

Final Status Report For Listy



Instructor:
Team:
Submitted:

Matthew Burlick
Sita Robinson, Miguel Pereyra, Shane Roland
December 10, 2017

Project Idea

Listy is a web application that allows people to make their own checklists and share it with any other member (Listlet) of Listy. The target audience for Listy are families, students, and coworkers. Although, on a broader scale, it could apply to any group of people who find use in it. Listy could be used in a variety of situations, such as when making grocery lists, homework lists, or dorm lists. If a Listlet made a grocery list, they would be able share it with other registered family members. The family members would be able to add items to the list and check items off once at the grocery store. Similarly, if the Listlet was taking a class with other registered friends, they could share homework that needed to be completed so all of them could keep track. If they had a group project they could share items that needed to be completed before the deadline. Lastly, if the Listlet and their friends were having trouble making a dorm list one year because there were a lot of items required, they could use Listy to make the process easier. In essence, giving multiple Listlets working on a list the ability to share them would streamline the process of checking off its items.

Functional Requirements

- A secure registration and login page.
- The database for the website, which contains a datatable for users as well as list datatables for each user.
- User should be able to create and save a list.
- User should be able to edit a list add entries.
- User should be able to retrieve stored lists.
- User should be able to delete a list
- It should be possible for users to share lists with other users.
- It should be possible for users to retrieve lists shared with them.

Scope of Development Effort

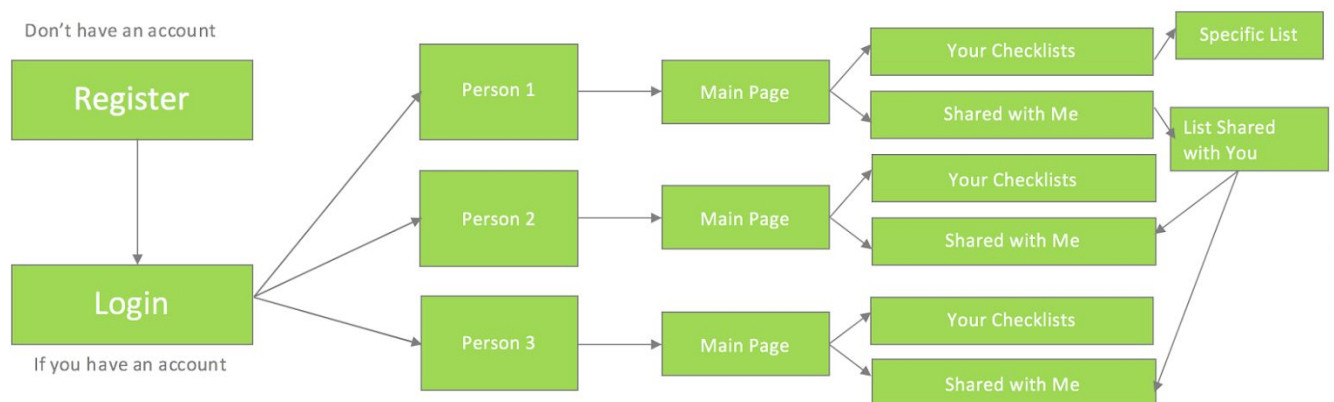
As a minimum viable web app people need to be able to register and sign in. Once they are signed in they need to be able to make lists and share them with other registered users. All the elements of the minimum scope was completed. In our future scope, we will add a feature so users can put in dates when specific items are due. We may also add a countdown on how much time is left to complete that specific item. Another idea we may implement would be the ability for users to categorize their lists. We will also add the ability of an item to grey out or move to the bottom in a section of finished tasks in the event a user checks it off. Another future goal might be creating a view checklist screen so when a Listlet shares their checklist with other Listlets, they have the option to either give them edit mode or view mode. This is because in some cases, you might not want to give someone edit permissions on your list.

Architectural Design

Technologies to be utilized

Listy has both a client and server side. The client side (frontend) technologies our group used is HTML, CSS, and Javascript. The server side (backend) technologies used was Node.js and MySQL.

