

*A Florida International University
Knight's Foundation School of Computing and Information
Sciences*

Software Engineering Focus

Final Deliverable

Mitigating the Impact of Covid 19 - Community Connect
Fall 2021

Team Members: Adrian Gonzalez, Niurbelis Gonzalez, Sebastian Gomez, Shaoting Wu, Chábeli Castano Arango, Narmeen Kibria, Manuel Kamboykos, and David Garcia

Product Owner(s): Shahin Vassigh

Mentor(s): Marcus Muñoz

Instructor: Dr. 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 NON-INFRINGEMENT. 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 serves as an informative presentation to gain an adequate understanding of the fourth iteration of the Community Connect website.

The previous iteration of this project focused on building upon the foundation of a simple forum by concentrating on the frontend aspect. This allowed a more user-friendly experience when navigating through the website. Despite the efforts made prior, there existed a lack of features to enhance the versatility of Community Connect.

For this iteration, the goal was to perform a complete redesign of the user-interface, add tools such as a carbon footprint calculator, create more features to gain better utility such as trends based on location, overhaul the project structure for a better developer experience, and continue building upon the features of the backend. The team was able to accomplish these features as well as continue to improve the user interface within six sprints.

Table of Contents

Introduction	6
Current System	7
Purpose of New System	8
User Stories	9
Implemented User Stories	9
Pending User Stories	10
Project Plan	11
Hardware and Software Resources Patterns	11
Sprints Plan	12
Sprint 1	12
Sprint 2	14
Sprint 3	17
Sprint 4	19
Sprint 5	21
Sprint 6	23
System Design	24
Architectural Patterns	24
System and Subsystem Decomposition	24
Deployment Diagram	28
Design Patterns	28
System Validation	29
Glossary	29
Appendix	31
Appendix A - UML Diagrams	31
Sequence Diagram	34
Development cycle sequence diagram	35
Appendix B - User Interface Design	36
Appendix C - Sprint Review Reports	41

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document, and other documents	51
Local Development Environment Setup Guide	51
For New Prototype	51
For Older Version of Community Connect	52
Running the Local Development Server	53
Deployed Development Server	53
SSH Login Information for Deployed Development Server	53
Deployed Development Server Usage	53
Deployed Production Server	54
SSH Login Information for Deployed Production Server	54
Deployed Production Server Note (IMPORTANT)	54
Production Email Configuration	55
Frontend Changes (CSS/JS)	55
Frontend Changes (HTML Templates)	56
Authentication Issues	56
Shortcomings/Wishlist Document	57
References	57

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 the COVID 19 health crisis. According to the Center for Disease Control and Prevention, only 32% of Americans have received their first dose of the COVID-19 vaccine at this time. The Community Connect website aims to raise the percentage of vaccinations by providing details on vaccination sites based on the user's location.

The research and development for this project consisted of utilizing data from multiple Google APIs to provide information most useful for a user inquiring about the Coronavirus. This Coronavirus data includes preventative measures, statistics, and featured articles based on location. Combining these elements of the platform serves as a solution for users to navigate through an ongoing pandemic.

Dr. Shahin Vassigh and her team designed Community Connect as a solution for users of all backgrounds to become more educated about the Coronavirus communicatively and reliably. The platform will use Google Trends, Maps, and Analytics to tailor the most advantageous learning experience for the user. This website consists of four principle features: guides on how to slow the spread of the Coronavirus, trending topics for the user's residential state, news articles from reliable media outlets, and a forum to discuss subjects related to COVID-19.

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

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 job was 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.

Purpose of New System

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.

User Stories

The following section provides the detailed user stories that were implemented in this iteration of the Miti 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 - 20 pts - As a developer I would like to redo the News page based on the new design
- SP2 - 20 pts - As a developer I would like to redo the Trends page based on the new design
- SP2 - 20 pts - As a developer I would like to redo the Guides page based on the

new design

- SP2 - 10 pts - As a developer I would like to redo the Home page based on the new design
- SP3 - 20 pts - As a developer I would like to create a REST API for the Thread class as described in the documentation
- SP3 - 20 pts - As a developer I would like to create a REST API for the Post class as described in the documentation
- SP3 - 20 pts - As a developer I would like to create a REST API for the User class as described in the documentation
- SP4 - 20 pts - As a developer I would like to create a frontend service that can consume the Django REST API for Threads
- SP4 - 20 pts - As a developer I would like to create a frontend service that can consume the Django REST API for the User
- SP5 - 10 pts - As a developer I would like to add a forgot password button in the login page
- SP5 - 10 pts - As a developer I would like to add a page for the Carbon Footprint Calculator

Pending User Stories

- SP6 - 20pts - As a developer I would like to create a REST API for the Polls class as described in the documentation.
- SP6 - 20pts - As a developer I would like to create a REST API for the Category class as described in the documentation.

