

Florida International University
Knight Foundation School of Computing and Information Sciences

Software Engineering Focus

Final Deliverable

Community Connect
Fall 2022

Team Members: Hunter Badillo, Zackary Santana, Natasha Rodriguez, Eduardo Lopez, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Castor Velasquez, Kenny Afonin, Daniela Agueros, Samuel Perez, Felix Rodriguez, Michael Banegas, Jasen Cadet, Miguel Sablan

Product Owner(s): Shahin Vassigh

Mentor(s): Biayna Bogosian, Giancarlo Perez

Instructor: Masoud Sadjadi

The MIT License (MIT)
Copyright (c) 2016 *Florida International University*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Abstract

This document presents the information necessary to gain a good understanding of the current iteration of the Community Connect website.

The previous iteration of this project focused on completing the redesign of the user-interface. It also focused on implementing several new features. Despite the efforts made prior, some features ended up not fully implemented or not working properly.

For this iteration, the goal was to add more functionalities to the redesigns made in the last iteration. Our focus was to create a working news page where users could see articles relevant to their community, and to create a working forum page where users could create posts and comment under other's posts.

Table of Contents

Introduction	6
Current System	6
Purpose of New System	9
User Stories	10
Implemented User Stories	10
Pending User Stories	10
Project Plan	11
Hardware and Software Resources	11
Sprints Plan	12
Sprint 1	12
Sprint 2	14
Sprint 3	16
Sprint 4	17
Sprint 5	18
System Design	20
Architectural Patterns	21
System and Subsystem Decomposition	21
<i>Front-end architecture tools</i>	21
<i>Back-end architecture tools</i>	22
<i>Database</i>	22
<i>Deployment Diagram</i>	23
Design Patterns	23
System Validation	23
Glossary	25
Appendix	26
Appendix A - UML Diagrams	26
Class Diagram	26
Appendix B - User Interface Design	29

Appendix C - Sprint Review Reports	36
Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents	41
Community Connect Prototype (skip all docker commands for now)	42
Required Services	42
Structure of the project	42
Running Project Locally	42
1. Clone the repository	43
2. Navigate to the project's root directory (i.e. where package.json, docker-compose.yml, and .dockerignore files are located)	43
3. Install node dependencies for the Frontend	43
Windows Backend Installation Instructions	43
4. Create a virtual environment for the backend.	43
5. Activate the virtual environment.	43
6. Install the dependencies in the environment.	43
7. Migrate the database models.	43
Mac/Linux Backend Installation Instructions	44
4. Create a virtual environment for the backend.	44
5. Activate the virtual environment.	44
6. Install the dependencies in the environment.	44
7. Migrate the database models.	44
Running the project locally (from the root directory, no environment)	44
Running the frontend	44
Running the backend	44
Windows	44
Mac	44
Deployment through Ocelot using docker	45
4. Build the docker image (skip for now)	45
5. Make and migrate database models	45

6. Create an admin user for the Backend (skip for now)	45
7. Change your Windows settings/MacOs settings in your host file (skip for now)	45
8. Run the project (skip docker commands)	46
9. Shutting down the project	47
Setting up the Backend Development Environment (Outdated)	47
1. Install pipenv	47
2. Enter the shell environment	47
3. Installing all required packages	47
References	48

INTRODUCTION

The Purpose of this capstone project is to continue developing the Community Connect website. Primarily, the website's goal is to be an interactive platform to allow users, ranging from medical professionals to ordinary individuals, to gain more information on guides, trends, and news related to any urgent issues in their communities.

Current System

First iteration

The first iteration of the Community Connect Website was completed by the Capstone I team of Gabriel Perez, Samara Ruiz-Sandoval, Mark Guadalupe, Nathanael Garcia, Eugene Theodorovych. The focus of the work completed prior was the custom user interface specifically with the addition of front-end components to the base forum, user experience, as well as deployment of the system to both development and production servers. This involved detailed research of the project packages that would fit the desired stack. Research on Docker, Django,

and Node.js was done during the development of the previous system as well as research of the complex Misago package structure, and proper troubleshooting techniques. The search bar was implemented with elastic search to yield results for basic profile information of users that are registered in the forum, keywords mentioned in threads, and a link to provide a full page of results found across both threads and users. A theme was previously developed using Adobe XD by Sophia Cabral to assist in the specification of colors, fonts, and styles for the pages within the discussion forum. Graphics were also implemented from the mock-up designs to enforce the purpose of the website that incorporates news relevant to the user, trending content from their location, and connecting to their community. Finally, custom card components were added for future development of the new features.

Second iteration

The second iteration of the Community Connect Website was completed by the Capstone II team of Gabriel Perez, Samara Ruiz-Sandoval, Mark Guadalupe, Nathanael Garcia, Eugene Theodorovych. The purpose of the new system was focused on designing new features that would make the system more attractive and intuitive for the user experience. The new system would have to serve as the platform for discussions on health-related issues and possible solutions for those, as well as the point of connection between different vendors and the customers. As the second iteration of the existing project, the development involved the creation of the new features that included cards in the dashboard that redirect to news pages for Guides, Trends, and News, a colored theme for the main and splash pages and improved December 2, 2021 Page 7 of 82 Final Deliverable Mitigating the impacts of COVID19 searching across the forum. Research and connection of different APIs that tightened all processes in the back end resulted in proper visualization, which allowed the content to be rendered in the needed way, eventually simplifying the user experience and fulfilling the needs of the project. Bing News and Google Trends API call, after certain data cleaning, allowed data population to correspond to pages. Cron jobs were scheduled on the servers to update the JSON files needed for the Trends page. IPAPI call was connected to get the user location and display relevant data on both News and Trends pages. An Embedded Google map was rendered to the Trends page to allow users to see the nearest vaccination sites.

Third iteration

The third iteration of the Community Connect Website was completed by the Capstone II and Senior Project team composed of Gretel Gomez Rodriguez, Juan Grajales, Dayana Garcia, Carolina Rodriguez, and Norlan A. Castillo. The team began a new system which recreated the previous semester's community connect project without the use of Misago. The purpose of the new system was to redesign the web application's architecture in order to decouple the frontend and backend to ensure a better and more organized structure for the project files. The new system

has the same intended purpose to serve as the platform for discussions on health-related issues and possible solutions for those, as well as the point of connection between different vendors and the customers. The team was able to recreate the frontend using React and created the foundation for the backend using Django and Docker. A basic news API was added towards the end of the last sprint.

Fourth iteration

The fourth iteration of the Community Connect Website was completed by the Capstone II and Senior Project team composed of Adrian Gonzalez, Niurbelis Gonzalez, Sebastian Gomez, Shaoting Wu, Chábeli Castano Arango, Narmeen Kibria, Manuel Kamboykos, and David Garcia. The purpose of this iteration is to continue the work of the team that completed the previous iteration. The team began with a complete overhaul of the design of the site, redesigning every page to the new UI/UX spec. The entire project was also restructured for a more streamlined developer experience when it comes to project installation, building, and running. Docker images were also refactored for better system efficiency and a reduction of the image size, improving build times. New systems and technologies were installed, such as Next.js, to utilize the built-in features that come with the new framework, such as server-side rendering, dynamic routing, and static site generation. The backend was also redone, removing unnecessary and unused endpoints. Towards the end of the last sprint, the foundation for a Carbon Footprint Calculator was implemented, but still needs to be connected to an API, such as Carbon Interface, and requires interactive graphs to be implemented with Chart.js.

Fifth iteration

The fifth iteration of the Community Connect Website was completed by the Capstone II and Senior Project team composed of Francisco Mazzei, Leinier Perez, Yelise Sanchez, Brock Williams and Kevin Escoto. The original purpose for the website was to be a platform for discussions on health-related issues and possible solutions for those, as well as the point of connection between different vendors and the customers. While this is still an intended purpose of the website, the latest iteration has expanded its purpose to provide an additional focus on the environment and sustainability. With this renewed purpose in mind, the goal of this iteration was to continue the work of the past team in building on the website design and functionality. This work consisted of adding new pages to the site, completing several design modifications from the original UX/UI specifications given to the team at the start, as well as completing the carbon

calculator and news feed components. The carbon calculator on the Carbon Footprint page was successfully connected to the Carbon Interface API, effectively using real world data to calculate an estimate of a user's carbon footprint based on their inputs in various fields. The news feed was also successfully connected to the Google Scholar API as well as the News API on the News page, so users may filter between reliable sources of scholarly articles or news articles about their selected topics of interest. While the News page is functional in development it is not functional after deployment because the News API key is only free for development purposes. Using the News API key in deployment requires a subscription payment which the Product Owner has decided to postpone. The Forums page is not yet functional as more work is needed for posting discussions and replying to ongoing threads created by other community members. No changes have been made to the technologies utilized in the prior iteration of the project, and the updated website has been deployed for public use through FIU's server.