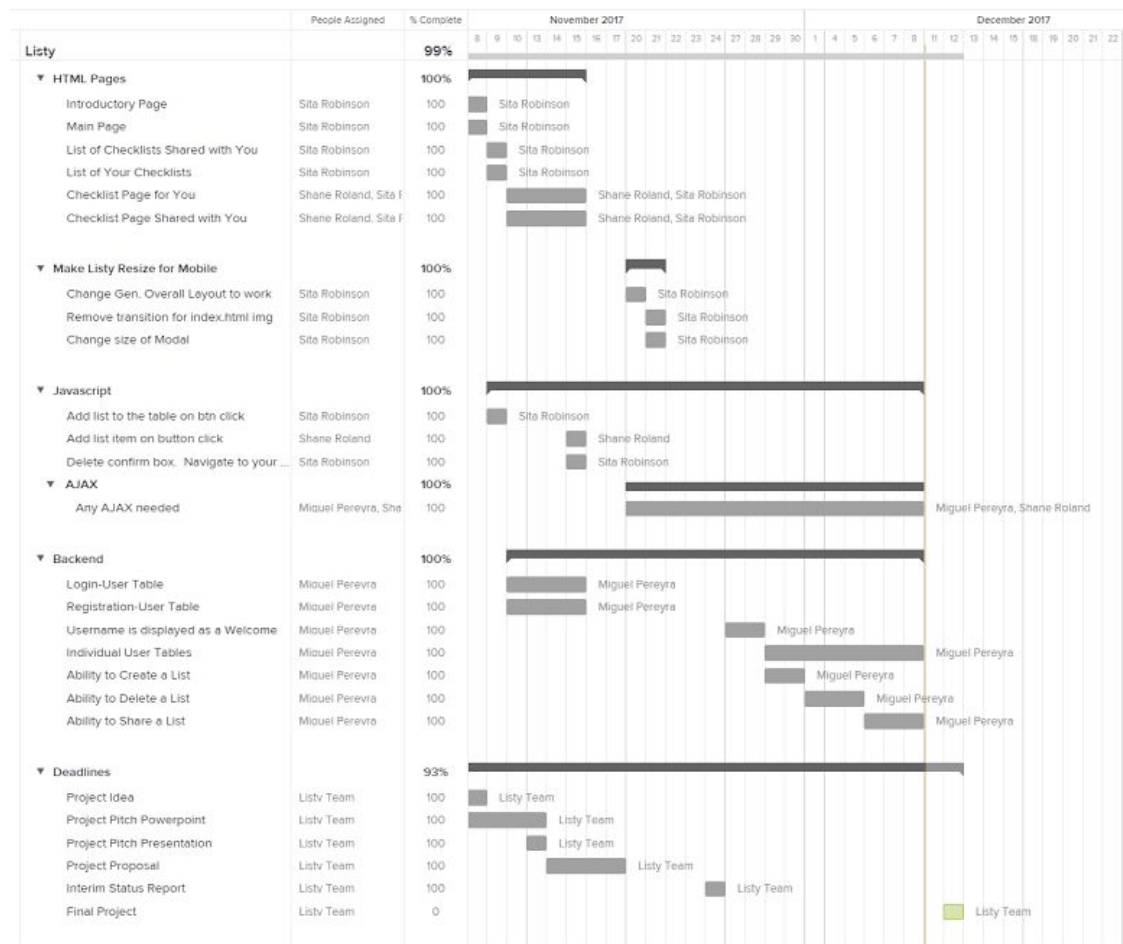
Information Flow Diagram



Gantt Chart

Key details of the plan / schedule

Finished items are in grey and tasks still to do are in green. The HTML of the web application is completed. Listy now resizes correctly for mobile devices with the help of JQuery Mobile. The backend including the Node.js, MySQL, and also the AJAX was completed as well. The backend database with the required data tables is up and running. The login and registration modules are now finished. They correctly handle errors such as empty fields and incorrect login information. A session variable containing the user's username has also been implemented, so that the site will recognize for which user to handle list information. We finished everything that was in the minimum scope.



Roles and Responsibilities

Sita Robinson

- Make sure group is on track to finish by week 11.
- Frontend and Listy Design
- Made sure Listy resizes correctly for mobile devices using JQuery Mobile

Miguel Pereyra

- Backend - specifically Node.js and MySQL.
- Created the database with the tables needed by the website.
- Created login, registration, list creating, and list sharing capabilities.

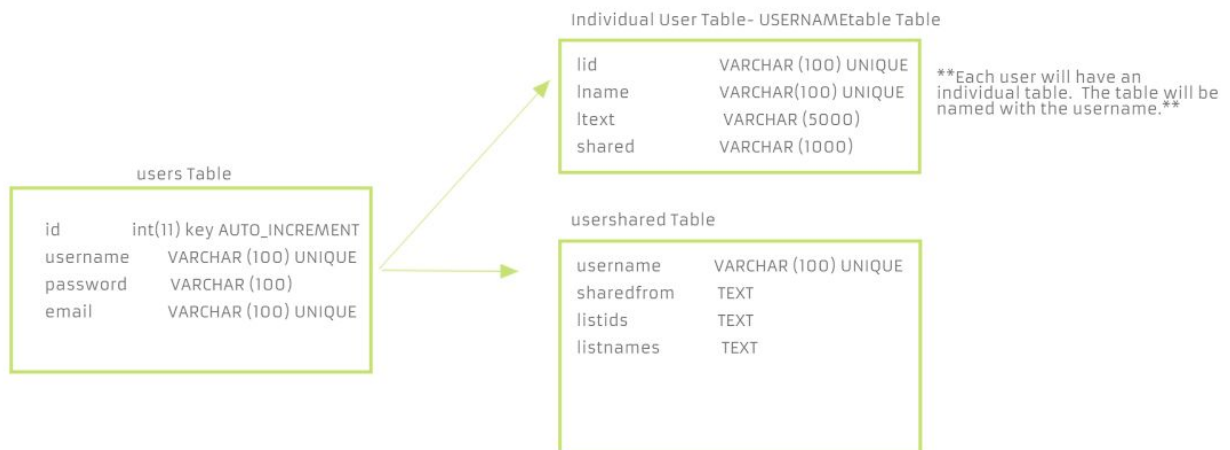
Shane Roland

- Frontend - created some of the HTML and Javascript needed.
- Wrote the Javascript and the AJAX necessary for communicating with the backend.

All group members did testing to ensure that the errors were minimal.