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 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.

For Future Development:

- **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.
- **ElasticSearch:** Distributed, RESTful search and analytics engine capable of storing your data for lightning-fast search, fine-tuned relevancy, and powerful analytics that scale with ease.

- **Digital Ocean droplet:** A Linux VM instance hosted on a digital ocean website that serves as the development server hosting.
- **FIU Server:** Linux server on FIU premises that hosts actual, production versions of the application

Sprints Plan

Sprint 1

Sprint Planning Meeting Minutes:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 5:00 PM

End time: 6:00 PM

After discussion, the velocity of the team was estimated to be 7 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 complete the project installation guide. (1 hour)
- User Story 2 - As a developer I would like to familiarize with the project structure (1 hour)
- User Story 3 - As a developer I would like to familiarize with React (1 hour)
- User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows (2 hours)
- User Story 5 - As a developer I would like to implement a better User Interface for the Home Page (2 hours)

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

- Niurbelis Gonzalez
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- Sebastian Gomez
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- Shaoting Wu

- User Story 1 - As a developer I would like to complete the project installation guide.
- User Story 2 - As a developer I would like to familiarize with the project structure.
- User Story 3 - As a developer I would like to familiarize with React.
- Adrian Gonzalez
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- Chábeli Castano Arango
 - User Story 1 - As a developer I would like to complete the project installation guide.
 - User Story 2 - As a developer I would like to familiarize with the project structure.
 - User Story 3 - As a developer I would like to familiarize with React.
- Narmeen Kibria
 - User Story 2 - As a developer I would like to familiarize with the project structure (1 hour)
 - User Story 3 - As a developer I would like to familiarize with React (1 hour)
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page (2 hours)
- Manuel Kamboykos
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- David Garcia
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 1 - As a developer I would like to complete the project installation guide. (1 hour)
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows (2 hours)

Planned User Stories:

- As a developer I would like to complete the project installation guide.
- As a developer I would like to familiarize with the project structure (1 hour)
- As a developer I would like to familiarize with React (1 hour)
- As a developer I would like to research a work around solution to run Docker in Windows

- As a developer I would like to implement a better User Interface for the Home Page

Completed User Stories:

- As a developer I would like to complete the project installation guide.
- As a developer I would like to familiarize with the project structure (1 hour)
- As a developer I would like to familiarize with React (1 hour)
- As a developer I would like to research a work around solution to run Docker in Windows
- As a developer I would like to implement a better User Interface for the Home Page

Sprint 2

Sprint Planning Meeting Minutes:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 15 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 redo the Guides page based on the new design (5 hours)
- User Story 2 - As a developer I would like to redo the Trends page based on the new design (5 hours)
- User Story 3 - As a developer I would like to redo the News page based on the new design (5 hours)

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

- Niurbelis Gonzalez
 - User Story 3 - As a developer I would like to redo the News page based on the new design
 - User Story 1 - As a developer I would like to redo the Guides page based on the new design

- User Story 2 - As a developer I would like to redo the Trends page based on the new design
- Sebastian Gomez
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- Shaoting Wu
 - User Story 3 - As a developer I would like to redo the News page based on the new design (5 hours)
- Adrian Gonzalez
 - User Story 1 - As a developer I would like to redo the Guides page based on the new design
 - User Story 3 - As a developer I would like to redo the News page based on the new design
 - User Story 2 - As a developer I would like to redo the Trends page based on the new design
- Chábeli Castano Arango
 - User Story 3 - As a developer I would like to redo the News page based on the new design
 - User Story 1 - As a developer I would like to redo the Guides page based on the new design
 - User Story 2 - As a developer I would like to redo the Trends page based on the new design
- Narmeen Kibria
 - User Story 1 - As a developer I would like to redo the Guides page based on the new design
- Manuel Kamboykos

- User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows
- User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- David Garcia
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 1 - As a developer I would like to complete the project installation guide. (1 hour)
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows (2 hours)

Planned User Stories:

- As a developer I would like to redo the Guides page based on the new design
- As a developer I would like to redo the Trends page based on the new design
- As a developer I would like to redo the News page based on the new design

Completed User Stories:

- As a developer I would like to redo the Guides page based on the new design
- As a developer I would like to redo the Trends page based on the new design
- As a developer I would like to redo the News page based on the new design

Sprint 3

Sprint Planning Meeting Minutes:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 35 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 create a REST API for the Thread class as described in the documentation. (7 hours)
- User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation. (7 hours)
- User Story 3 - As a developer I would like to create a REST API for the Polls class as described in the documentation. (7 hours)
- User Story 4 - As a developer I would like to create a REST API for the Category class as described in the documentation. (7 hours)
- User Story 5 - As a developer I would like to create a REST API for the Post class as described in the documentation. (7 hours)

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

- Niurbelis Gonzalez
 - User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation.

