

CitySource

Team No. 12

Team Members: Sydney Combs, Daniel Gonzalez, Jacob Parnell, Nathan Pelletier, Branden Taylor

Project Name: CitySource

Project Synopsis

A platform for public engagement with city development for residents, architects and developers.

Project Description

New construction and architectural projects are undertaken every day, yet there is no efficient or easy way for the populace to obtain information about them. Both future and in-development projects are often mysteries to the public. This not only elicits concern from citizens but also raises difficulties for architects and developers on the project. Being unable to confer with community members prevents developers from receiving feedback about the projects and fully understanding the needs of the community.

Here arises the need for an application that facilitates communication between architects and the general population. Our plan is to create a tool that can be used by architects to present their ideas for future projects to the members of the community. Architects post projects on a map UI where community members may post and have conversations addressing these projects. Community members can also post suggestions about developmental enhancements for the area, such as parks or restaurants. With this tool, companies and cities will be able to better understand public opinion regarding the projects they work on, and citizens will have more influence over community development.

Project Milestones

- First Semester
 - o High level project design - October 2019
 - o UI design - November 2019
 - o Database Setup - December 2019

● Second Semester

- o Back End/Front End Integration - February 2020
- o User Testing: March 2020
- o Final Deployment - April 2020

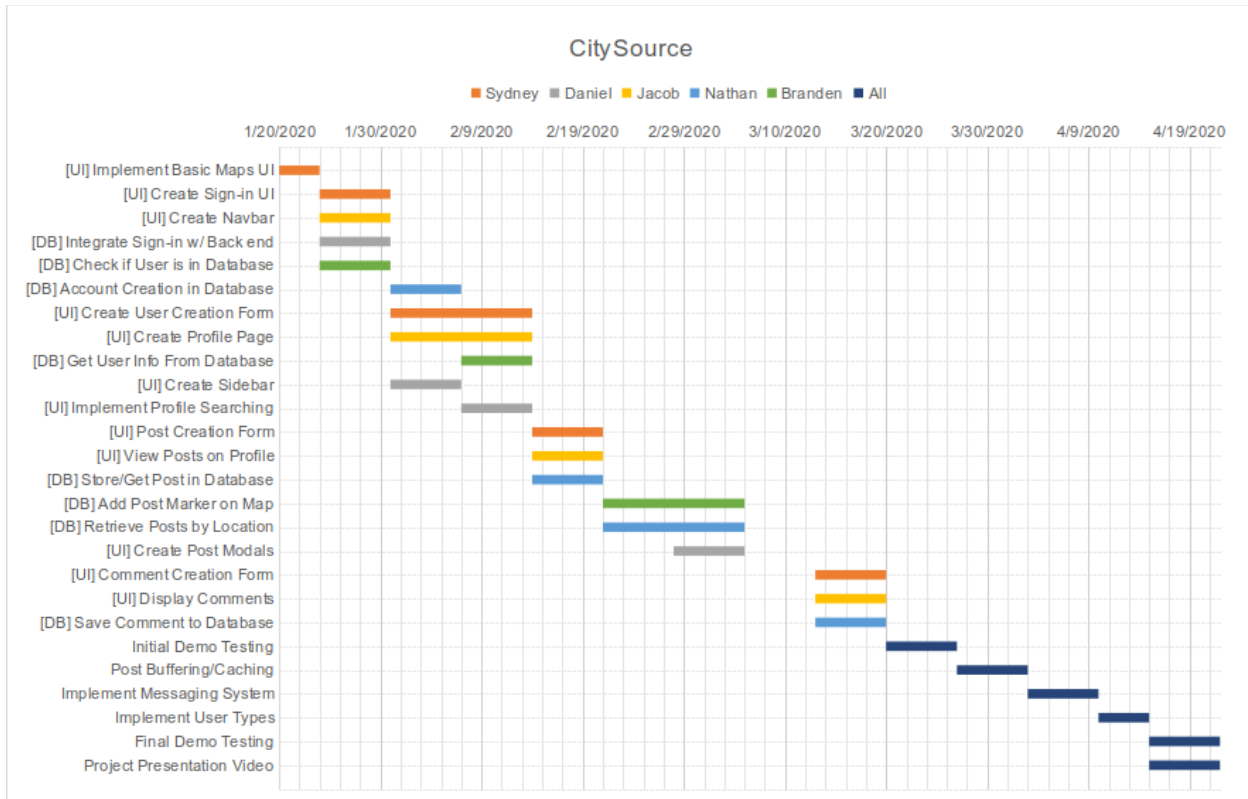
Project Budget

- Domain Name: ~\$12/year.
- Cloud storage/database hosting: \$0-\$25/month. Storage pricing will be largely dependent on expected load. AWS and Google Firebase offer free tiers which will be sufficient in early development.

Work Plan

Daniel, Jacob, and Sydney are responsible for the frontend. Branden and Nathan are responsible for the database and backend connection. All are additional responsible for testing, documentation, and integrating their part of the application from the front end to the back end or vice versa.

Gantt Chart



Preliminary Project Design

CitySource will be developed as a responsive web application, using React. We chose React for ease of use due to its component-based, modular design.

We will have to store information, such as users, emails, posts, comments and messages. Posts will include videos and images, which will encompass the bulk of our data. We plan to use Google Firebase for our database, in order to consolidate the sources we draw from, as we already intend to use the Google Maps API.