SQL Database Schema

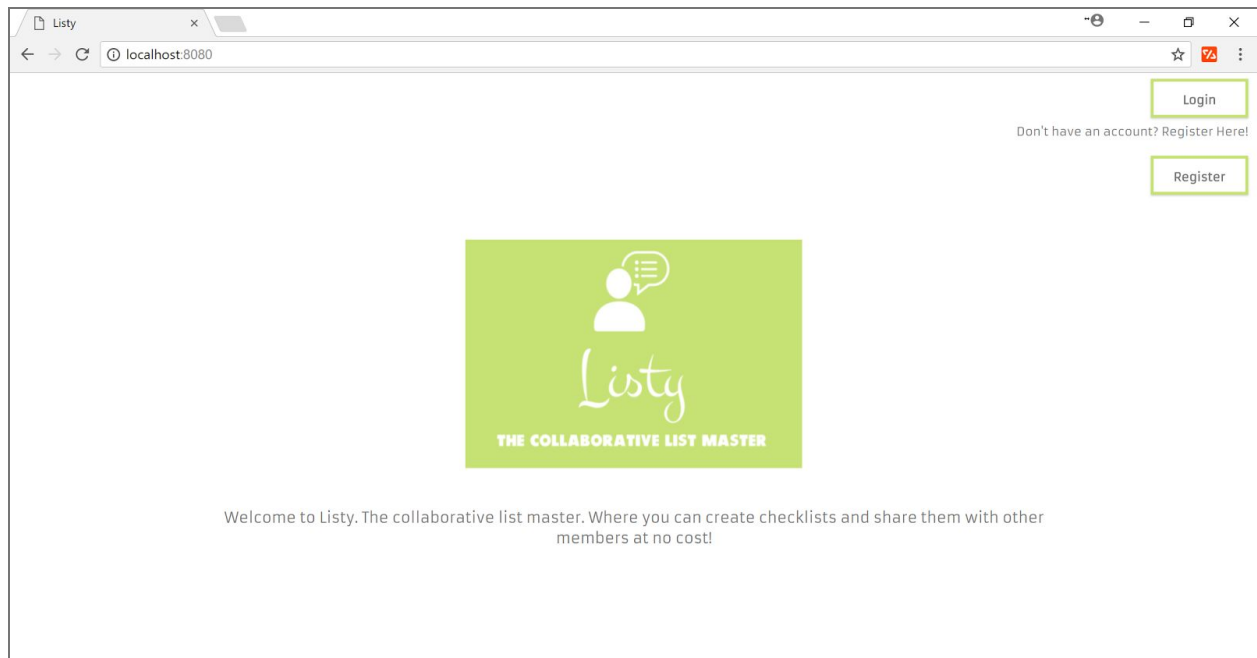
The database has the user Table for registered users. At registration, a list data table is created for each individual user (with the name being <username>table Table). This is where the user's lists are stored. One column contains the name of the list, the next would contain the text of the list, and then the last column contains the users it has been shared with. The users shared column contains all the users the list will be shared with, but have the names separated by a special character so that the server will send back the multiple users. Multiple list information is put into the respective individual user table. Finally, to allow for sharing of lists, another table, 'usershared' will hold a row for each user. The row will hold all the lists shared with them. There is a column for list names, list ids, and from whom that list was shared. This information collectively provides a connection between a user and the lists with them on other users table.



Screenshots

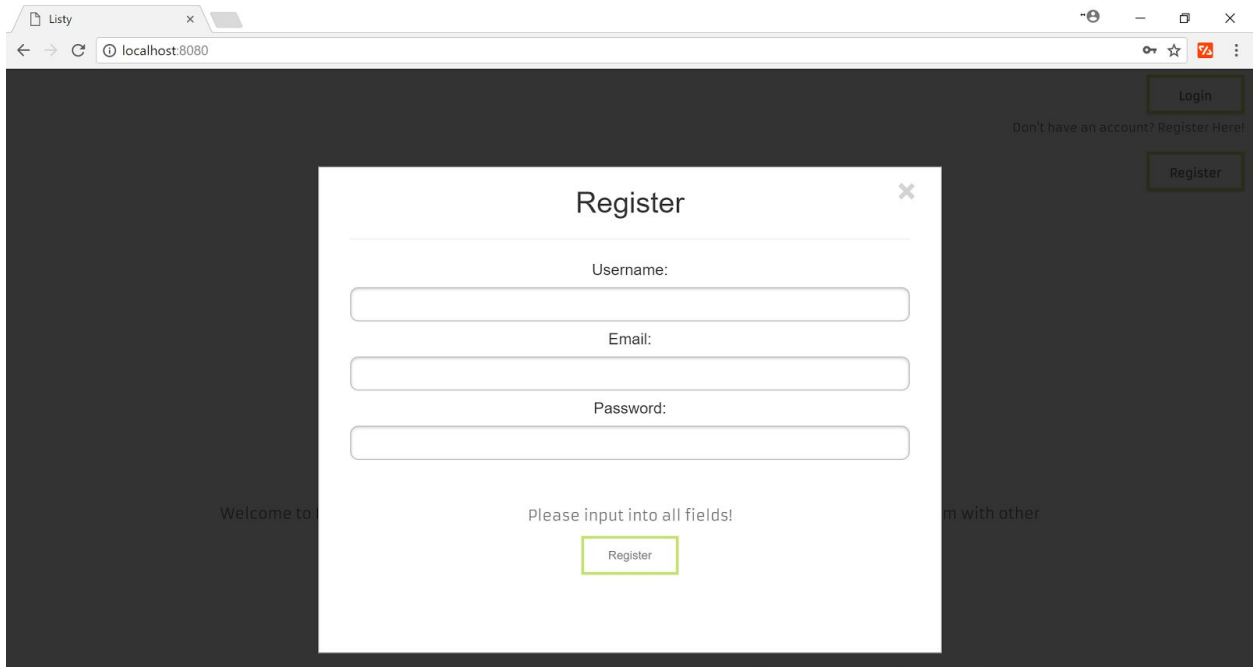
Introductory Page

First page someone sees if they go to localhost:8080 when the server is running. Has a welcome message to the user along with an animated Listy logo for four seconds. If the screen is too small for example for mobile devices, the animation will not occur. This is also the page that a user can login and register. A modal box will popup for login or registration respectively when the buttons are clicked.