- User Story 1 - As a developer I would like to create a REST API for the Thread class as described in the documentation.
- User Story 4 - As a developer I would like to create a REST API for the Category class as described in the documentation.
- User Story 3 - As a developer I would like to create a REST API for the Polls class as described in the documentation.
- User Story 5 - As a developer I would like to create a REST API for the Post class as described in the documentation.
- Sebastian Gomez
 - User Story 5 - As a developer I would like to create a REST API for the Post class as described in the documentation.
- Shaoting Wu
 - User Story 5 - As a developer I would like to create a REST API for the Post class as described in the documentation. (7 hours)
- Adrian Gonzalez
 - User Story 1 - As a developer I would like to create a REST API for the Thread class as described in the documentation.
 - User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation.
 - User Story 3 - As a developer I would like to create a REST API for the Polls class as described in the documentation.
- Chábeli Castano Arango
 - User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation.
 - User Story 1 - As a developer I would like to create a REST API for the Thread class as described in the documentation.
- Narmeen Kibria
 - User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation. (7 hours)

- Manuel Kamboykos
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- David Garcia
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 1 - As a developer I would like to complete the project installation guide. (1 hour)
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows (2 hours)

Planned User Stories:

- As a developer I would like to create a REST API for the Thread class as described in the documentation. (7 hours)
- As a developer I would like to create a REST API for the User class as described in the documentation. (7 hours)
- As a developer I would like to create a REST API for the Polls class as described in the documentation. (7 hours)
- As a developer I would like to create a REST API for the Category class as described in the documentation. (7 hours)
- As a developer I would like to create a REST API for the Post class as described in the documentation. (7 hours)

Completed User Stories:

- As a developer I would like to create a REST API for the Thread class as described in the documentation. (7 hours)
- As a developer I would like to create a REST API for the User class as described in the documentation. (7 hours)

- As a developer I would like to create a REST API for the Post class as described in the documentation. (7 hours)

Sprint 4

Sprint Planning Meeting Minutes:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 24 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 create a frontend service that can consume the Django REST API for Threads. (8 hours)
- User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- User Story 3 - As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)

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

- Niurbelis Gonzalez
 - User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for User.
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts.
 - User Story 1 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads.

- Sebastian Gomez
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Shaoting Wu
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Adrian Gonzalez
 - User Story 1 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
 - User Story 3 - As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Chábeli Castano Arango
 - User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for User (8 hours)
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Narmeen Kibria
 - User Story 1 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
- Manuel Kamboykos
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- David Garcia
 - User Story 2 -As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)

Planned User Stories:

- As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
- As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)

Completed User Stories:

- As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)

Sprint 5

Sprint Review Meeting Minutes

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 24 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 redo the Home page based on this design
- User Story 2 - As a User I would like forgot password in the Login Page.
- User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
- User Story 4 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- User Story 5 - As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)

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

- Niurbelis Gonzalez
 - User Story 1 - As a developer I would like to redo the Home page based on this design
 - User Story 5 - As a developer I would like to create a frontend service that can consume the Django REST API for User.

- User Story 4 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts.
- User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads.
- User Story 2 - As a User I would like forgot password in the Login Page.
- Sebastian Gomez
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Shaoting Wu
 - User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
- Adrian Gonzalez
 - User Story 1 - As a developer I would like to redo the Home page based on this design
 - User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
 - User Story 4 - As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)
 - User Story 5 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
 - User Story 2 - As a User I would like forgot password in the Login Page.
- Chábeli Castano Arango
 - User Story 1 - As a developer I would like to redo the Home page based on this design
 - User Story 5 - As a developer I would like to create a frontend service that can consume the Django REST API for User.
- Narmeen Kibria
 - User Story 1 - As a developer I would like to redo the Home page based on this design

- Manuel Kamboykos
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- David Garcia
 - User Story 2 -As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)

Planned User Stories:

- As a developer I would like to redo the Home page based on this design
- As a User I would like forgot password in the Login Page.
- As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
- As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)

Completed User Stories:

- As a developer I would like to redo the Home page based on this design
- As a User I would like forgot password in the Login Page.
- As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
- As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)

Sprint 6

Sprint Review Meeting Minutes

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 24 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 create a REST API for the Polls class as described in the documentation. (8 hours)
- User Story 2 - As a developer I would like to create a REST API for the Category class as described in the documentation. (8 hours)
- User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages) (8 hours)