As the main visual feature of the CitySource app is its map UI, we will utilize the Google Maps API to implement the map UI. Users must be able to access developer and resident posts from the map, as well as create new posts using the map. The Google Maps API provides functionality which will aid in ease of implementation.

The app will distinguish between three classes of user, developer and resident, and will include a subscription and verification process to register a developer account. These users will be able to create profiles and interact with a map UI which shows developments based on the user's geographic location. They will be able to enter a posting mode where they can drag and drop a marker on the map to add a new post. Users with developer accounts will be able to post proposed building projects. These posts can be commented on by other users. Users with resident accounts will be able to comment on developer posts. They will also be able to post development requests for their area. There will be a visual identifier for a developer account. In the comments of a post, there will be a visual identifier to distinguish local commenters from non-local commenters. There will also be messaging functionality between users. There will be tabs at the bottom of the screen for notifications, messages, trending and recommended posts, user profile, etc. There will be a reporting system for malicious or spam comments. A city planner user will be able to view all comments, messages, and posts.

Below is a high level description of app functionality, accompanied by supplementary images.

User Accounts

When opening the app, users will be prompted to sign in or create an account. They will be able to sign in through Google or Facebook, and choose to create either a resident or developer account.

Developer accounts will have the ability to create and edit a user profile, create and edit posts (development or potential development), comment on posts, and send and receive messages. They will be verified on a case-by-case basis and pay

a subscription fee, which can be managed through their profile. Verified developer accounts will have a visual identifier attached to the account name (for example, a star or other symbol). They may manage comments on their posts through a reporting system.

Resident accounts will have the ability to create and edit a user profile, create and edit posts (desired development), comment on and “like” posts, and send and receive messages. They will not require verification and no subscription fee is required.

Posting

A user enters a **posting mode** by clicking a create a post button (in the below figure, “Post an idea!”). By clicking this button, the user will enter into another mode where the user can tap the map, adding a marker, and drag the marker to its intended location. The user will then be able to edit the auto-filled address fields and the post information, which will include adding pictures, a title and the project description.