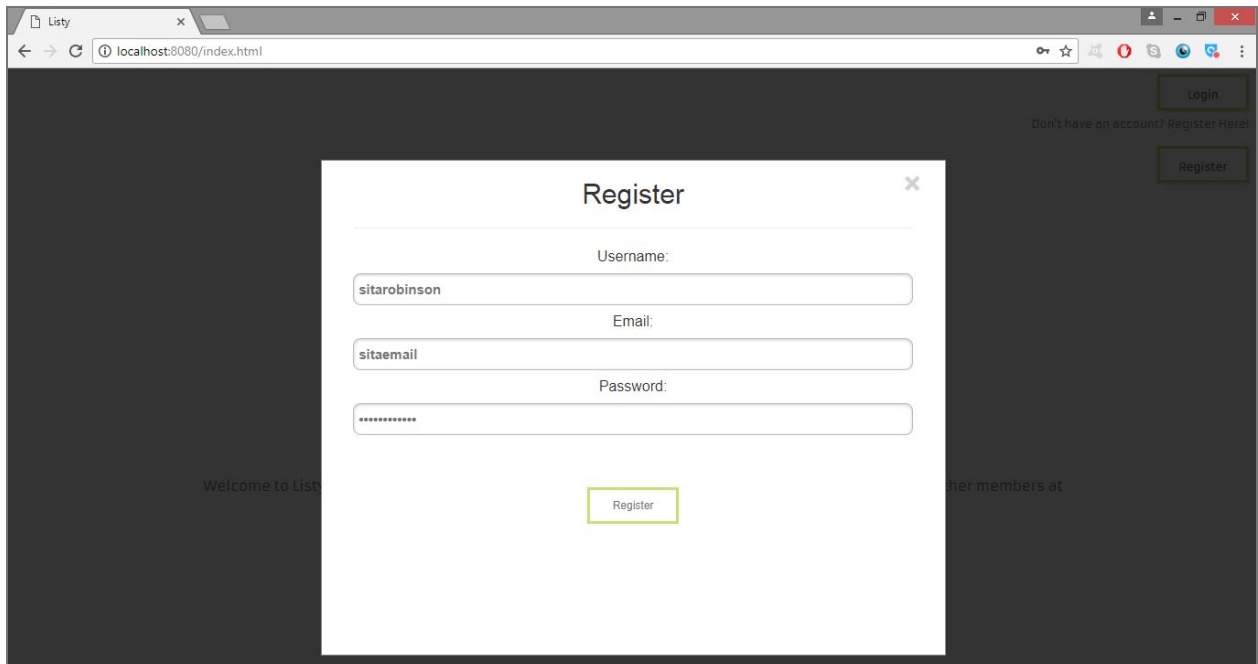


Introductory Page-Registration Modal

When the login button is pushed this modal pops up. It prompts for a user to enter a username, email, and password. When a user incorrectly enters one of the fields an error message will pop up.



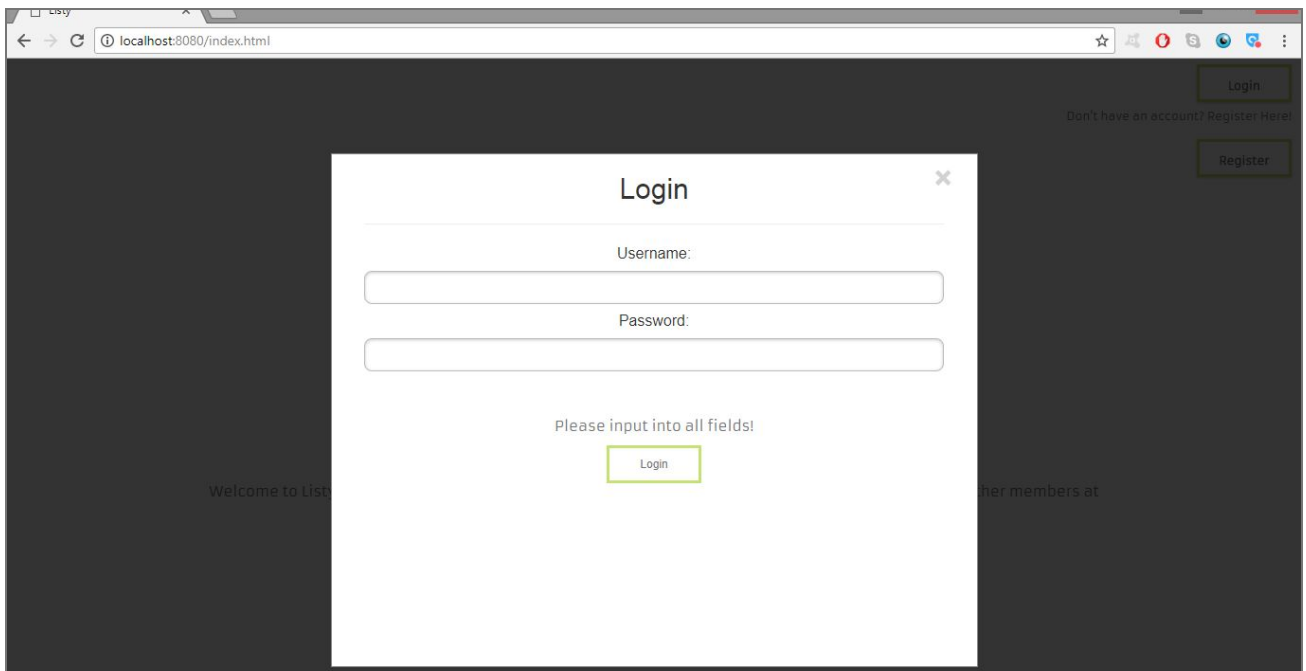
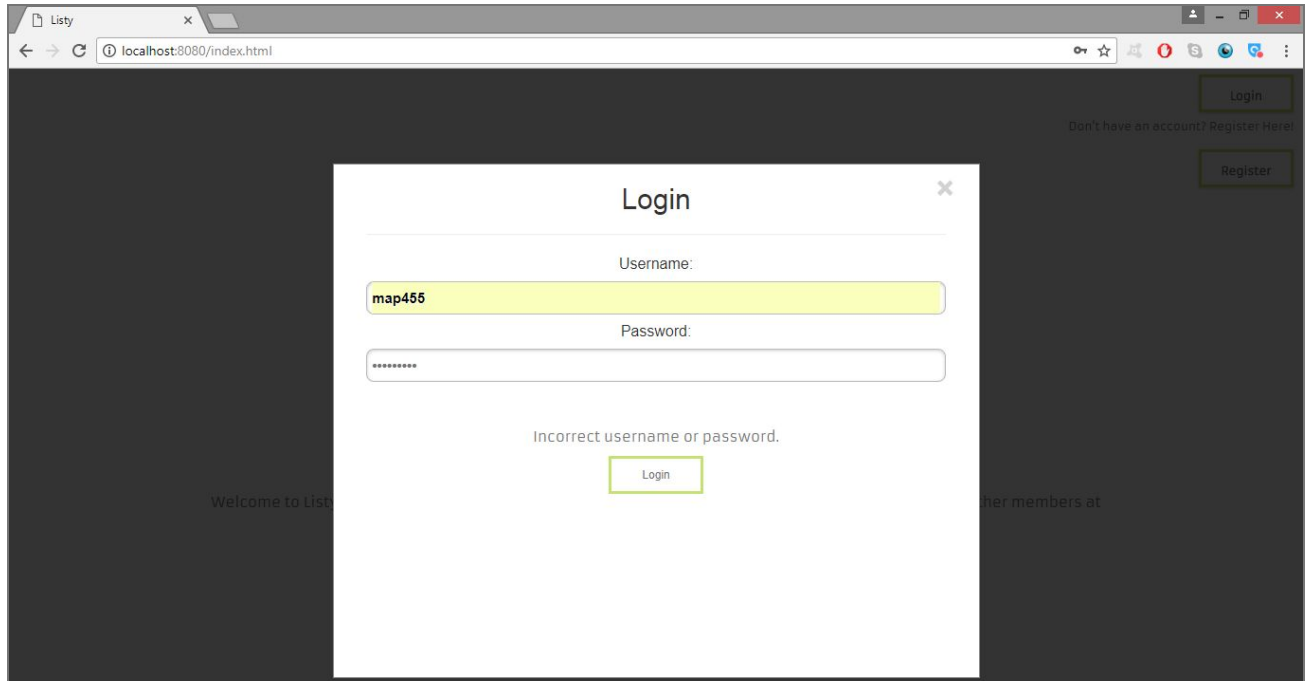
A screenshot of a web browser window showing a registration modal. The browser's address bar displays "localhost:8080". The modal, titled "Register" with a close button (X), contains three input fields: "Username:", "Email:", and "Password:". Below these fields is a message "Please input into all fields!" and a "Register" button. In the background, a dark grey page is visible with a "Login" button and a link "Don't have an account? Register Here!".