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

- Niurbelis Gonzalez
 - User Story 2 - As a developer I would like to create a REST API for the Category class as described in the documentation.
 - User Story 1 - As a developer I would like to create a REST API for the Polls class as described in the documentation.
 - User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages)

- Sebastian Gomez
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
 - User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages)
- Shaoting Wu
 - User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages) (8 hours)
- Adrian Gonzalez
 - User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages)
 - User Story 2 - As a developer I would like to create a REST API for the Category class as described in the documentation.
 - User Story 1 - As a developer I would like to create a REST API for the Polls class as described in the documentation.
- Chábeli Castano Arango
 - User Story 2 - As a developer I would like to create a REST API for the Category class as described in the documentation.
 - User Story 1 - As a developer I would like to create a REST API for the Polls class as described in the documentation.
- Narmeen Kibria
 - User Story 1 - As a developer I would like to create a REST API for the Polls class as described in the documentation. (8 hours)
- Manuel Kamboykos
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- David Garcia

- User Story 2 -As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)

Planned User Stories:

- As a developer I would like to create a REST API for the Polls class as described in the documentation. (8 hours)
- As a developer I would like to create a REST API for the Category class as described in the documentation. (8 hours)
- As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages) (8 hours)

Completed User Stories:

- As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages) (8 hours)

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

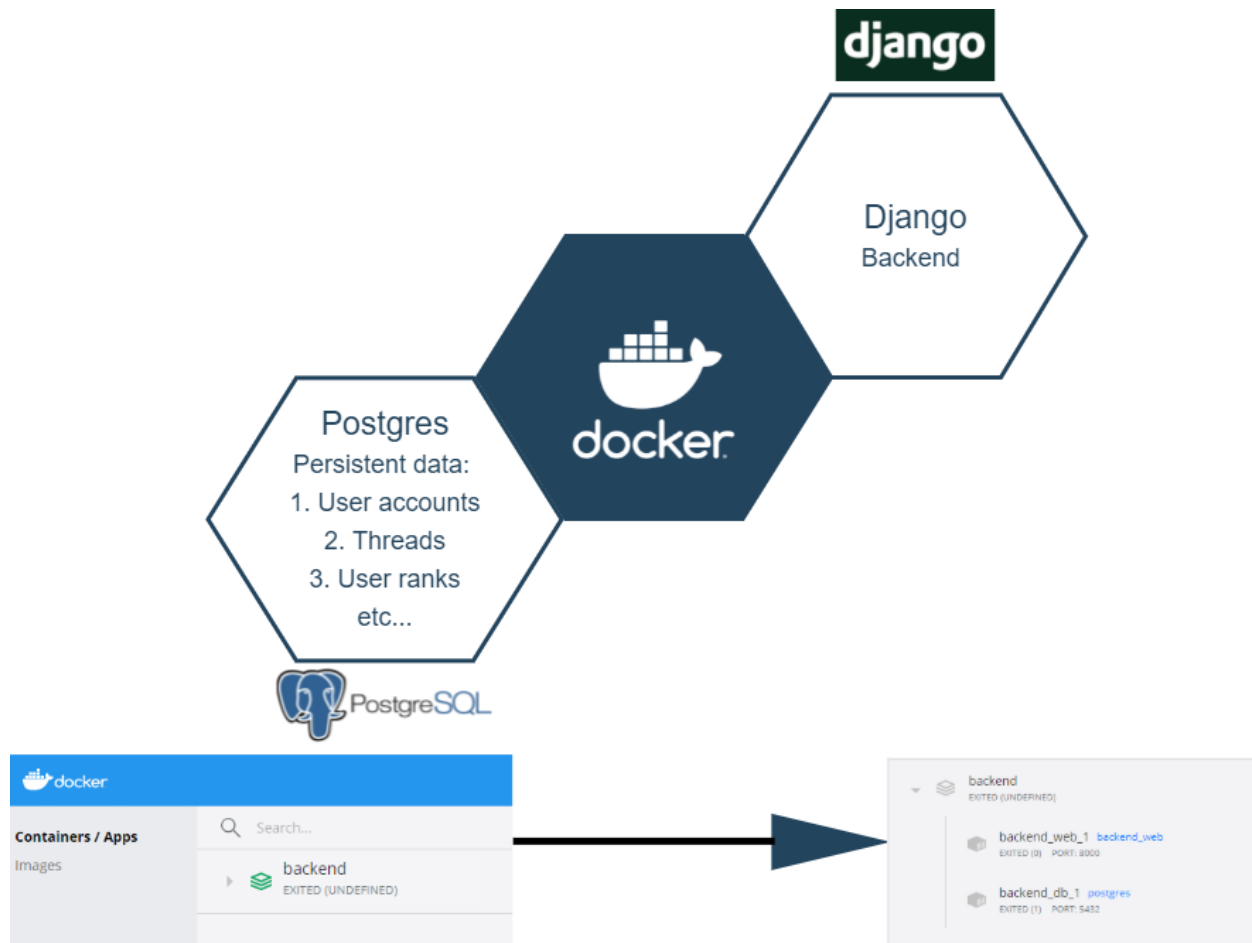
The new Community Connect was built from the ground up; thus, there is no used code. The original Community Connect used open-source Misago, which gave a lot of functionality out of the box but most of the functionality that it offered was not used. The new architecture utilizes React for the client and Django for the server while still using Docker. One of the main benefits of the new Community Connect is the scoped CSS, which was not possible using Misago. The original Community Connect website is built upon an open-source Misago project that provided Community Connect with the basic forum functionality. 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. Each part of the application has a separate Docker image and the application as the whole is composed in one big Docker container while running locally or 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 original Community Connect is heavily coupled. There is no distinction between the frontend and backend since some HTML was rendered with React and some with Django. In addition, the Misago template that was used contained a lot of unused features, which made it difficult to maintain and onboard new developers. The new architecture was created to improve on the following: 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. We also included several global style files to facilitate changes in the future such as color theme, button, and fonts. The backend uses Django and Postgres.

