

CPSC 304 Project Cover Page

Milestone #: 3

Date: November 1, 2023

Group Number: 82

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Stanley Cheung	32009722	s4k7q	yinstanleycheung@gmail.com
Matias Gauvin	27796267	l6f8n	matiasgauvin@gmail.com
Sunny Nie	59484840	c1p1k	sunnymingnie@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia.

Milestone 3

Due: Nov. 1, 2023 @ 9 AM

Meeting: Nov. 3, 2023 @ 11:40 AM - 11:55 AM

Summary

Our project, tentatively titled “Biddr”, is a combination between a traditional social-media app and a bidding/fundraiser site. A user can create auctions and fundraisers within a particular community, as well as make bids or donations using their wallets. In addition, a user can maintain a community and privately message other users.

Description of challenges / things left to do

We have completed designing the relational model and have fleshed out the idea for our application. What’s left is the actual implementation. These are the main tasks we need to complete:

- Designing the UI of the entire application using Figma
- Building the database in PostgreSQL
- Building the frontend UI using React and ChakraUI
- Creating a backend API with Express.js
- Implementing an authentication system that securely saves passwords and local sessions
- Connecting the frontend with the backend
- Resolving any bugs or glitches
- Deploying the app on a remote server

Timeline

The deadline for the tasks below is the last day of the week they are assigned under. Each task’s assigned person is subject to change.

Week One (Oct. 30 – Nov. 5)

Create the majority of our UI designs. Build our SQL tables. Setup the basic structure and functionality of our backend and frontend apps, and connect them together.

- **Database:**
 - Deploy a remote PostgreSQL database using Render.com (**done**)
 - Create tables and sample tuples for the PostgreSQL database (create SQL DDL Script), in this order:

- LocationAgeOfMajority (**Matias**)
- LocationDateOfBirthLegalAge (**Matias**)
- User (**Matias**)
- Wallet (**Matias**)
- Community (**Matias**)
- Joins (**Matias**)
- Post (**Matias**)
- Auction (**Matias**)
- Fundraiser (**Stanley**)
- Bid (**Stanley**)
- Donation (**Stanley**)
- Comment (**Stanley**)
- Chat (**Stanley**)
- PrivateMessage (**Stanley**)
- EngagedIn (**Stanley**)
- Likes (**Sunny**)
- Sample tuples for all tables (**Sunny**)
- **Backend:**
 - Make test queries to the remote SQL database (**Stanley**)
 - Create a structure for the backend app (**Stanley**)
 - Create basic API endpoints in Express.js for retrieving database data (**Stanley**)
- **Frontend:**
 - Create a React app w/ ChakraUI & Javascript (**Sunny**)
 - Test if we can make successfully call the API endpoints of our backend app (**Sunny**)
- **Figma UI Designs:**
 - Log-in / sign-up pages (**Stanley**)
 - User feed, which is based on the community that the user is currently viewing (**Stanley**)
 - Community page view, one for community managers, another for regular users (**Matias**)
 - "Liked Posts" feed, which shows all the posts a user has liked (**Sunny**)
 - Create post view, for either auction or fundraiser (**Sunny**)
 - Single post view, with comments (**Sunny**)
 - User donate to fundraiser view (**Stanley**)
 - User bid on auction view (**Stanley**)
 - User historic transactions view (**Stanley**)
 - User wallets view (with CRUD functionality) (**Matias**)
 - User profile view (from both user's POV and other users' POV) (**Matias**)
 - Chat view, with the option to add users to the chat and to send a message (**Matias**)
 - User's existing chats view, with the option to create new chats and to select a chat (**Sunny**)

Week Two (Nov. 6 – Nov. 12)

Start creating the core of the app: implement both frontend, backend endpoints/logic, and add documentation of the following features (rough estimate).

- **Stanley:**
 - Log-in / sign-up pages, with authentication
 - User feed, which is based on the community that the user is currently viewing
- **Matias:**
 - Community page view, one for community managers, another for regular users
- **Sunny:**
 - Create post view, for either auction or fundraiser
 - Single post view, with comments

Week Three (Nov. 13 – Nov. 19)

Continue implementing the 'core' of the app: implement both frontend, backend endpoints/logic, and add documentation of the following features (rough estimate).

- **Stanley:**
 - User donate to fundraiser view
 - User bid on auction view
 - User historic transactions view
- **Matias**
 - User wallets view (with CRUD functionality)
 - User profile view (from both user's POV and other users' POV)
 - Chat view, with the option to add users to the chat and to send a message
- **Sunny**
 - User's existing chats view, with the option to create new chats and to select a chat
 - "Liked Posts" feed, which shows all the posts a user has liked

Week Four (Nov. 20 - Nov. 26)

- Get the UI to match Figma designs (**All members work on the components they designed**)
- Test frontend and resolve frontend bugs (**All members**)
- Test backend and resolve backend bugs (**All members**)

Week Five (Nov 26 - Nov 30)

- Write description of the project on GitHub, with demo images and videos (**All members**)
- Deploy project on Render.com (or another free hosting service) (**All members**)