Purpose of New System

Sixth iteration

The sixth iteration of the Community Connect Website was completed by the Capstone II and Senior Project team composed of Hunter Badillo, Zackary Santana, Natasha Rodriguez, Eduardo Lopez, Michael Olivero, Miguel Ferreiro, and Raul Gutierrez Niubo. The purpose of this platform is to connect people in a particular community who may be facing some natural or man-made disaster of any kind. While the previous rendition focused on allowing users to gauge their environmental impact, this iteration focuses on refining the places where most users would be interacting with each other; the News and Forum page. The News page is a place for users to find out what is happening in their community. This page is supposed to display articles based on the user's chosen topics of interest. These are predetermined topics that have to do with environmental / man-made events or disasters. The forum page is supposed to show posts from users who are in the same general area as the user who is viewing them. For example, a user who is in a place prone to hurricanes should be able to see other users posting content about their hurricane tips and experiences. This gives users who are in the same community the opportunity to collaborate and possibly receive help from one another.

USER STORIES

The following section provides the detailed user stories that were implemented in this iteration of the project. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories

SP2 - 8 pts - As a developer, I would like the News page to pull articles from the internet

SP2 - 2 pts - As a developer, I would like the 'Information Source' header changed to a News API Citation

SP2 - 5 pts - As a developer, I would like forum posts to include a username

SP3 - 4 pts - As a developer, I would like a secondary forum page for reading the content of the forum posts

SP3 - 2pts - As a developer, I would like the 'Log In' button to say 'Log Out' after a user has logged in

SP4 - 6pts - As a user I should be able to see the content of a forum post with a comment box to write comments

SP5 - 8pts - As a backend dev, I should have access to consume the database via the preplanned schema to allow for endpoints

SP5 - 8pts - As an end user, I should be able to use a fully functional news page

Pending User Stories

- User story #12 - As a developer, I would like to fix the scholarly articles api to search for articles. (8pts)
- User Story #2 - As a developer, I would like to fix the news api to search for news. (8pts)

PROJECT PLAN

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

- **Docker:** A free and open-source platform of OS-level virtualization to host the needed images(parts) of the project.
- **Docker desktop application** - optional visualization tool for Docker images
- **GitHub:** A web-based version-control and collaboration platform for software developers.
- **GitHub desktop:** Optional support application that enables Git collaboration.
- **Visual Studio Code:** a source-code editor developed by Microsoft for Windows
- **Bootstrap:** An open-source CSS framework that is also used for quickly and efficiently creating the designs and layouts for web pages.
- **Django:** A Python free-open source framework that manages the back-end of the entire application and custom apps added to Misago core.
- **React:** An open-source, front-end, JavaScript library for building user interfaces or UI components that serve open-source as the engine for the front-end of the application.
- **Next.js:** An open-source development framework that is built on top of Node.js a React that provides additional developer functionality and support for features such as server-side rendering, static site generation, built-in routing and more.
- **PostgreSQL:** A free, open-source database management system that has a separate Docker container and servers as the database for the application.

Sprints Plan

Sprint 1

Sprint Planning Meeting Minutes:

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zachary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadet, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez

Start time: 12:00 PM

End time: 12:25 PM

After discussion, the velocity of the team was estimated to be 10 hours

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story 1 - As a developer I would like to install the project on my local machine. (1 hour)
- All team members must become a contributor to the site's GitHub repository. (1 pt)
- User Story 2 - As a developer I would like to familiarize myself with React. (2 hours)
- User Story 3 - As a developer I would like to familiarize myself with github. (1 hour)
- User Story 4 - As a developer I would like to familiarize myself with git. (1 hour)
- User Story 5 - As a developer I would like to familiarize myself with Nextjs. (2 hours)
- User Story 6 - As a developer I would like to familiarize myself with Python. (2 hour)
- User Story 7 - As a developer I would like to familiarize myself with Django. (2 hour)
- User Story 8 - As a developer I would like to familiarize myself with Docker. (1 hour)

The team members indicated their willingness to work on the following user stories.

Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zachary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadet, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez

- User Story 1 - As a developer I would like to install the project on my local machine. (1 hour)
 - User Story 2 - As a developer I would like to familiarize myself with React. (2 hours)

- User Story 3 - As a developer I would like to familiarize myself with github. (1 hour)
- User Story 4 - As a developer I would like to familiarize myself with git. (1 hour)
- User Story 5 - As a developer I would like to familiarize myself with Nextjs. (2 hours)
- User Story 6 - As a developer I would like to familiarize myself with Python. (2 hour)
- User Story 7 - As a developer I would like to familiarize myself with Django. (2 hour)
- User Story 8 - As a developer I would like to familiarize myself with Docker. (2 hour)

Sprint 2

Sprint Planning Meeting Minutes:

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zackary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadet, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez
Start time: 12:00 PM
End time: 12:25 PM

After discussion, the velocity of the team was estimated to be 40 points

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

News page

<News1> [2 pt]: Remove 'Information Source' header and reformat side tab to account for additional space

<News2> [2 pt]: Display news articles from google

Forum page

<Forum1> [1 pt]: Add items to the composition of a discussion post (topic tags, username)

<Forum2> [1 pt]: Prototype a secondary forum page for reading conversations and posting replies

Login

<Login1> [3 pt]: Remove the 'Log In' button after a user has logged in

<Login2> [2 pt]: In the place of the 'Log In' button, place a profile button where users can go to edit their profile information

The team members indicated their willingness to work on the following user stories.

1. **Hunter Badillo <News1>**
2. **Michael Olivero <Login 1>**
3. **Miguel Ferreiro <Forum1>**
4. **Raul Gutierrez Niubo <Login>**
5. **Eduardo Lopez <News1>**
6. **Natasha Rodriguez <news>**
7. **Zackary Santana: <Login2>**

- 8. Kenny Afonin <News1>**
- 9. Daniela Agueros <Login2>**
- 10. Michael Banegas <Forum1>**
- 11. Jasen Cadet <News 1>**
- 12. Samuel Perez <Login1>**
- 13. Felix Rodriguez <Forum>**
- 14. Miguel Sablan <News2>**
- 15. Castor Velasquez <Forum>**

Sprint 3

Sprint Planning Meeting Minutes:

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zackary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadet, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez

Start time: 12:00 PM

End time: 12:25 PM

After discussion, the velocity of the team was estimated to be 32.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

<Forum1> [5 pt]: Add items to the composition of a discussion post (topic tags, username)

<Forum2> [5 pt]: Prototype a secondary forum page for reading conversations and posting replies

The team members indicated their willingness to work on the following user stories.

1. **Hunter Badillo <Forum1&2>**
2. **Michael Olivero <Forum1&2>**
3. **Miguel Ferreiro <Forum1&2>**
4. **Raul Gutierrez Niubo <Forum1&2>**
5. **Eduardo Lopez <Forum1&2>**
6. **Natasha Rodriguez <Forum1&2>**
7. **Zackary Santana: <Forum1&2>**
8. **Kenny Afonin <Forum1&2>**
9. **Daniela Agueros <Forum1&2>**
10. **Michael Banegas <Forum1&2>**
11. **Jasen Cadet <Forum1&2>**
12. **Samuel Perez <Forum1&2>**
13. **Felix Rodriguez <Forum1&2>**
14. **Miguel Sablan <Forum1&2>**
15. **Castor Velasquez <Forum1&2>**

Sprint 4**Sprint Planning Meeting Minutes:**

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zackary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadet, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez

Start time: 12:00 PM

End time: 12:25 PM

After discussion, the velocity of the team was estimated to be 48.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

<Forum1> [5 pt]: Add items to the composition of a discussion post (topic tags, username)

<Forum2> [5 pt]: Prototype a secondary forum page for reading conversations and posting replies

The team members indicated their willingness to work on the following user stories.

1. Hunter Badillo <Forum1&2>
2. Michael Olivero <Forum1&2>
3. Miguel Ferreiro <Forum1&2>
4. Raul Gutierrez Niubo <Forum1&2>
5. Eduardo Lopez <Forum1&2>
6. Natasha Rodriguez <Forum1&2>
7. Zackary Santana: <Forum1&2>
8. Kenny Afonin <Forum1&2>
9. Daniela Agueros <Forum1&2>
10. Michael Banegas <Forum1&2>
11. Jasen Cadet <Forum1&2>
12. Samuel Perez <Forum1&2>
13. Felix Rodriguez <Forum1&2>
14. Miguel Sablan <Forum1&2>
15. Castor Velasquez <Forum1&2>

Sprint 5

Sprint Planning Meeting Minutes:

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zackary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadet, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez
Start time: 6:00 PM

End time: 6:30 PM

After discussion, the velocity of the team was estimated to be 48

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

<Frontend Forum Implementation> [6 pt]: As a user, I should be able to see the discussion/comments/posts of a specific post by selecting the title and going to the post page. I should also be showed the comment box to make my own comment.