Back-end architecture tools



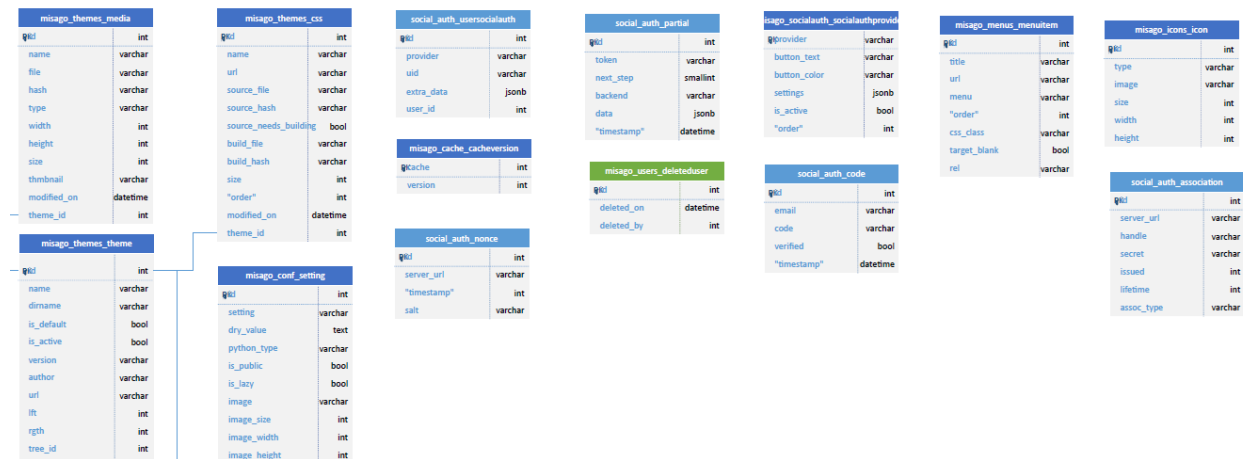
Front-end architecture tools

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

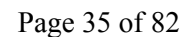
and data structure, the view is the user interface, and the controller accepts and converts the input.

Databases

The database structure will be fully implemented in upcoming semesters but without Misago.