A screenshot of the same web browser window, but the registration modal now contains sample data. The "Username:" field is filled with "sitarobinson", the "Email:" field with "sitaemail", and the "Password:" field with "*****". The "Register" button remains highlighted. The background page is the same as in the previous screenshot.

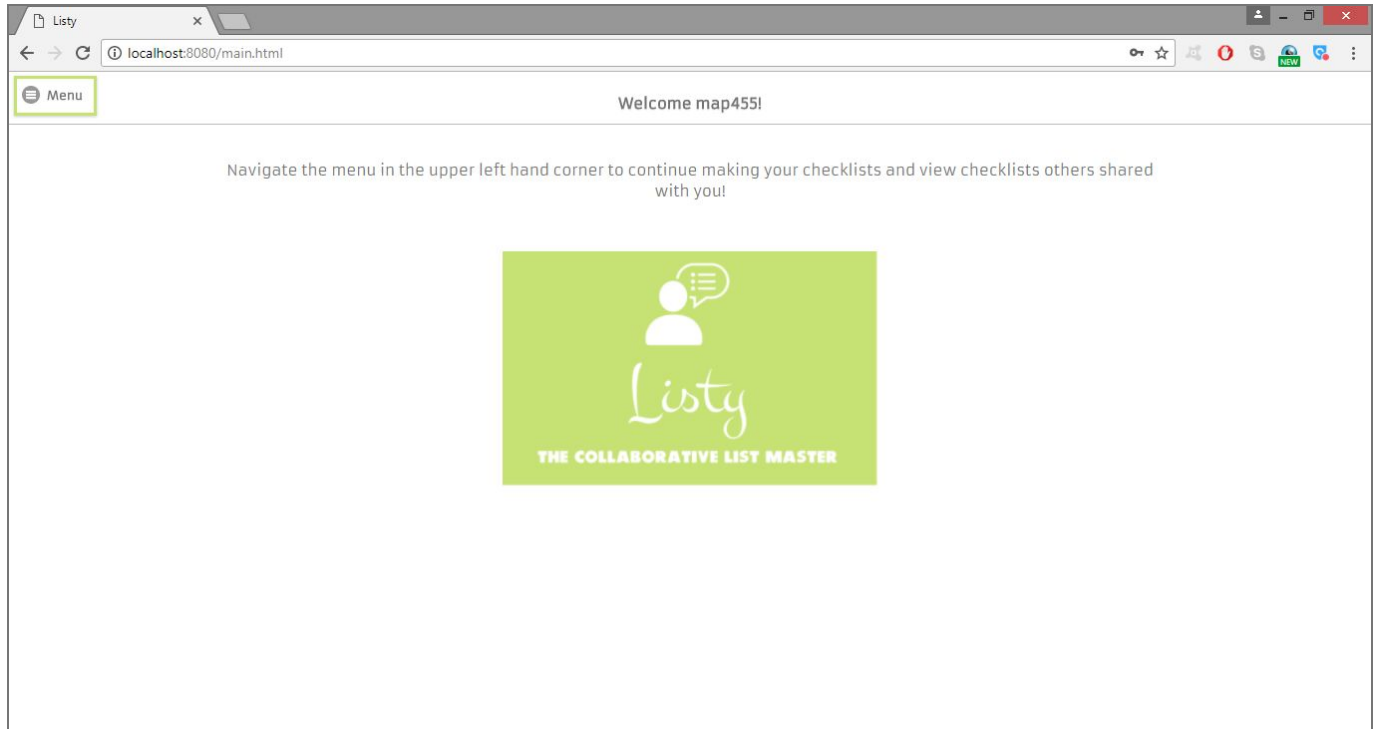
Introductory Page-Login Modal

If a user has already registered, they can try to login. In order to login they need to enter a correct username and password. When successful they will be moved to the main page.