<Frontend Forum Design [4 pt]>: As a user, the post page should be presentable and fit the branding of the rest of the website.

<Backend Forum Database Schema> [6 pt]: As a backend dev, I should have access to consume the database via the preplanned schema to allow for endpoints.

<Backend Forum Endpoints [6 pt]>: As a frontend dev, I should have access to endpoints that allow me to accomplish tasks like viewing a post, viewing many posts, viewing comments on a post (maybe the same endpoint as viewing a post).

<Frontend/Backend connecting [8 pt]>: As an end user, I should be able to use a fully functional post page that can do everything a forum/post page should do (see below).

We have separated the capstone team to two teams, one that focuses on frontend changes and one that focuses on backend changes. The user stories assigned to each person is broken up as such between the teams:

Frontend: 1, 2, 5

Backend: 3, 4, 5

Here is the separation between the teams:

Frontend Team:

Zackary Santana

Natasha Rodriguez

Eduardo Lopez

Samuel Perez

Felix Rodriguez

Raul Gutierrez Niubo

Jasen Cadet

Backend Team:

Hunter Badillo

Castor Velasquez

Miguel Ferreiro

Miguel Sablan

Michael Banegas

Michael Olivero

Kenny Afonin

Daniela Agueros

Raul Gutierrez Niubo

Jasen Cadet

SYSTEM DESIGN

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. Finally, the design patterns used in the project are discussed.

Architectural Patterns

The current architecture for Community Connect utilizes React for the client and Django for the server while still using Docker. This is a full-stack application that utilizes React Javascript framework as the main frontend component, whereas Django Python framework manages backend processes. Additionally, the main database is managed by PostgreSQL in order to manage the cache of the application. The project has 2 versions for development and deployment accordingly. When the local-development version consists of parts described above, the deployed version additionally consists of an Nginx proxy and SSL certificate handler as well utilizing Docker. Each part of the application has a separate Docker image and the application as the whole is composed in one big Docker container before being deployed to the server. The backend runs completely separate from the frontend, so if for some reason the backend is broken, it does not affect the way the user sees the application and its components.

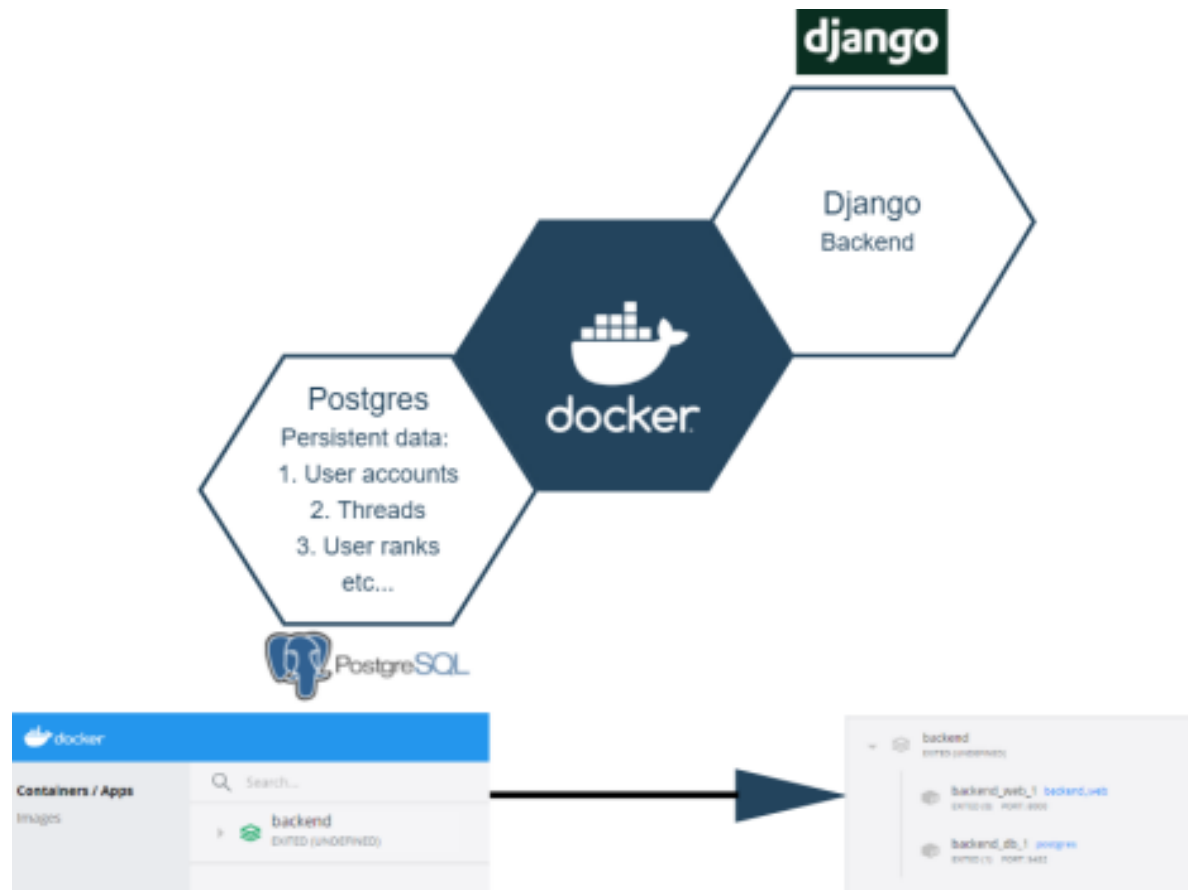
System and Subsystem Decomposition

The current architecture was decided by the previous team, and was created with the following benefits in mind: decouple frontend, backend, and services, facilitate new developer onboarding, and scope CSS to each component. The frontend components are handled by React using hooks. We are using Sass for styling and most components have their own Sass file for styling. There are also several global style files included to facilitate changes in the future such as color theme, button, and fonts. The backend uses Django and Postgres.

Front-end architecture tools

This project is built off of the model, view, controller architecture to separate each of the project's components into easily understandable parts. The model is our database and data structure, the view is the user interface, and the controller accepts and converts the input.

Back-end architecture tools

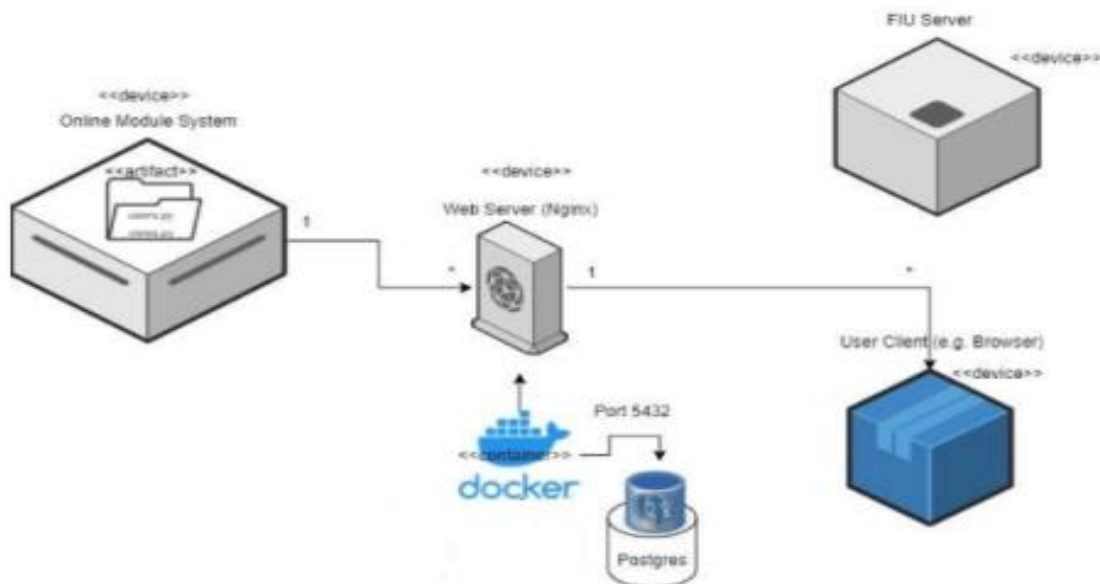


Database

The current database structure was decided upon by the third iteration team, but will have to be redesigned without the use of Misago as the architecture of the project no longer uses Misago. Since our priorities were focused on improving the website design as well as implementing the carbon calculator and news page functionalities, the database will be redesigned in upcoming semesters.

Deployment Diagram

Different developers commit changes, and fetch updates to/from the shared GitHub repository. All component changes are merged and then pulled to the deployment server hosted on an Ubuntu OS where the Docker container resides and is restarted after updates. The project is then served to the Community Connect domain using a Nginx web server where it can be seen by other users.



