password. If they do not have an account, they can click our 'Sign Up' link at the bottom of the page. Once they click it, a Sign-Up form will pop up prompting them to input their details (e.g., Username, email, Mobile Number, password, confirm password) to create an account.

When the account is created, the user will be able to login and the website will direct them to the homepage.

The login page also consists of a link that takes the user to the forgot password page where they can reset their password, if by chance they have forgotten their old password.

For Manager by default, we have set an username and password (username=Admin and password=12345) and once we provide the correct details, the manager will be able to login and the website will direct the manager to the Manager homepage screen. As the Manager's login details are given by default and as they are very secure, the Sign-Up page and Forgot password page will not be applicable for manager's login.

The homepage for users consists of two navigation links, the User Profile link directs them to their profile and the Loan Application link directs them to the loan application page. In the homepage for the manager- the Application Review link directs them to the loan applications review page (where they can see/ review the loan application details of users) and the My Profile link directs them to their profile.

Users and managers are able to log off the website when they click the sign off button at the bottom of the home page.

The profile details page in both manager and user login is used to get the details of both manager and users like age, gender, address etc. and those details will be saved into the database along with their username so that it will be easy to track the person based on the username in case if there is a need.

The user can click the loan application link and they will be directed to the loan application page, where they can fill the form and submit it.

The manager can click the application review link in which all loan applications made by the users will be displayed and then the manager can contact the applicant about the loan decision based on details available.

The loan application information will be saved into the database when users submit it and are taken from the database to display under the Manager Review Applications page.

SPECIFICATIONS AND DESIGN:

Requirements technical and non-technical:

Our website needs to have two types of users, the Applicants (who will apply for the loan) and the Manager (who will review the applications and make a decision).

Our technical requirements are:

For Applicants:

- A signup page for users to create an account with their first name, last name, mobile number, email, username, password, and confirm password with validation.
- A login page for users who have accounts to log into the application.
- A forgot password page where the user can reset their password and change their password if they have forgotten it.
- A loan application page where applicants will be able to fill and submit the loan application form.
- A User profile page for user to save their personal details
- A home page that has a navigation link to the profile and loan application pages.

For Manager:

- A login page for manager to log into the application
- The username and password details for manager login as they are given by default. (Username=Admin and Password=12345)
- A profile page for user to save their personal details
- A review application page for manager to review applications of the users
- A home page that has a navigation link to the profile page and loan application review.

Our Non-technical Requirements were:

Usability:

The website is easy and is very clear even for a non-technical user to use.

Security:

As the manager login details are fixed by default, users cannot have access to other's details and all the details are safely stored into the database.

Performance:

The performance of the website is good for both users to apply for the loan and for the manager to review applications.

Maintainability:

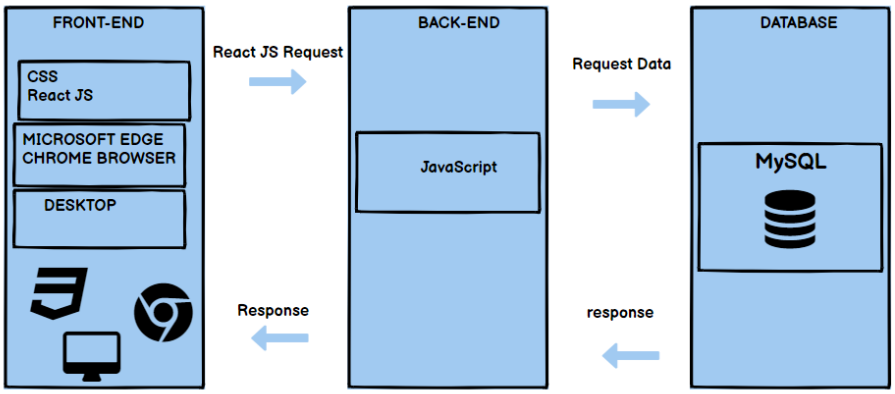
The manager flow and the user's flow are maintained properly throughout the website without any difficulties in understanding the flow.