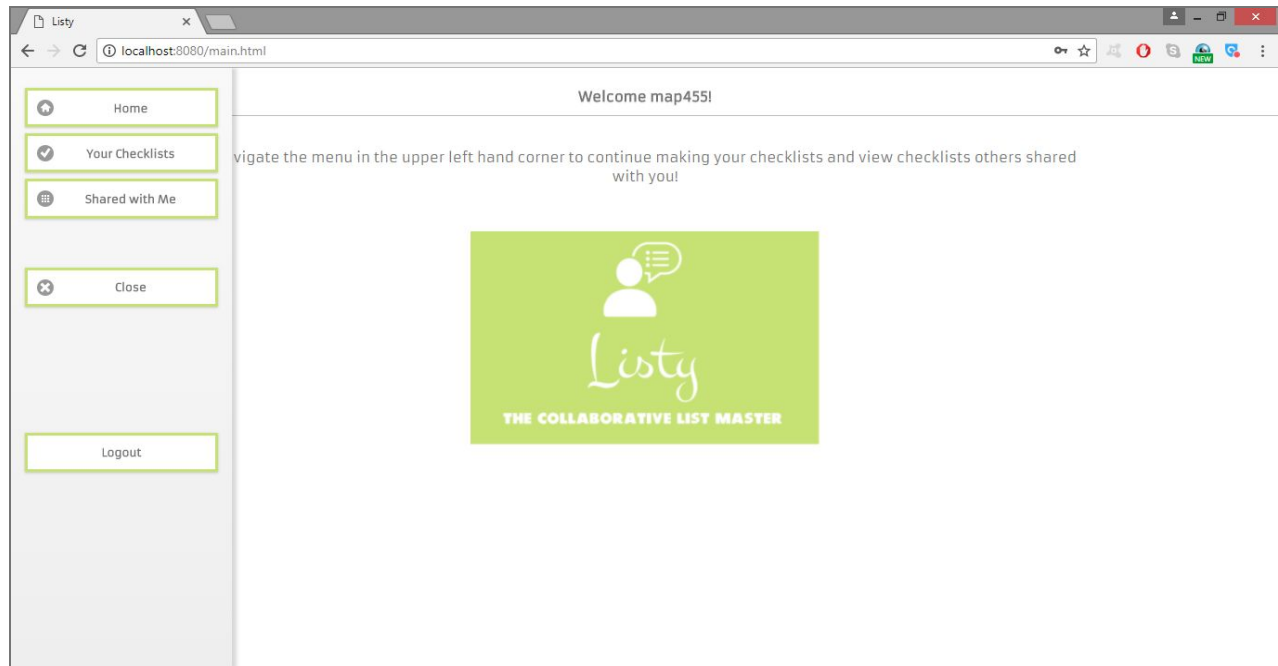
Main Page

Once a user logs in they will be moved to the main page. It says Welcome along with the username of the user that just logged in. If you hit the button that says menu, a panel will be overlayed to the left.



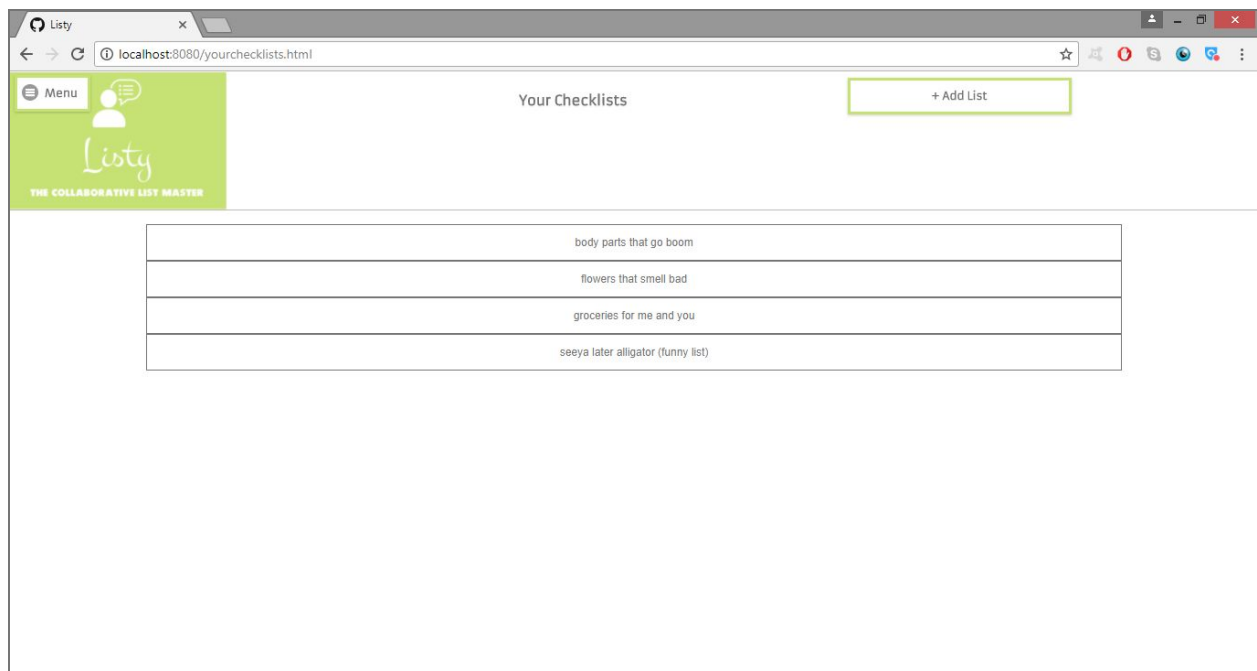
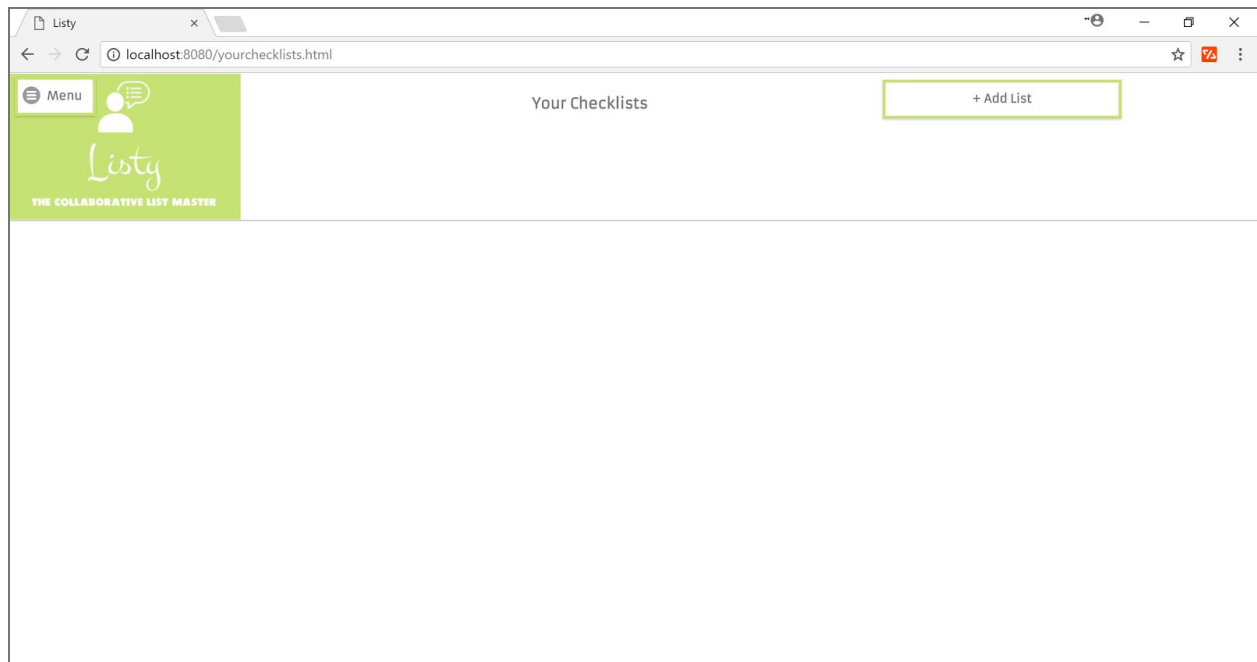
Panel