Design Patterns

Our prototype uses the Model View Controller (MVC) architecture to separate each of the project's components into easily understandable parts. The model is our database and data structure, the view is the user interface, and the controller accepts and converts the input.

SYSTEM VALIDATION

The basic testing our team has done in this iteration concerns the UI/UX of the website and ensuring its appearance and functionality is consistent across different browsers,

OS's, and screen sizes.

The third iteration of Community Connect handled more thorough testing and validation through the use of a python script located in the main project directory. By using the command `./dev test`, the project will run the pytest suite which will provide expressive and readable unit tests for each of the site's applications. These unit tests have not yet been implemented, however, this will be worked on as the project nears completion in future iterations.

GLOSSARY

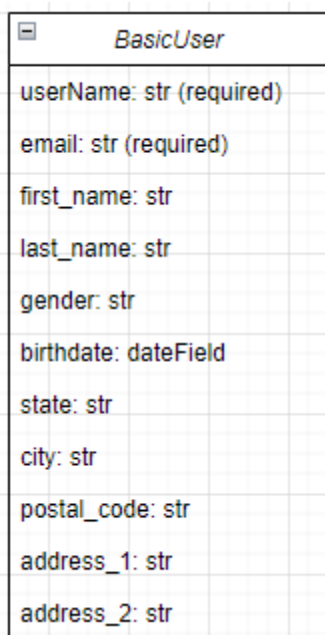
- **Framework:** A free and open-source platform of OS-level virtualization to host the needed images(parts) of the project.
- **API:** Application programming interface allowing two pieces of software to communicate with each other.
- **News API:** REST API that returns search results for current and historic news articles published by over 80,000 worldwide sources.
- **Carbon Interface:** API that generates accurate emissions estimates from flights, vehicles, shipping, electricity consumption and fuel combustion based on the latest data and methodologies.
- **Open-Source:** optional visualization
- **Docker:** A free and open-source platform of OS-level virtualization to host the needed images(parts) of the project.
- **Docker desktop application** - optional visualization tool for Docker images.
- **GitHub:** A web-based version-control and collaboration platform for software developers.
- **GitHub desktop:** Optional support application that enables Git collaboration.
- **FIU Server:** Linux server on FIU premises that hosts actual, production versions of the application.
- **Visual Studio Code:** a source-code editor developed by Microsoft for Windows.
- **Bootstrap:** An open-source CSS framework that is also used for quickly and efficiently creating the designs and layouts for web pages.
- **Django:** A Python free-open source framework that manages the back-end of the entire application and custom apps added to Misago core.
- **React:** An open-source, front-end, JavaScript library for building user interfaces or UI components that serve open-source as the engine for the front-end of the application.
- **Misago:** A forum package that is taken as the base for the application and has a separate Docker container.
- **Next.js:** An open-source development framework that is built on top of Node.js a React that provides additional developer functionality and support for features such as server-side rendering, static site generation, built-in routing and more.
- **PostgreSQL:** A free, open-source database management system that has a separate Docker container and serves as the database for the application.
- **Celery:** An open-source asynchronous task queue or job queue which is based on distributed message passing and that has a separate Docker container.
- **Redis:** An open-source (BSD licensed), in-memory data structure store, used as a database, cache, and message broker; which manages application cache and has a separate Docker container.
- **VM:** Virtual machine is the virtualization/emulation of a computer system.

- **OS:** An operating system is system software that manages computer hardware, software resources, and provides common services for computer programs.