Design and architecture:

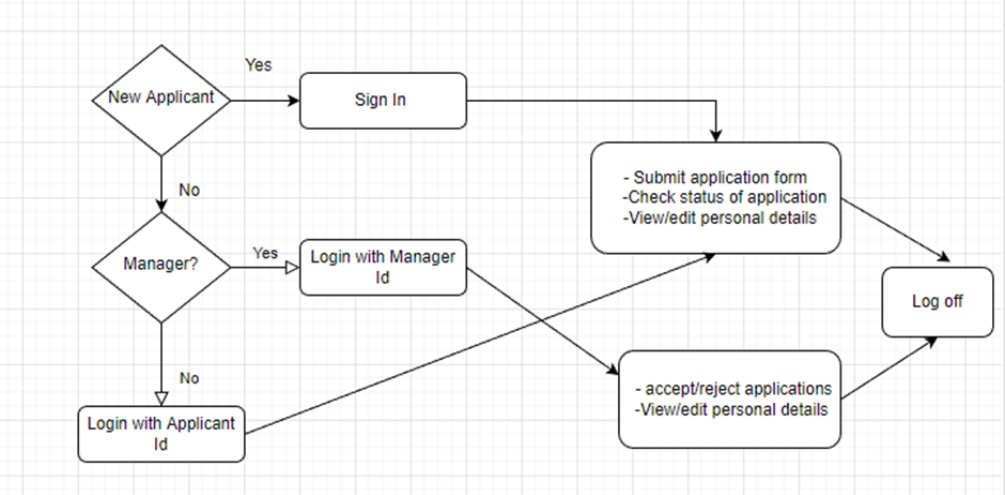
To showcase the structure and design for our system, we drew an architectural diagram and flow chart. Since our Loan application website is a full stack web application, our diagram has been split into three components: front-end, back-end and database.

Loan Inc. website should be able to allow the user to interact with our web app via the front end and back end when they reach our login page. This page prompts the user to input their user credentials (if they already have an account) or to sign up on the bottom of the page.

Once the user inputs their details, you reach the database component where the credentials are automatically stored to MySQL.



We decided to use MySQL to store our data because MySQL is globally renowned for being the most secure and reliable database management system.



IMPLEMENTATION AND EXECUTION:

Development approach and team member roles:

Initially we planned to do daily scrums and assign a scrum master each week. We also planned to work on small sprints (short-term goals) and divide the work between us for each sprint. However, due to lack of sufficient time we had to share the application into different parts and had to work on the shared parts as individuals.

We ended up allocated the tasks accordingly:

- **Thulile Mbuthuma- All the Frontend and documentation**
- **Chandralekha Veerath – All the Backend and the functionalities that integrate front end with backend and database as well as testing**
- **Sagarika Reddy Thumma- Loan application screen/page and presentation slides**
- **Nirmalarani Pallapu – Manager Review Application screen/page**

We ended up having meetings at least 3 times a week just to discuss our progress and to showcase our progress and to help each other with errors and to decide what to do next. We

used the Git repository to push the source code after making changes so the whole team had access to it and continued to work on their parts without disturbing the original flow.

Week 1: Planning

- Project conceptualization (Chandralekha)
- Define all key features (Chandralekha and Thulile)
- Architecture diagram and flow chart (Chandralekha)
- Consolidate project ideas, aims and objectives.

Week 2: Development

- Build database (Chandralekha)
- Connection to database from back-end through api calls (Chandralekha)
- Login and Signup page Frontend (Thulile)
- Functionalities for Login and Signup pages both frontend and backend (Chandralekha)

Week 3: Development

- Forgot password, Profile and Home page Frontend (Thulile)
- Functionalities for Forgot password and profile pages in Frontend and backend (Chandralekha)
- Routing (Thulile)

Week 4: Development

- Validation of pages (Thulile)
- Front end code for Loan Application (Sagarika and Thulile)
- Code for Manager Review page Frontend (Nirmalarani and Thulile)
- Functionalities of both Loan Application and Manager Review Application page in frontend and backend (Chandralekha)
- Testing (Chandralekha)
- Documentation (Thulile and Chandralekha)

Tools and Libraries:

- **Database** - We used MySQL for our database and storing user authentication.
- **Code editor** – We used Visual Studio Code for all code, this included all code to connect the backend to the frontend.
- **Backend Logic** - JavaScript was the coding language used in our project
- **Frontend** - The frontend of our project was created using CSS, React JavaScript
- **Group communication** - Our main methods of communication and collaboration was through emails, slack group and Zoom for all our meetings/coding sessions.