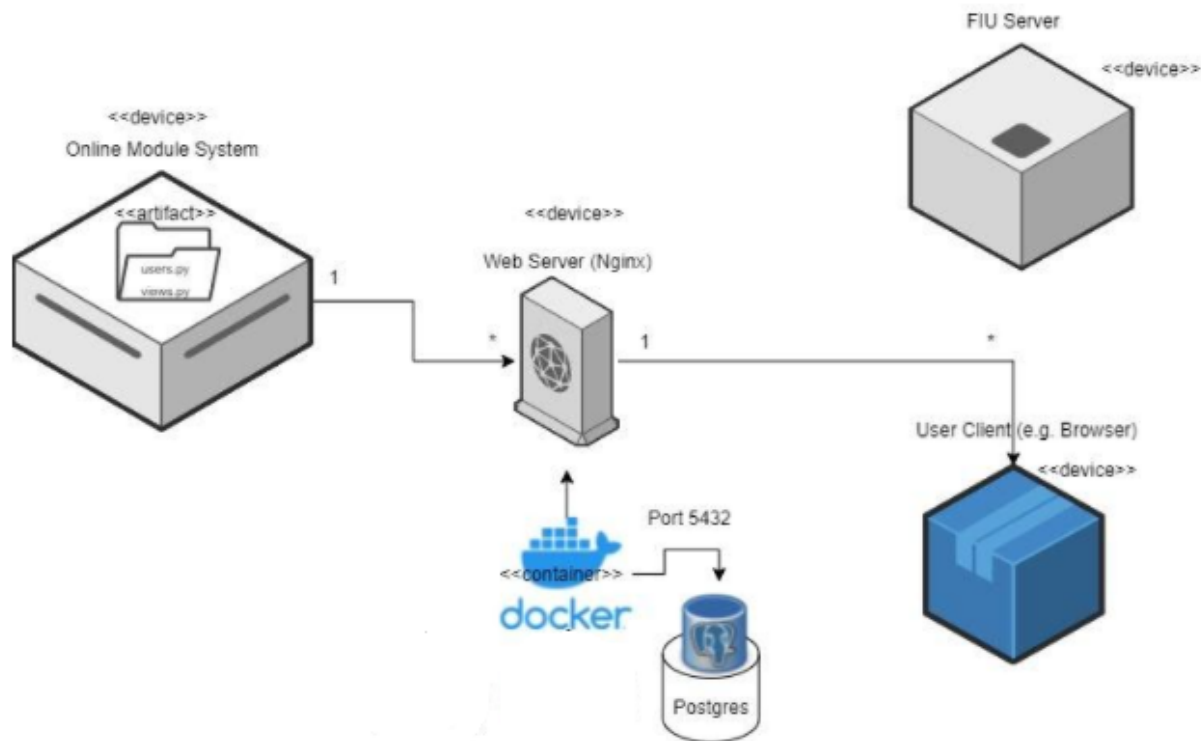


(Continued below)



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 an 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 previous iteration of Community Connect handled 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 of each of our forum applications. This will be worked on in the next iteration.

Glossary

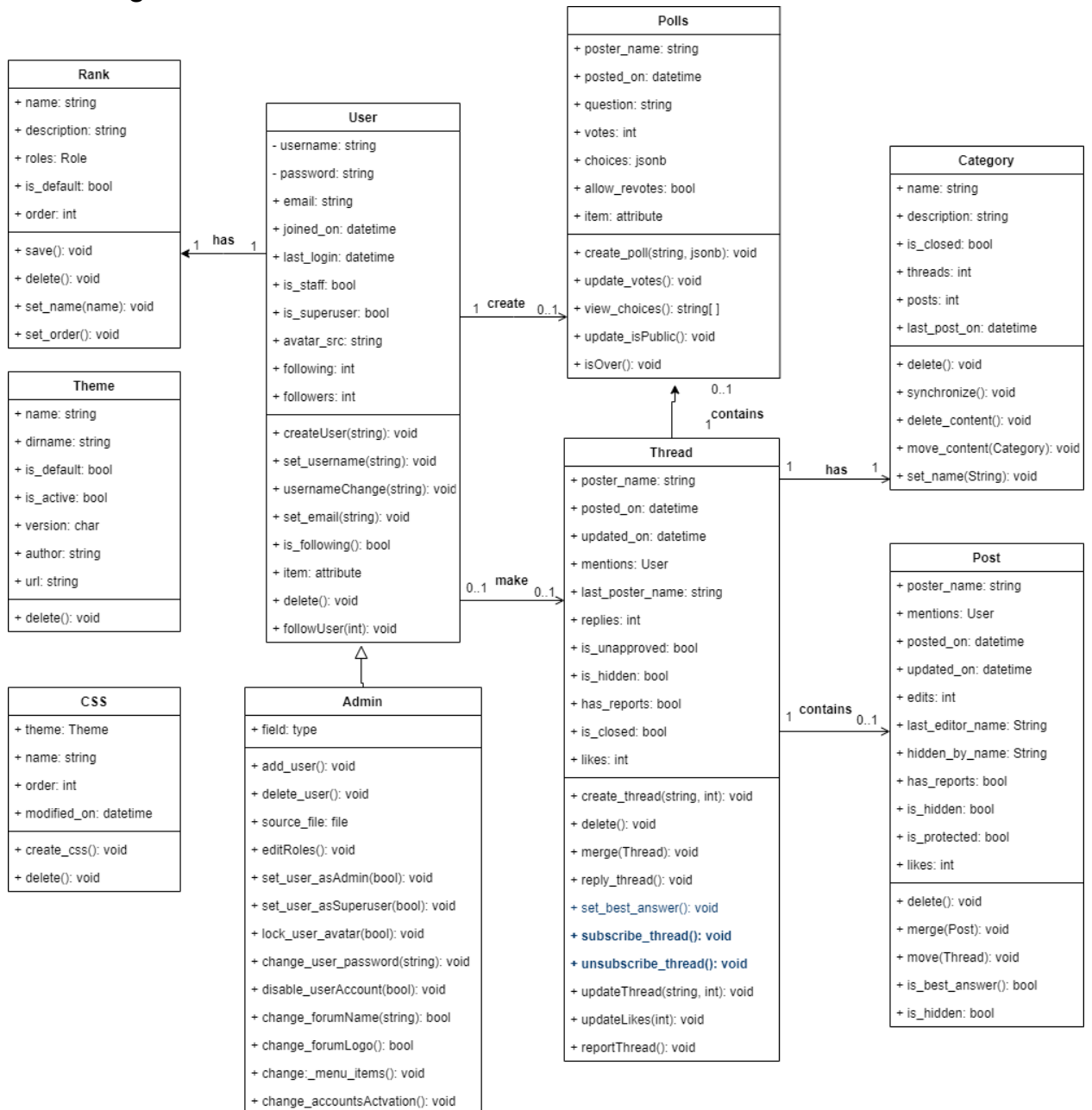
- **Framework:** A free and open-source platform of OS-level virtualization to host the needed images(parts) of the project.
- **API:** optional visualization tool for Docker images
- **Open-Source:** optional visualization
- **Docker:** A free and open-source platform of OS-level virtualization to host 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.
- **Digital Ocean droplet:** A Linux VM instance hosted on a digital ocean website that serves as the development server hosting.
- **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.
- **ElasticSearch:** Distributed, RESTful search and analytics engine capable of storing your data for lightning-fast search, fine-tuned relevancy, and powerful analytics that scale with ease.
- **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.