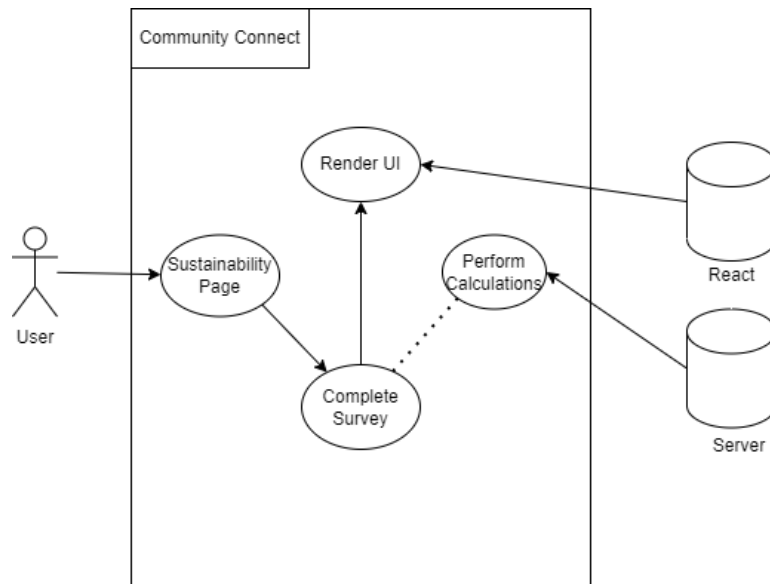
APPENDIX

Appendix A - UML Diagrams

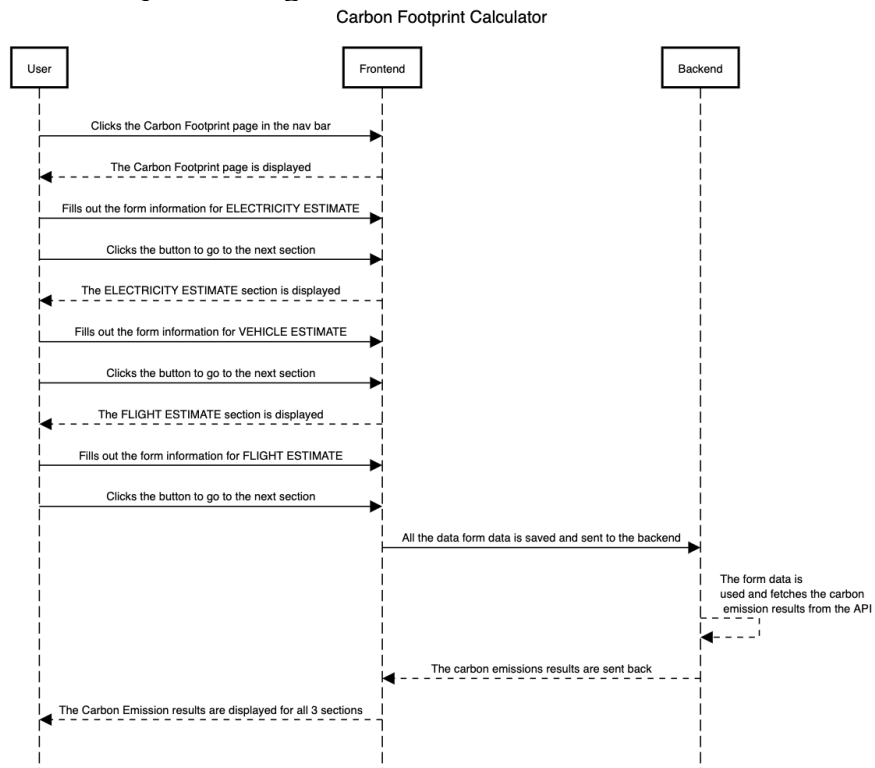
1. Class Diagram



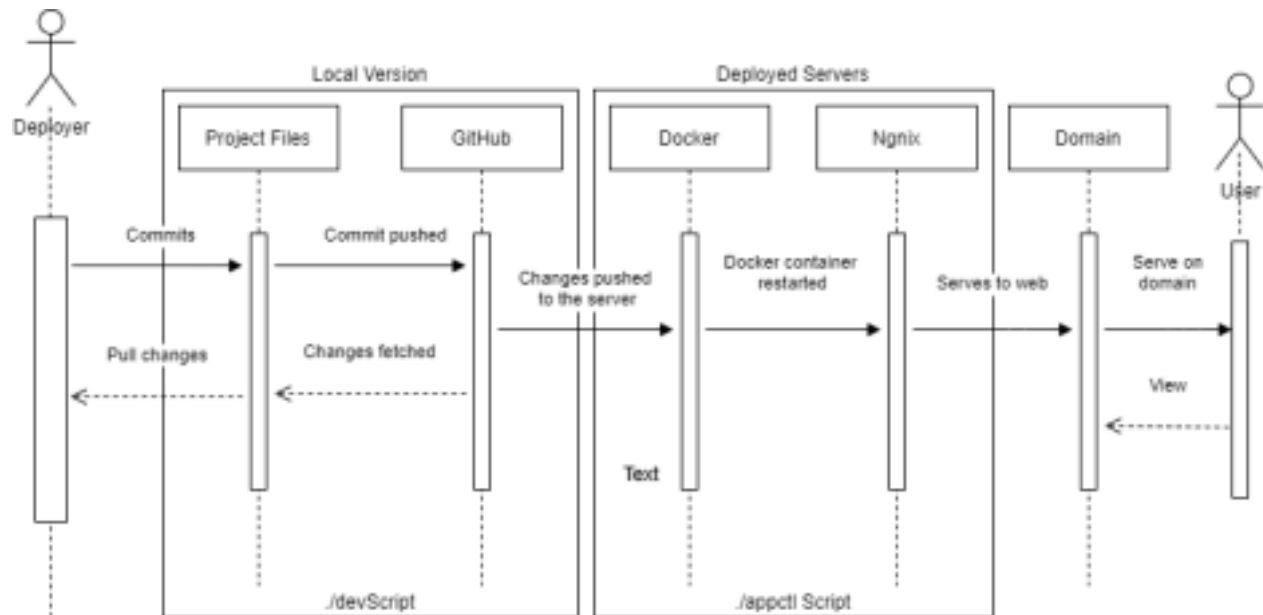
2. Use Case Diagram



3. Sequence Diagram



4. Development Cycle Sequence Diagram

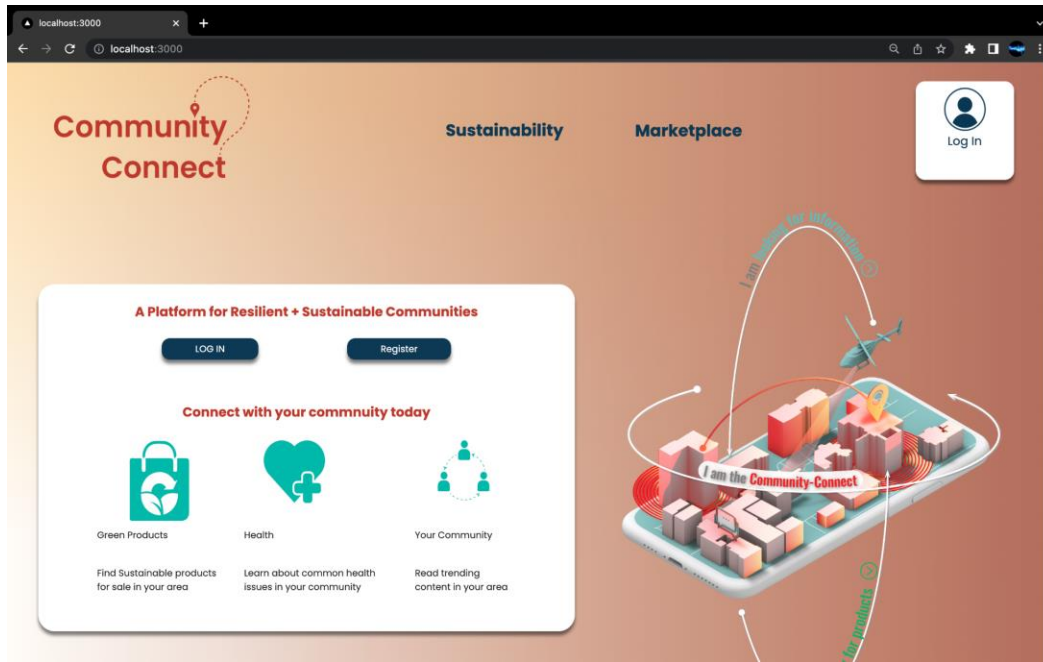


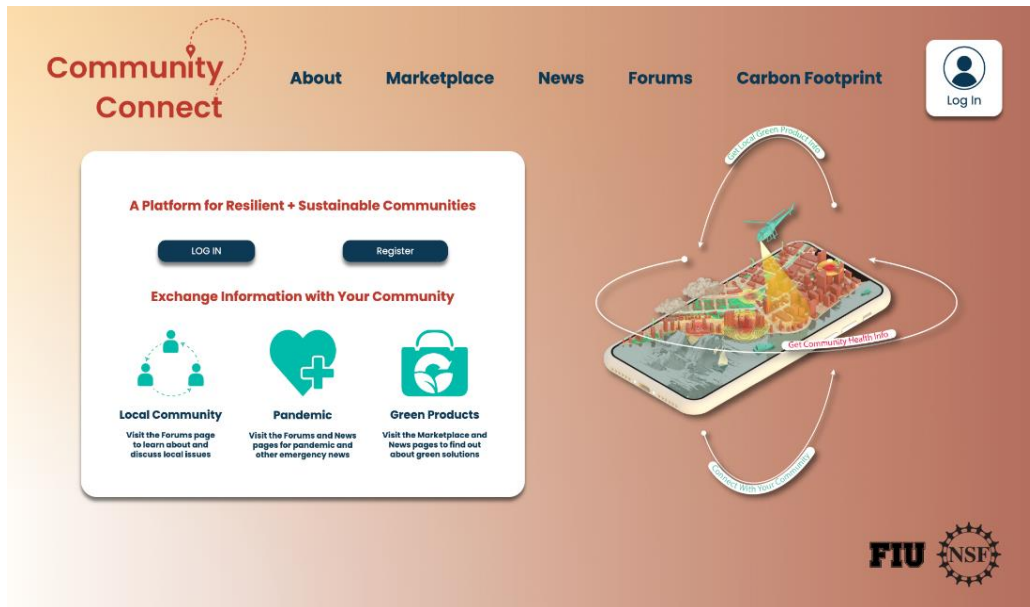
Appendix B - User Interface Design

1. Home Page

November 30th, 2022

Page 29 of 49

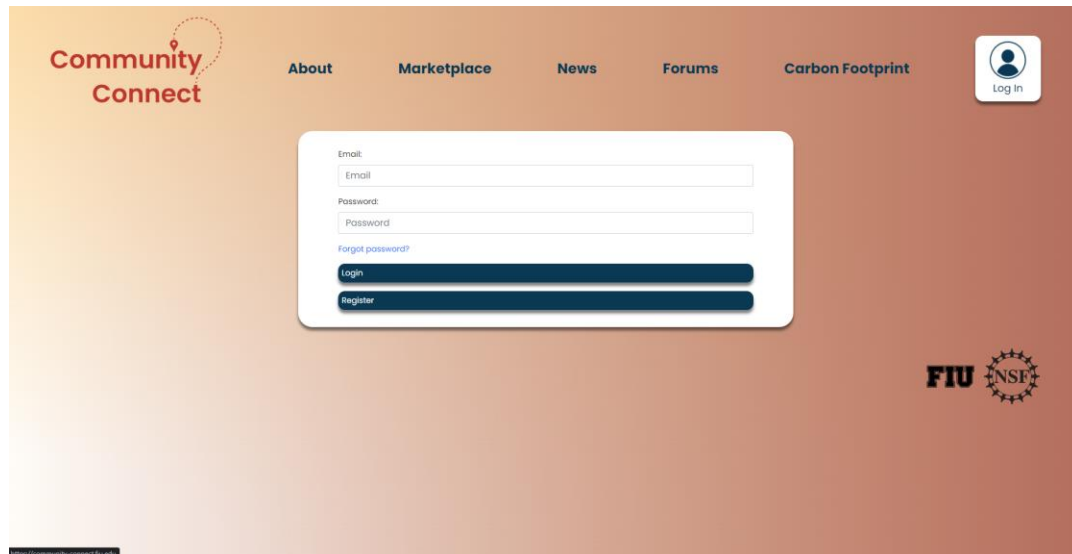
Old design:New design:



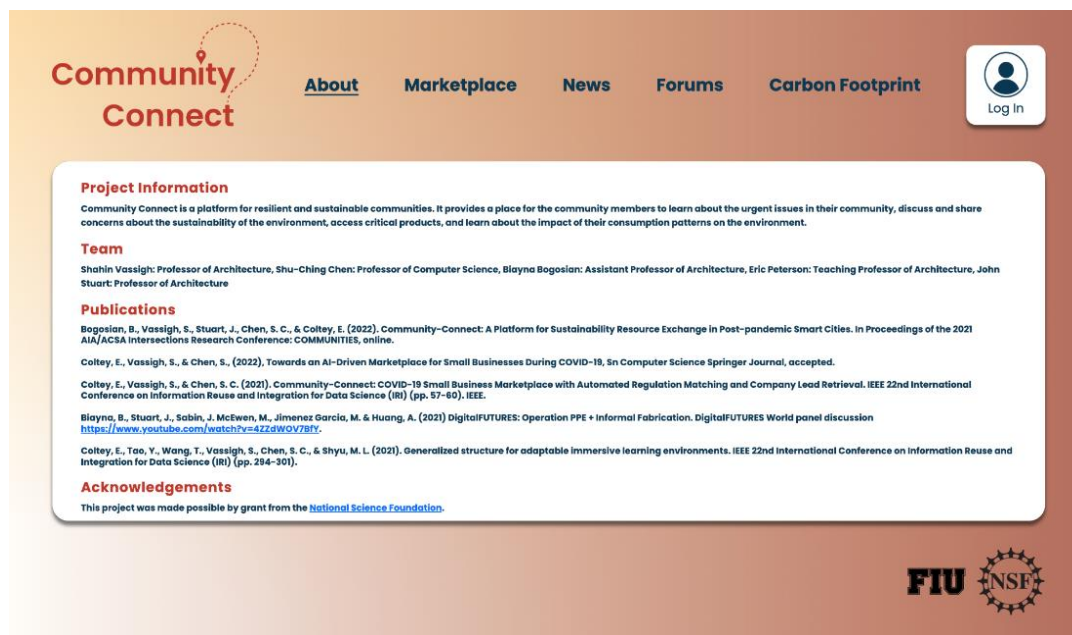
2. Register Page (zoomed out to fit page)

