

## Contributions

Gary:

GitHub Submissions: 110

- Milestone Formatting
- Front Page
- Hover Implementation of Front Google Map
- Filter
- Mobile on Front

Justin:

GitHub Submissions: 77

- Detailed View
- Form View
- About View
- Stylesheet
- Website appearance/aesthetic
- Testing Responsiveness

Gerren:

GitHub Submissions: 77

- Initial project setup
- Database setup (creating tables, defining schema)
- CRUD functions to modify database tables (issues, search, status categories tables)
- Implemented the search functionality on the back-end using %LIKE
- Back-end functionality for submitting new issues (process the form, store the image onto the file system, store the issue into the database)
- Worked some of the back-end functionality for the user dashboard

Uzair:

GitHub Submissions: 107

- Implemented Authorization and Authentication
- Wrote CRUD functions for issue submission and retrieval
- Maintained Git repository
- Took care of deployment and remote database connections
- Contributed to home page design, and form validation

Eva:

GitHub Submissions: 80

- User Details/City Manager Dash Board (Full Front End)

- Helped with Milestone
- Code Analysis

Lily:

GitHub Submissions: 81

- Added longitude and latitude for map location of each issue
- Displayed issues on Google map in search pages
- End-to-end implementation of create issues:
- Added more inputs for the existing html form
- Wrote javascript to validate input, call Google geoencoder to map address to longitude and latitude, and display errors
- Wrote backend code to process the post request, extract and copy image files to public/images, and insert the new issue into the database
- Showed Google map for the issue on detailed views: javascript to make AJAX query to server to get issue long and lat and display it on the page.
- End-to-end implementation of user comments:
- Created a new table for comments in db migrations
- Wrote front-end code to load all comments for each issue (using AJAX), make sure only logged-in user can comment, validate comment content, send post request to server, and append the newly posted comment to the comment list
- Wrote back-end code to query comments for an issue from the database and insert a new comment into the database