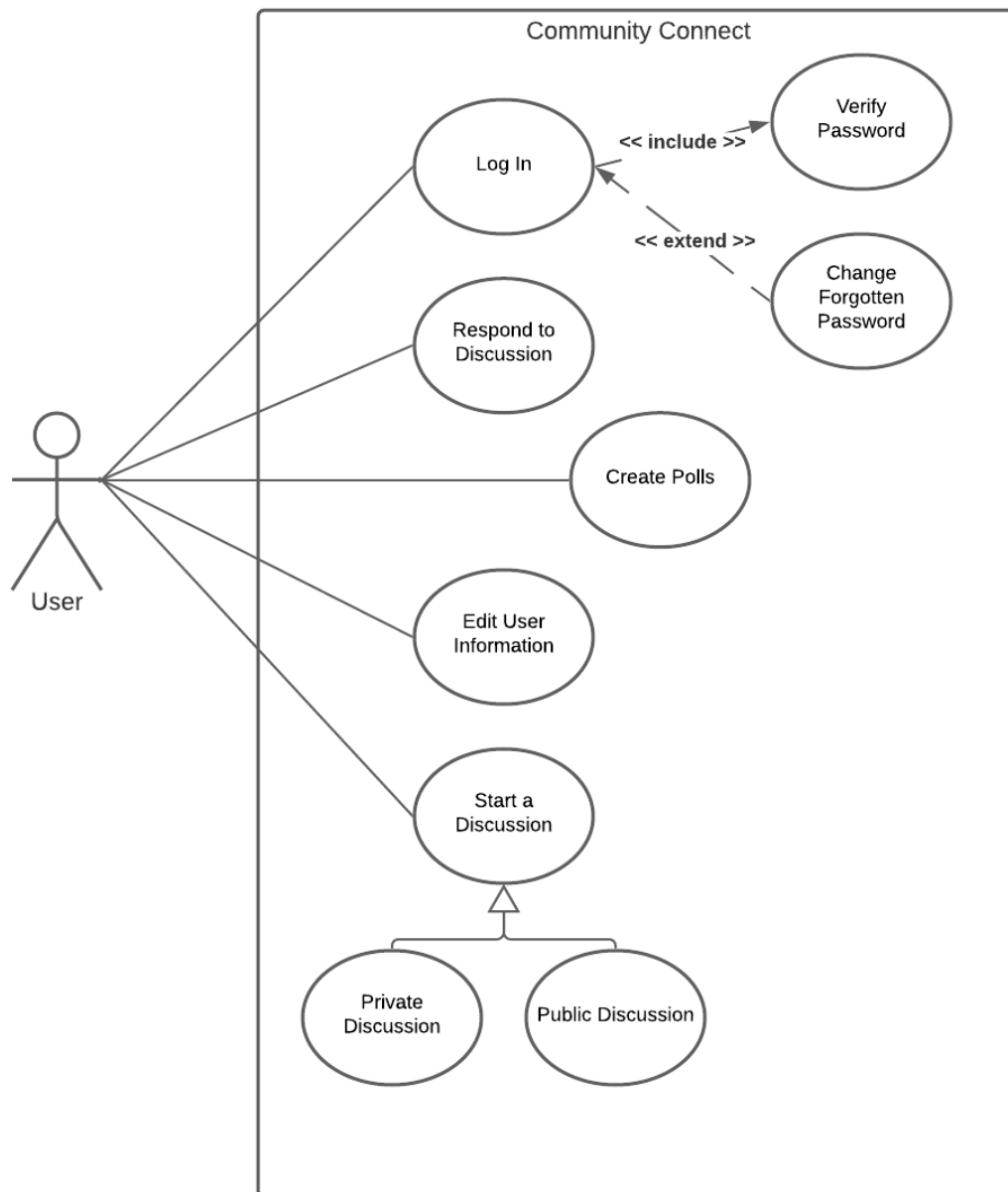
Appendix