The image shows a registration form for 'Community Connect'. The form is titled 'Create Account' and includes fields for Email, Password, First Name, Last Name, Last Year (dropdown), and Birthdate (date picker). Below these are gender options (Female, Male, Non-Binary, Other) and address fields (Address Line 1, Address Line 2, Zip Code, City, State). There is a checkbox for 'Only use letter case abbreviations' and a 'Sign Up' button. The form also includes a list of interests: Pandemic, Climate Change, Green Living and Environment, Energy and Environment, and Sustainability and Innovation. The footer includes the FIU and NSF logos.

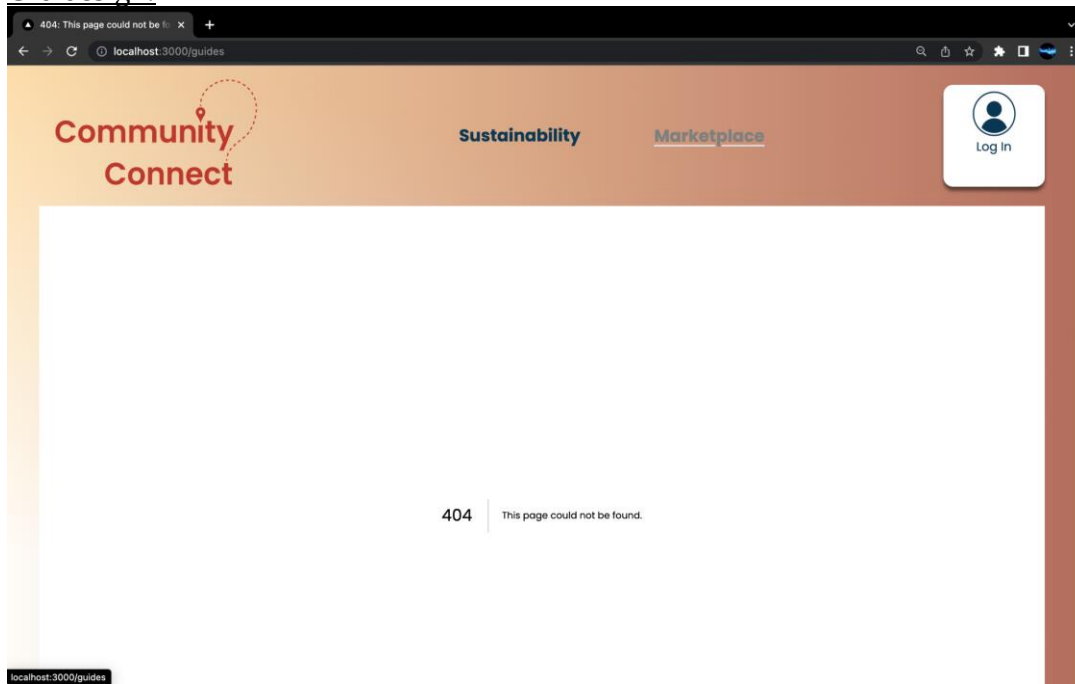
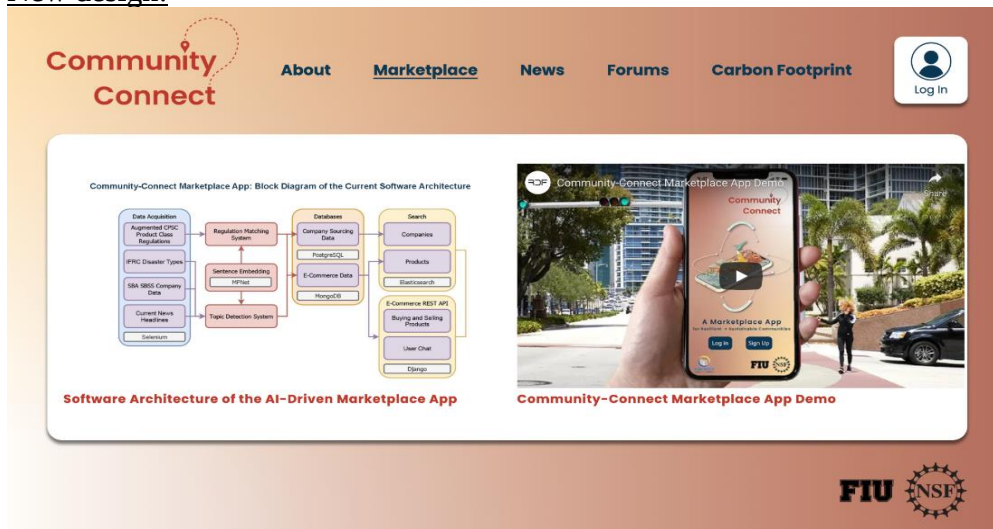
3. Login Page

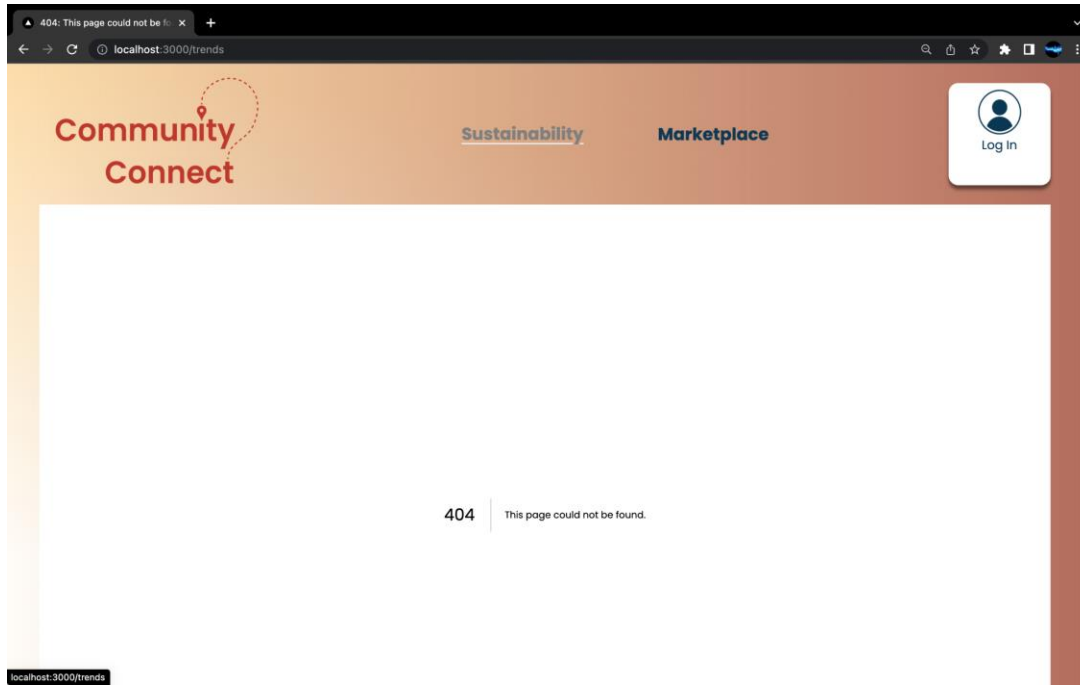
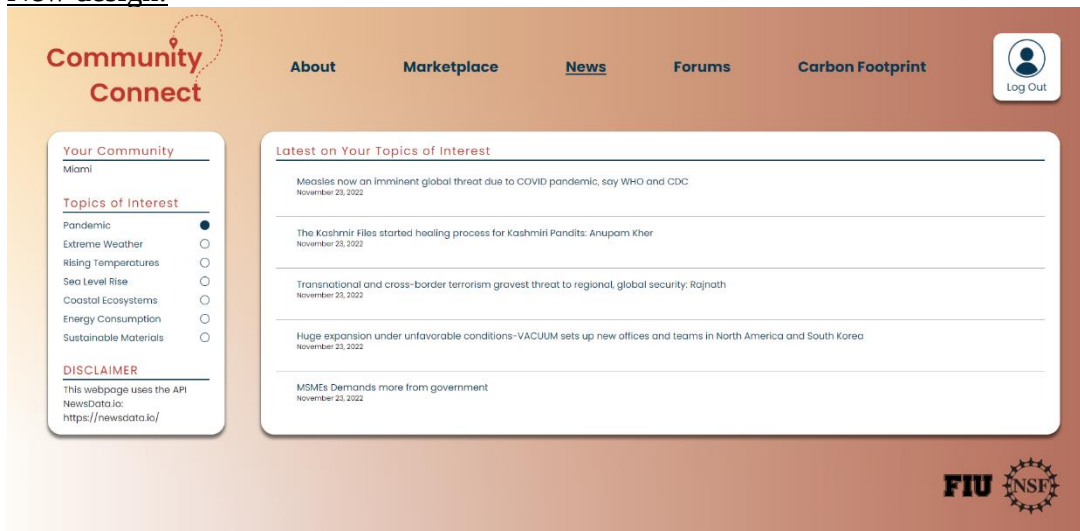