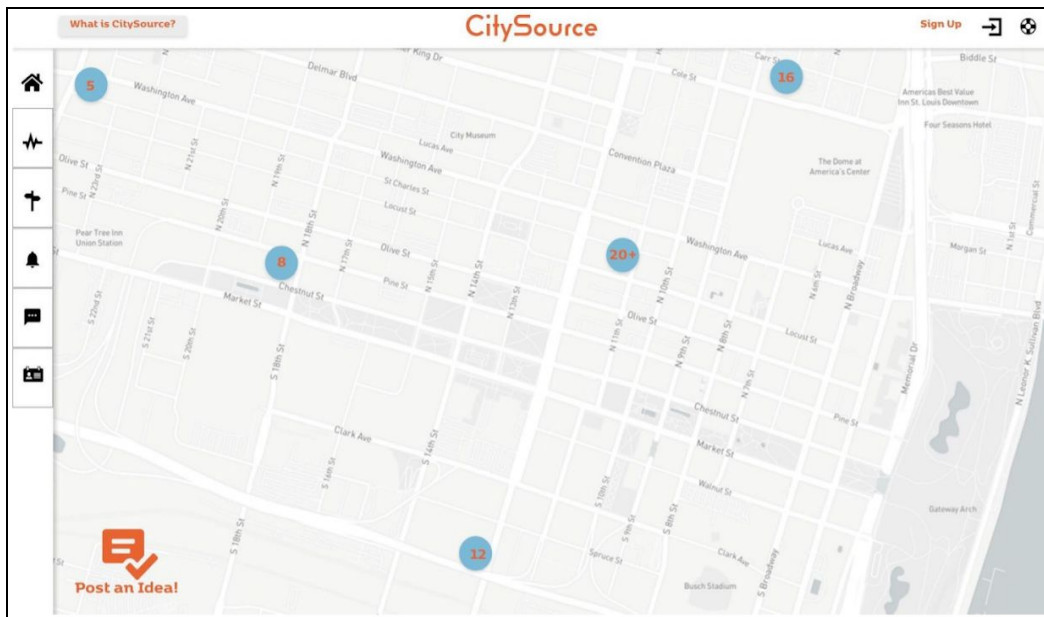
Developer posts will refer to that developer’s intended and potential building projects. The post will include a title, the developer account name, pictures, and a description. It will include a comments section.

Resident posts will refer to desired building projects (for example, to suggest building a coffee shop or restaurant in a particular area). It will include a comments section.

Map UI

The **map UI** encompasses most of the app’s visual space. When a user enters the app, the map UI will take up most of the screen. It shows markers that designate building projects which have been added by users, either developers or residents. When zoomed out, close-together markers are replaced by a single marker showing the number of posts it encompasses (for example, “20+”), and when zoomed in far enough, each post has its own distinct marker. Tapping on a marker will open the post.

The map UI is similarly present in the posting mode, which has been described in the previous section.



The above figure shows a web-based design for the CitySource app. It shows the map UI and examples of zoomed-out post clusters.