On every page except login(introductory page). The panel has three main options along with a logout button. The three options are to view your checklists which will show the list of your checklists, checklists shared with me to show the list of checklists other users shared with you, and home which is just the main page (the current page in the image).



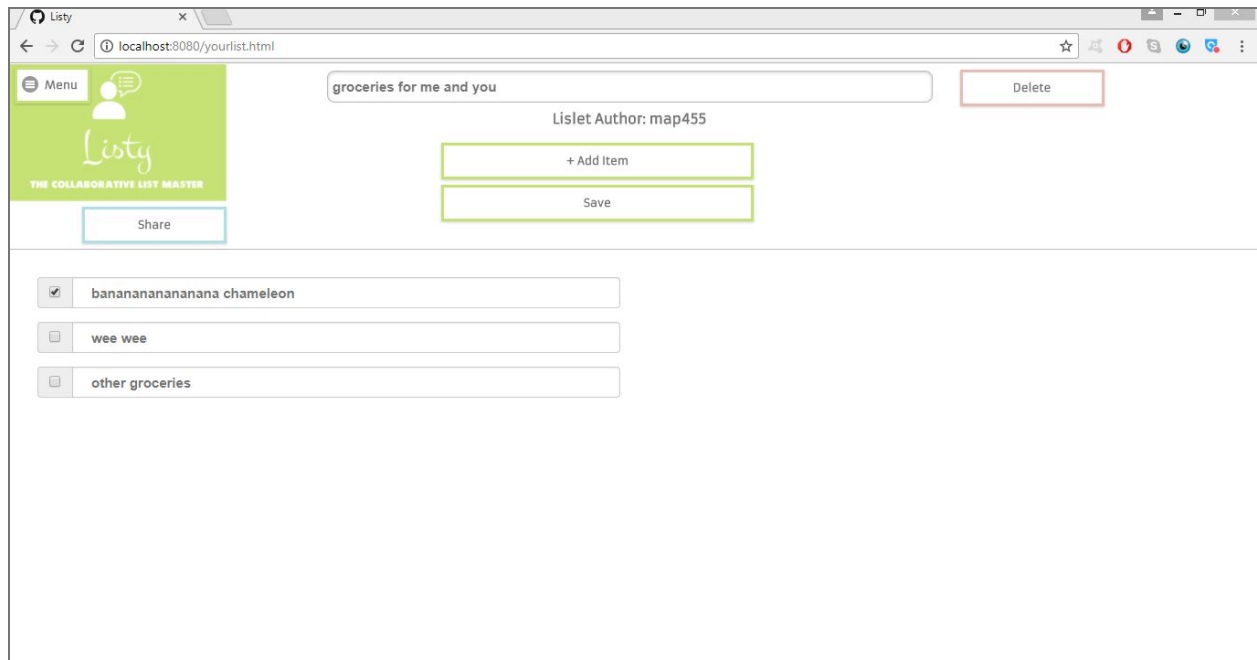
Your Checklists Screen

Show the current user's checklist. If they click the button Add List, it will move the user to a list page where they can add the title and add items. Once they hit save, it should save the list information in the database and be shown as a new row of the table will be added with the list name that the user specified. To edit the already created list they can click on the row which will take them back to the your list screen for that list.