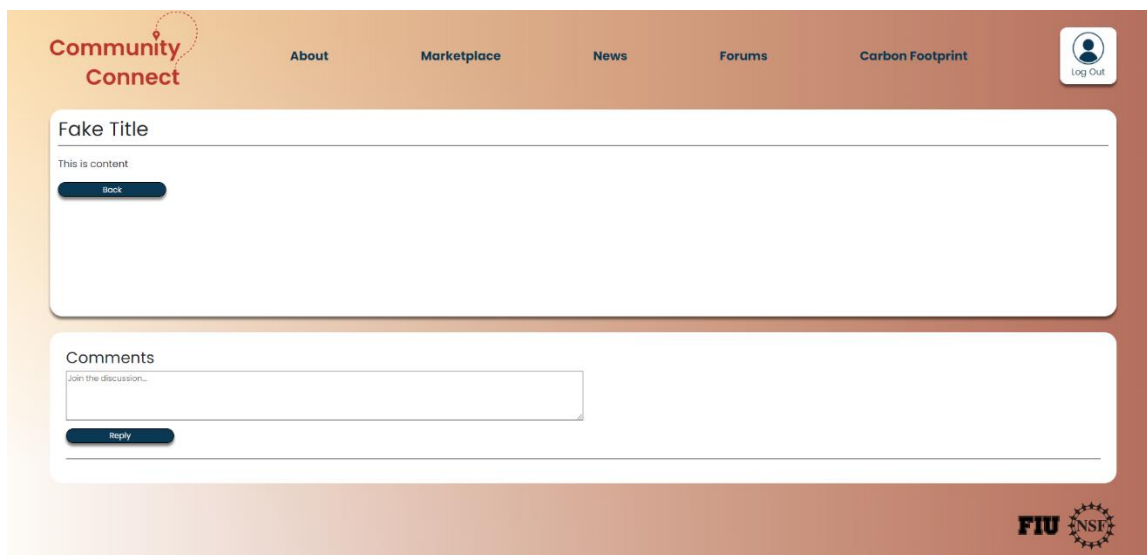
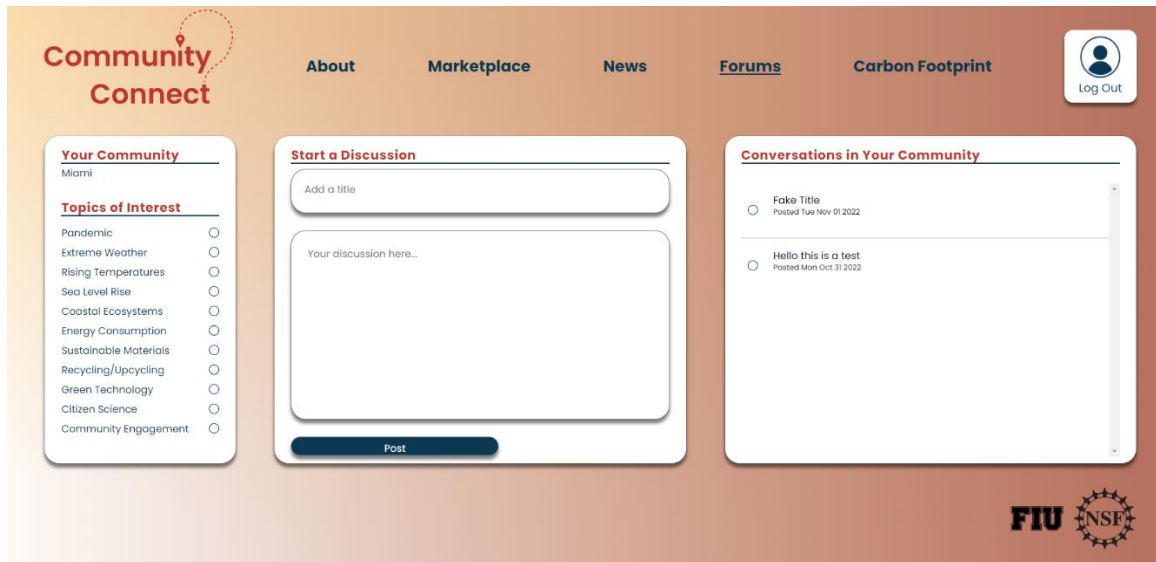
4. About Page



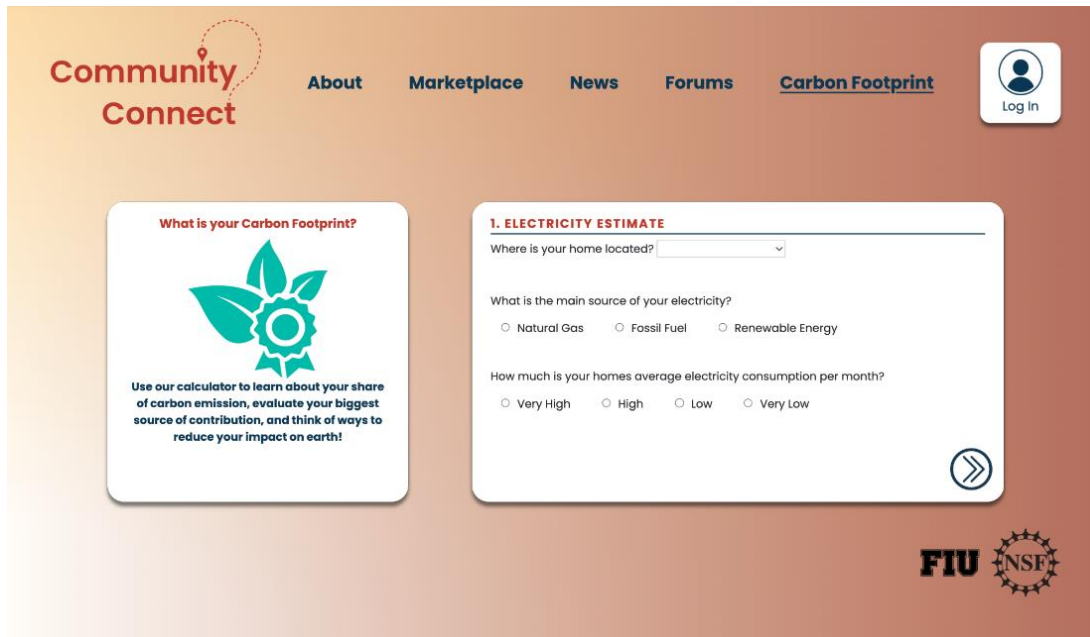
5. Marketplace Page

Old design:New design:**6. News Page**

Old design:New design:**7. Forums Page**



8. Carbon Footprint Page



The image shows a web interface for the 'Community Connect' Carbon Footprint Calculator. The header features the 'Community Connect' logo on the left and navigation links for 'About', 'Marketplace', 'News', 'Forums', and 'Carbon Footprint' in the center. A 'Log In' button is on the right. The main content area is divided into two white boxes. The left box, titled 'What is your Carbon Footprint?', contains a green leaf and gear icon and text encouraging users to use the calculator to learn about their carbon emission share and reduce their impact. The right box, titled '1. ELECTRICITY ESTIMATE', contains a form with three questions: 'Where is your home located?' (a dropdown menu), 'What is the main source of your electricity?' (radio buttons for Natural Gas, Fossil Fuel, and Renewable Energy), and 'How much is your homes average electricity consumption per month?' (radio buttons for Very High, High, Low, and Very Low). A double arrow icon is at the bottom right of the form. The footer on the right shows the 'FIU' logo and the 'NSF' logo.

Community Connect

[About](#) [Marketplace](#) [News](#) [Forums](#) [Carbon Footprint](#) [Log In](#)

What is your Carbon Footprint?

Use our calculator to learn about your share of carbon emission, evaluate your biggest source of contribution, and think of ways to reduce your impact on earth!