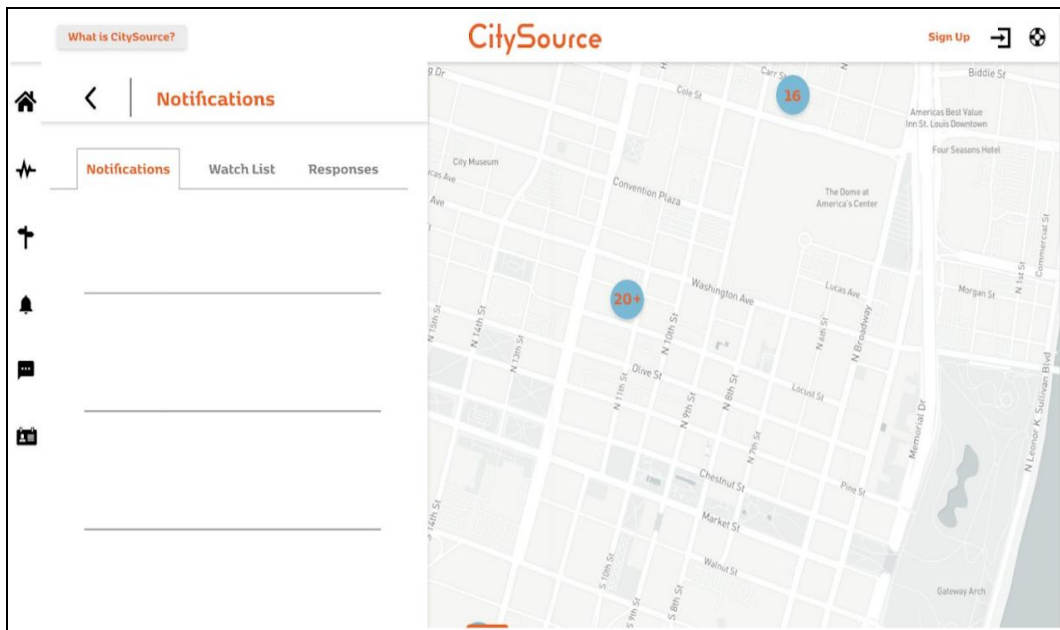
Commenting

Users can **comment** on both developer and resident posts. The comments section of a post (developer or resident) will be divided into comments by local users versus non-local users. Developers can manage unwanted comments on their posts through a reporting system.

Posting and commenting is the most important functionality of CitySource, as these are the avenues by which the dialogue this app intends to foster is created. These functionalities have high priority.

Notifications

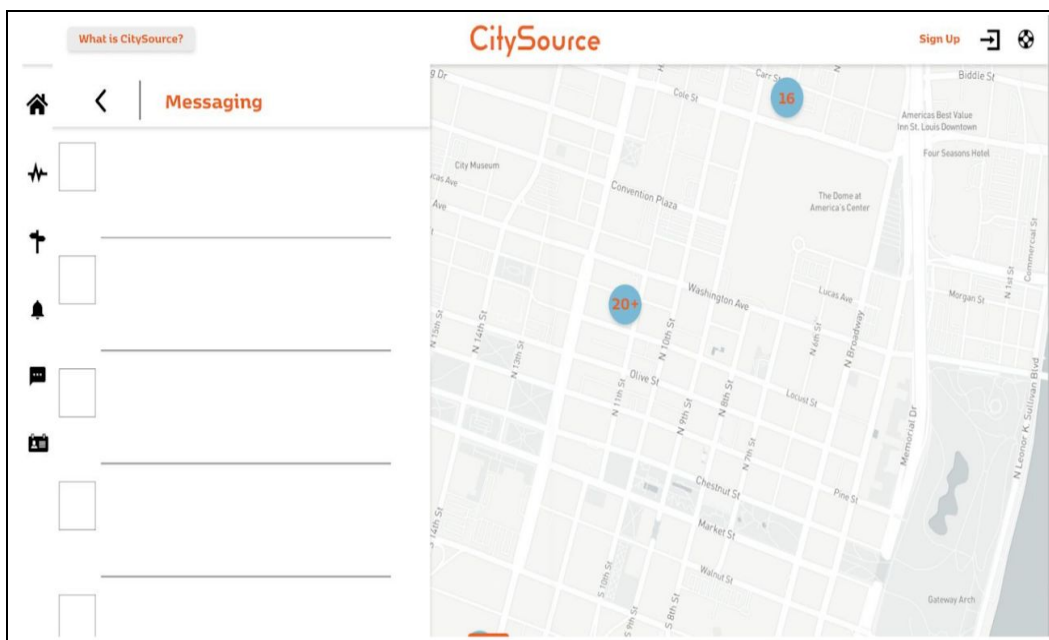
Users will receive **notifications** about comments on their posts. This notification will also be delivered to their email address. Users will also receive notifications about new messages.



The above figure shows a web-based design for the CitySource app with the notifications tab visible.

Messaging System

Users will be able to send direct **messages** to each other. Messages can be sent between resident users and between resident and developer accounts. Messaging between developer accounts may not be supported.



The above figure shows a web-based design for the CitySource app with the messaging tab visible.

