

FRONTEND Implementation Log

6th March-

- Aditya did the initial code setup in react and set the basic layout for the application. Also made the options tab active in the sidebar.
- Pratham set the routing to the pages in the second sidebar and made the layout according to the specifications in the design doc. Also, tokenized the work into components with proper routing to enable group collaboration.

7th March-

- Siddhant completed the 'Profile' page
- Pratham pushed the "Notifications page"

8th March-

- Akshat pushed the 'Vendor' page.
- Kalika pushed the 'Transaction History' page
- Ravija pushed the 'All Dues' page.

9th March-

- Siddhant pushed the 'Registration' page.
- Pratham and Aditya completed routing for vendor-side pages and modified components built on the customer side to meet vendor requirements.

10th March-

- Kalika added functionality to search bar and pushed the changes

11th March-

- Pratham added pages for instant payment and add dues and made links from overview page
- Akshat added the signin page background images.

15th March-

- Aditya wrote the api call to fetch vendor data (with hardcoded customer ID).

16th March-

- Siddhant added the api call to fetch profile data for users (with hardcoded customer ID).
- Pratham added the api call to fetch notification data for users (with hardcoded customer ID).

18th March-

- Aditya added the authentication for login and logout.

19th March-

- Siddhant wrote the api call for the 'update profile' page (customer side)
- Siddhant made some styling changes for the entire application and updated the overview page.

- Siddhant added vendor data, notification, transaction history callback.
- Aditya wrote the vendor transaction data api call.
- Aditya added the api calls for 'All Dues', 'Profile', 'Update Profile' on the vendor side.
- Siddhant added api calls to make instant and add dues transactions (customer side).

BACKEND Implementation log

5th March:

- Harsh and Pulkit wrote the first draft of User and transaction models.

6th March

- Harsh revamped the CustomUser, Vendor, Customer, Wallet and Transaction models, with logic for creation and saving of models.

7th March

- Pulkit implemented the Notification model for sending notifications on new user profile creation (vendor/customer), transaction success and transaction failure.
- Harsh updated the django admin panel for the Transaction, User, and Wallet models.

9th March

- Harsh setup the initial views and urls for API calls in the backend manager.
- Pulkit implemented logic for pending dues functionality
- Monil added the views for the notification functionality as well as started to work on the Issues model.

10th March

- Harsh and Monil added the views for pending dues(showing total dues for each sender-receiver pair)
- Monil implemented an issue model for getting the issues of users regarding any transaction

11th March

- Pulkit improved the Issue() model, implemented an issue resolution system and added notifications for Issues creation / resolution

14th March

- Harsh implemented the User register, login, logout Auth on the DjangoRESTFramework for the CustomUsers in the application.

15th March

- Harsh implemented the User Password Reset in case of the 'forgot password' scenario on the DjangoRESTFramework.

16th March

- Harsh implemented the protection of API routes from unauthorized users to protect data of users
- Harsh implemented the backend service for updating the profile and POSTing transactions.

19th March

- Harsh implemented the clear all dues functionality from the backend (clear dues and clear all dues)
- Harsh added the (since the functionality is abstracted out, this is just a mimic of a banking request) Add Balance to wallet functionality.
- Harsh wrote the backend service to fetch data to be displayed on the Overview Navbar and Table.