1. ELECTRICITY ESTIMATE

Where is your home located?

What is the main source of your electricity?

☐ Natural Gas ☐ Fossil Fuel ☐ Renewable Energy

How much is your homes average electricity consumption per month?

☐ Very High ☐ High ☐ Low ☐ Very Low

FIU **NSF**

Appendix C - Sprint Review Reports

Sprint Review - Sprint 1

November 30th, 2022

Page 36 of 49

Sprint Review Meeting Minutes

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zachary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadel, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez

Start time: 6:00 PM

End time: 6:15 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- <UserStory1>
- <UserStory2>
- <UserStory3>
- <UserStory4>
- <UserStory5>
- <UserStory6>
- <UserStory7>
- <UserStory8>

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- N/A

Sprint Review - Sprint 2**Sprint Review Meeting Minutes**

November 30th, 2022

Page 37 of 49

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zachary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadel, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez
Start time: 6:00 PM
End time: 6:15 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- <News1>
- <News2>
- <Forum1>
- <Forum2>

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- <Login1>
- <Login2>

Sprint Review - Sprint 3

Sprint Review Meeting Minutes

November 30th, 2022

Page 38 of 49

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zachary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadel, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez
Start time: 6:00 PM
End time: 6:15 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- <Forum1>
- <Forum2>

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- <Login1>
- <Login2>

Sprint Review - Sprint 4

Sprint Review Meeting Minutes

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zachary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadel, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez
Start time: 6:00 PM
End time: 6:15 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- <Forum1>
- <Forum2>
- <Forum3>

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- <Login1>
- <Login2>

Sprint Review - Sprint 5

Sprint Review Meeting Minutes

November 30th, 2022

Page 40 of 49

Attendees: Hunter Badillo, Michael Olivero, Miguel Ferreiro, Raul Gutierrez Niubo, Eduardo Lopez, Natasha Rodriguez, Zachary Santana, Kenny Afonin, Daniela Agueros, Michael Banegas, Jasen Cadel, Samuel Perez, Felix Rodriguez, Miguel Sablan, Castor Velasquez
Start time: 6:00 PM
End time: 6:15 PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

- <Frontend Forum Implementation>
- <Frontend Forum Design>
- <Backend Forum Database Schema>
- <Backend Forum Endpoints>
- <Frontend/Backend connecting>

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- <Login1>
- <Login2>

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents

Installation

Community Connect Prototype (skip all docker commands for now)

Required Services

- [Git](#)
- [Node v16.13.0](#)
- [Yarn](#)
- [Python](#)
- [Docker](#)
- [Docker-Compose](#) only if you're using Linux
- [Firefox Browser](#)
- [PostgreSQL 13+](#)

Structure of the project

- Postgres database
- [Django Rest Framework](#) backend which is a rest api Note: it is different from default Django tools
- [Nextjs](#) / [React](#) Front-end
- Frontend and Backend are connected using a javascript library called [axios](#)
- Docker is used to containerize and deploy

highly recommend to look through the documentation and understand tools as a beginner
Service installation guides for: | MacOS | Windows | Linux |

Running Project Locally

Once you've installed all of the required services, open a **Terminal Window** and follow these instructions:

1. Clone the repository

HTTPS

```
$ git clone https://github.com/fiu-rdflab-communityconnect/Community-Connect.git
```

SSH

```
$ git clone git@github.com:fiu-rdflab-communityconnect/Community-Connect.git
```

2. Navigate to the project's root directory (i.e. where package.json, docker-compose.yml, and .dockerignore files are located)

```
$ cd Community-Connect
```

3. Install node dependencies for the Frontend

```
$ yarn front:i
```

Windows Backend Installation Instructions

4. Create a virtual environment for the backend.

```
$ cd backend
```

```
$ py -m venv VE
```

5. Activate the virtual environment.

```
$ VE\Scripts\activate
```

6. Install the dependencies in the environment.

```
$ py -m pip install -r requirements.txt
```

```
$ py -m pip install TK
```

If it tells you that you are missing a module, install that module using python.

7. Migrate the database models.

```
$ py manage.py makemigrations
```

```
$ py manage.py migrate
```

Mac/Linux Backend Installation Instructions

4. Create a virtual environment for the backend.

```
$ cd backend/
```

```
$ python3 -m venv VE
```

5. Activate the virtual environment.

```
$ source VE/bin/activate
```

6. Install the dependencies in the environment.

```
$ python3 -m pip install -r requirements.txt
```

```
$ python3 -m pip install TK
```

If it tells you that you are missing a module, install that module using python.

7. Migrate the database models.

```
$ python manage.py makemigrations
```

```
$ python manage.py migrate
```

Running the project locally (from the root directory, no environment)

Running the frontend

```
$ cd frontend
```

```
$ npm run dev
```

Running the backend

Windows

```
$ cd backend
```

```
$ VE\Scripts\activate
```

```
$ py manage.py runserver
```

Mac

```
$ cd backend/
```

```
$ source VE/bin/activate
```

```
$ python manage.py runserver
```

Deployment through Ocelot using docker

This section was the old method of testing the backend, but it is now only in use for deployment.

4. Build the docker image (skip for now)

```
$ docker-compose build
```

5. Make and migrate database models

This builds the DB models:

```
$ docker-compose run web python manage.py makemigrations
```

After the top command finishes executing, run the following:

```
$ docker-compose run web python manage.py migrate
```

6. Create an admin user for the Backend (skip for now)

```
$ docker-compose run web python manage.py createsuperuser
```

- You will be prompted to create an account with a username, email (not necessary), and password.

7. Change your Windows settings/MacOs settings in your host file (skip for now)

This change will change the web url community-connect.fiu.edu to your local machine

Run Notepad as an administrator

Windows directory: MainDrive (exmp C:)\Windows\System32\drivers\etc\hosts

Open it as a notepad and at the bottom of the file add:

127.0.0.1 community-connect.fiu.edu

MacOs directory: open terminal and type

```
$ sudo nano /etc/hosts
```

At the end of the file add

127.0.0.1 community-connect.fiu.edu

then click control + X to save and press y and enter and then enter again to accept changes and save the file

8. Run the project (skip docker commands)

To run with live logs: (skip)

```
$ docker-compose up
```

To run in detached mode (no logs): (skip)

```
$ docker-compose up -d
```

To run backend server:

```
cd Community-Connect\backend python manage.py runserver
```

To run frontend:

```
cd Community-Connect\frontend npm run dev
```

- Once completed, you should be able to go to `http://localhost:3000` and view the website
- You can also go to `http://localhost:8000/admin` to access the admin page of the backend service

***Note:** It may take several seconds to load the first page.

9. Shutting down the project

If you're running the project with the live logs, you can press CTRL+C to shutdown the project, followed by entering the following command:

This will remove all declared and anonymous volumes that were attached to the containers

```
$ docker-compose down -v
```

- If you're running in detached mode, just run command above in the root directory of the project

Setting up the Backend Development Environment (Outdated)

This section is only necessary if you are working in the Backend so you can get IntelliSense support from the used libraries.

1. Install pipenv

On Linux and Windows, run:

```
$ pip3 install pipenv
```

On MacOS, run:

```
$ brew install pipenv
```

2. Enter the shell environment

```
$ pipenv shell
```

3. Installing all required packages

```
$ pip3 install -r backend/requirements.txt
```

Shortcomings

The project approach shifted in the Fall semester. There were new pages that had to be developed and modified to take full advantage of the design improvements from last semester.

The NewsData API was able to be used for free in deployment, but it does not offer enough free calls to allow us to properly filter through many different topics of interest. Realistically, if a user visits the news page and cycles through different topics, they could consume anywhere from 30 to 100 API calls in one sitting. This is unrealistic for a final version of this product and a subscription might be the only path to keep using an API.

The forum page was implemented pretty well, but there is still room to make formatting adjustments for comments under forum posts. At the moment comments appear as plain text lines under the main post, and there is no formatting to see who wrote these comments and when. Additionally, there needs to be a place for users to properly attach media to their forum post. The backend has the capabilities to store media attachments, but the forum page needs to be properly formatted to allow for that.

The final major component that could be improved is the user sign and user functionality of the website. We added a functionality where users cannot create forum posts without being signed into the site, but users are unable to edit any information related to their account. If the user moves, they cannot change their address and they cannot update profile pictures to allow for a more user friendly experience. This is something that students in the future will probably have to work on because this is a priority of the product owner, and the feature is behind on development. The only other errors are probably minimal to these that I listed so far so these features should take up most of the development time.

Wishlist

- Subscription to use News API key for deployment.
- Further develop Forums page
- Refactor code to be more efficient/clean
- General UI/UX improvements

REFERENCES

<https://docs.docker.com>

<https://www.django-rest-framework.org>

<https://reactjs.org>

<https://nextjs.org/>