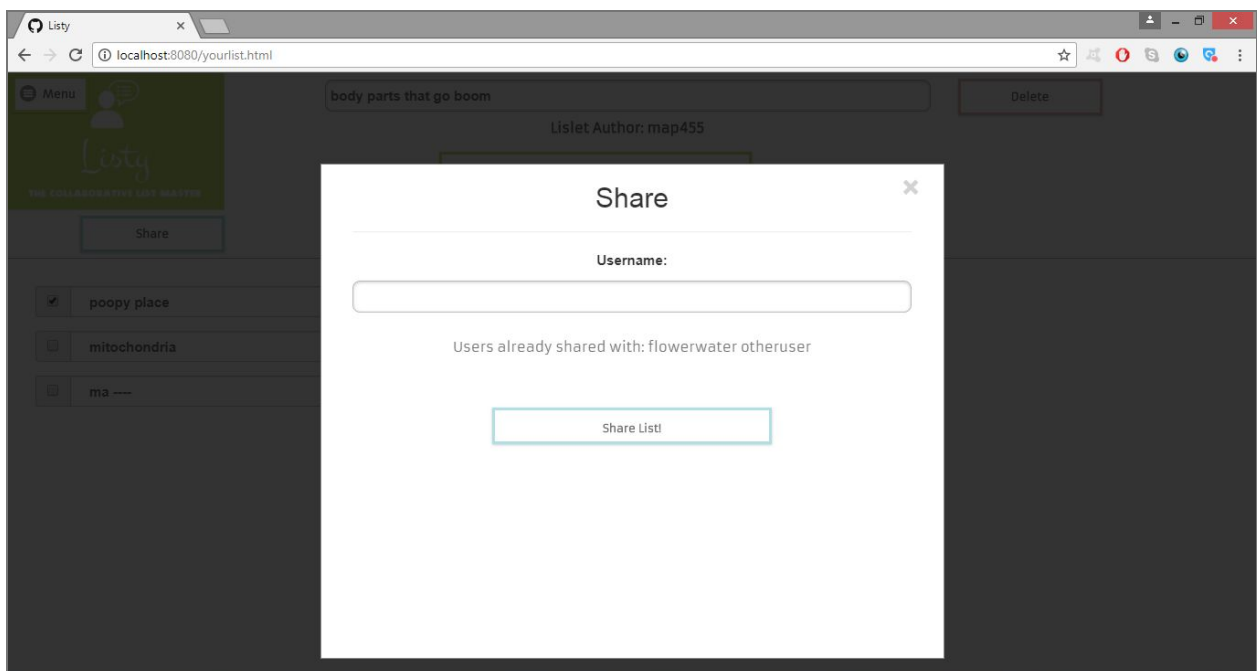
Your List

This is the list page. A user is directed to this once they either click on an existing row on the your checklists page or click the add list button. A user should be able to share, delete, and add an item to a list since they are the author.



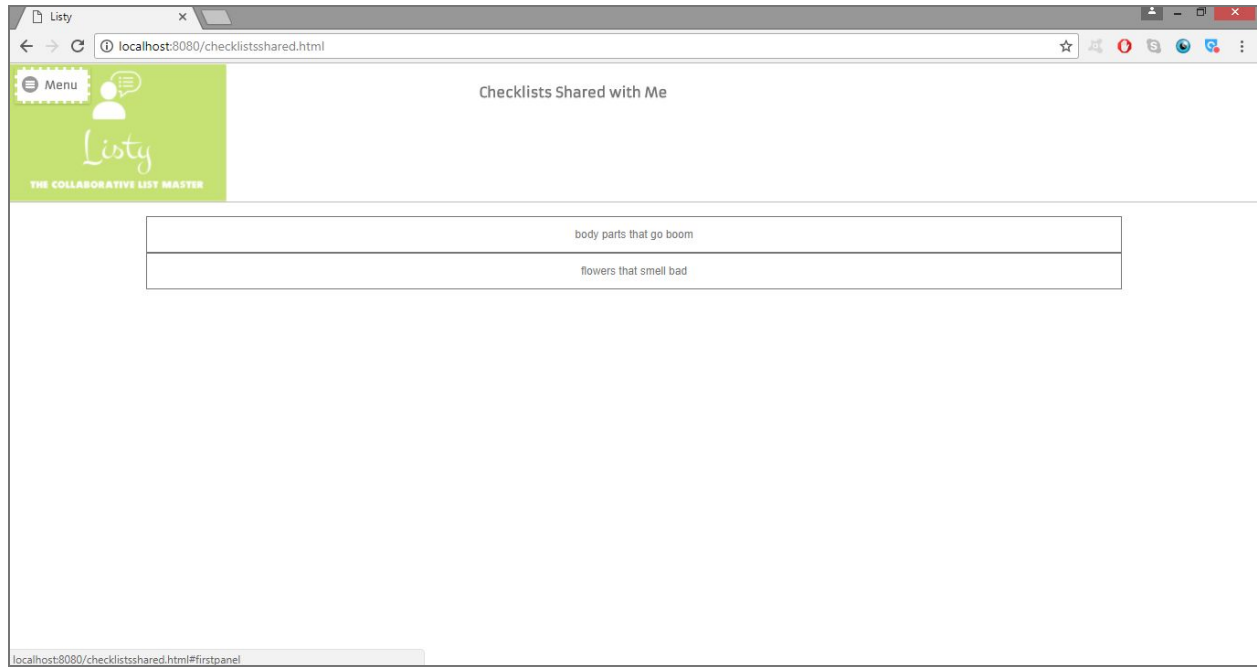
Your List-Share Modal

Will popup if you click the share button. A user is able to enter a username of another user to share a list with that user. If it is already shared with a user, it will show the usernames that the list was shared with.



Checklists Shared with Me

Shows the checklists shared with the current user. To edit the name or the list they can click on the row which will take them to the list shared screen for that list.



List Shared with You

This is the list shared page. A user is directed to this once they click a row on the checklists shared with me page. A user is able to add an item to a list but since they are not the author is not be able to share a list, delete a list, or edit the title of the list.

The screenshot shows a web browser window with the address bar displaying "localhost:8080/listshared.html". The page has a green header with a "Menu" button and the "Listy" logo, which includes the tagline "THE COLLABORATIVE LIST MASTER". The main content area features a title field containing "flowers that smell bad" and an author field showing "Lislet Author: map455". Below these fields are two buttons: "+ Add Item" and "Save". At the bottom, there is a list of items, each with a checkbox and a text input field:

- ☒ all of them
- ☒ (flowers == hate)
- ☒ totally emo boi