Libraries:

- **React-** For building components to develop UI
- **React Router -** For routing
- **ReactDOM** - It is a complimentary library to React which glues React to the browser DOM.
- **Axios** - For Api requests

Implementation process:

Achievements:

- Able to Sign Up, Login, reset password and save profile details into the database
- Able to submit the loan application form from the user's login
- Able to view the applications of all users under manager review application page

Challenges

- Git

Using Git to work collaboratively proved to be one of our biggest challenges. Within the first two weeks of the project, we were finding it difficult to coordinate branches and work within them, this meant there was confusion with pull requests, and it became hard to track branches and files. In addition, we were constantly finding ourselves creating new repositories which added further confusion. As a result, our collaboration was affected as and can only work on separate tasks independently.

- Duration of the project

We initially took a lot of time in developing small functionalities and had to rush at the end due to lack of time. We were also short of human resources and most tasks were taken by two members as the other members had other obligations (i.e., Works) and couldn't take tasking tasks. Also, the communication between team members was not great and there was a lot of friction which in fact affected the project duration.

- Implementing features

We also experienced issues when we tried to implement certain features we wanted for our application as it required us to constantly attempt to reformat already existing code which affected the connectivity between modules. For example, the Loan application page was coded in a separate react app and we experienced issues with integrating that code to the already existing code of the project (i.e., There were errors). We also had an issue with incorporating validation on this page.

Changes along the way:

1. Loan Application Page

    - The page was initially supposed to be accepting pdf files, but we didn't have enough time for this as we knew that the functionality for it will be complicated and complex and it will take a long time to do.

2. Status Page in User's Login

    - The status page in user's login had to be called off as the status value had to be passed from Review Application Page of Manager login to the Status page of User login which would be very complex and time-consuming.

3. Action Controls in Review Application Page in Manager Login

- As the status page has been removed from the application, we didn't find any meaning in adding action controls in the review application page in manager login.

Agile Development:

- As our project was distributed into individual tasks, we had code reviews with each other's code after each page development and made changes whenever there was a need.
- Also, we improved the functionalities of the existing code during code reviews.

Implementation Challenges:

- During implementation we initially faced difficulties while making the connection between front-end back end as well as to the database.
- However, we were able to tackle the connection difficulties.
- Also had struggled during the CSS properties as it differs in other screens but managed to resolve it finally.

TESTING AND EVALUATION:

- **Testing Strategy:**

  We used ReactTestingLibrary for running our test cases. React Testing Library is a testing utility tool that's built to test the actual DOM tree rendered by React on the browser. With the help of the ReactTestingLibrary we are able to write tests that resemble how UI screens are designed in our application.

- **Functional and User testing:**

  The ReactTestingLibrary library focuses on testing from the end-user's experience rather than testing the implementation and logic of the underlying React components.

  ReactTestingLibrary provides a virtual DOM where we can render components, provides methods to query the DOM for various elements, interact with those elements and make assertions on them.

  We have written most of the test cases with the help of getByText and getByTitle that made our test cases work easily.

- **System limitations:**

  As there are many child components and navigation, the unit test cases are limited only to the user testing.

  The child components are however tested by rendering to BrowserRouter.

CONCLUSION

Overall, we are pleased with the outcomes of our project even if it didn't turn out the way we hoped it would, it was the best we can do with time and human resources we had to work with. We were able to overcome the challenges such as Git issues, integrating our code together and time limitations to produce a decent user friendly full-stack application.

We set out to build an application that solves the problem of having to spend too much time applying for a loan and the complicated loan application process. Our application reduces the paperwork problem for loan applications.

The end products of a website help users to apply for short loans easily and helps managers keep track of applicant's applications. This is a scalable project with an array of possible areas for future development.

Whilst it has been stressful at times, overall, we have enjoyed putting our skills learnt throughout the course to this project. We feel we have been able to practice areas of coding that individually may not have originally been our strength or preferred topic

Moving forward we will look to improve on our understanding of Git/Github, unit testing, teamwork and integrating code as we feel as those are the challenges that hindered our ability to include all of the requirements we stated in the planning stages.