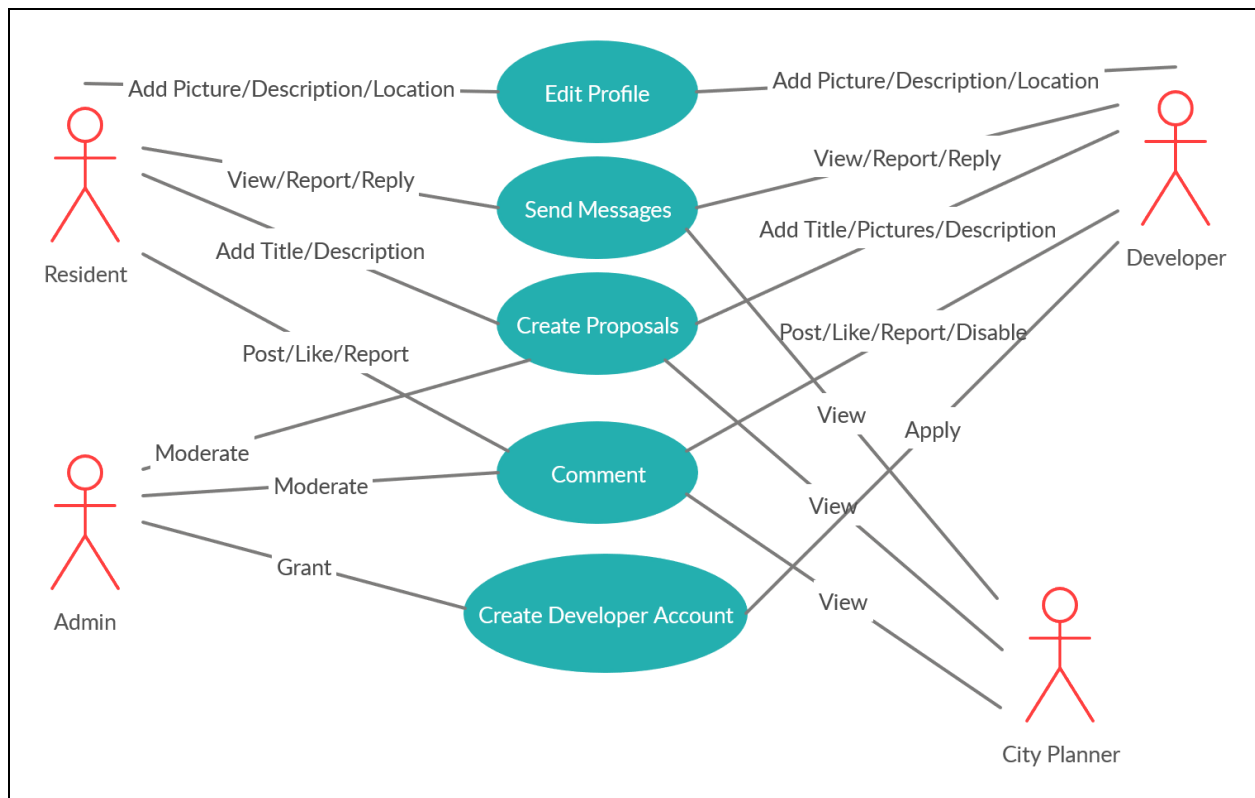
Tabs

Information can be easily accessed via the **tabs** at the bottom of the screen. The tabs include: recommended projects, trending posts, notifications, messages, and user profile. Additional sections may or may not be included.

Design Constraints

The basic constraints of the project are the fact that it needs to be written in React, utilizes the Google Maps API, and also be finished by the end of April.

Use Case Diagram



Ethical Issues

One planned feature in the comment section is the ability for users or developers to report comments for moderation. This feature will primarily be used to report spam comments, comments that don't contribute to the discussion of the development, or comments that are unconstructive. Moderation of these reported comments will be handled by the app developers,

as opposed to allowing the developers to delete comments off of their own posts. If given access to this privilege, some developers might use it to delete public criticism of their posts, even those that do provide constructive feedback. In order to be fair and nondiscriminatory, we (the developers) will be the only ones who will have the ability to delete comments, or the developers will have the option to make all comments private and be able to respond to posts privately.

Intellectual Property Issues

For the app, the intellectual property of the end app would belong to the members of Team 12. As per the agreements with Google Firebase which we will be using for the databases, the rights of any information we store in them would still be retained by the app. The rights for Google maps are retained by Google; we are merely utilizing it for the app. The users of the app would retain all of their rights to any information that they post on the app, be it developments, ideas, or comments that are posted. However, we do reserve the right to monitor and remove any content that is posted.

Change Log

Changed description of the project to reflect a change from mobile app to responsive web app.