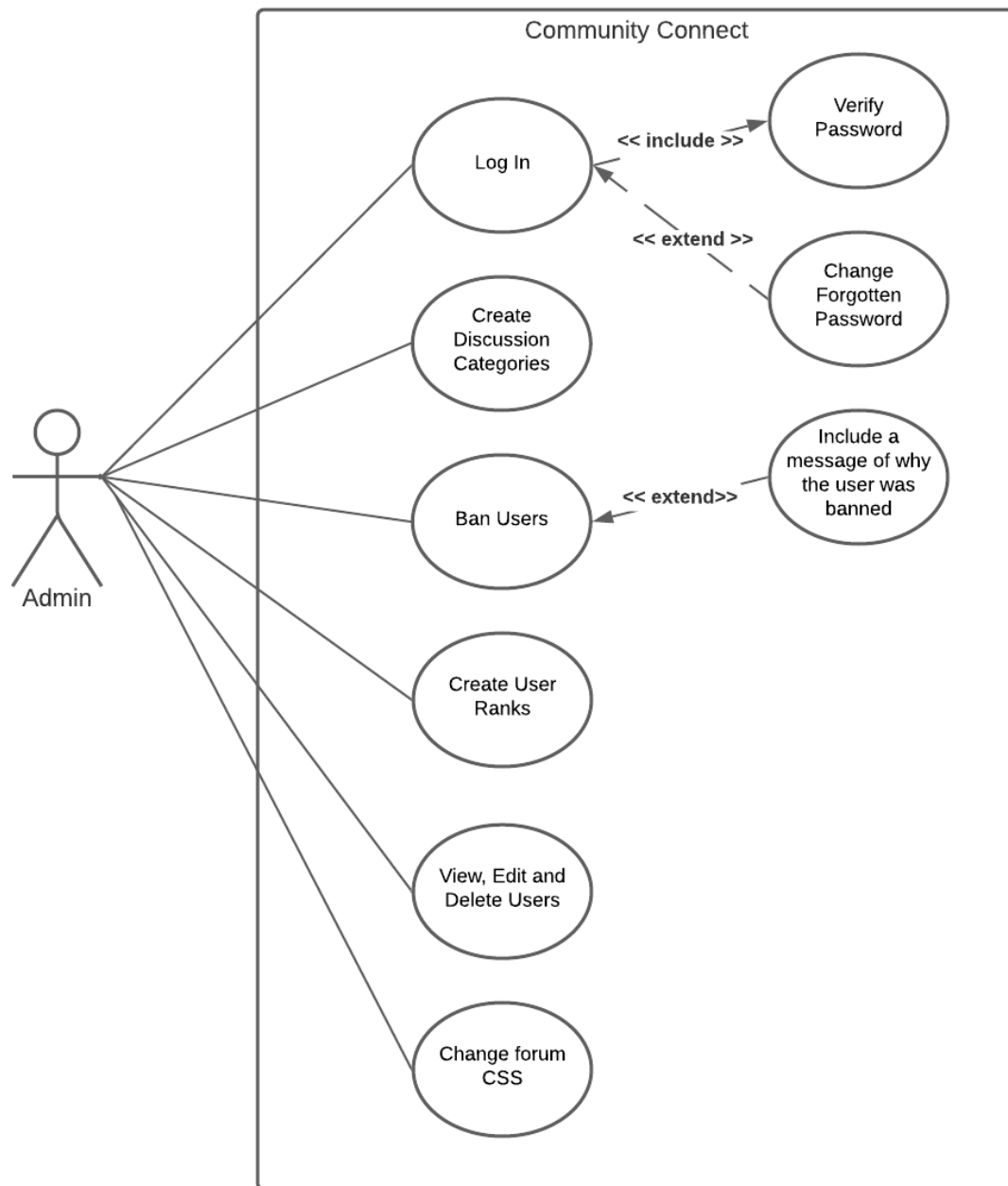
Appendix A - UML Diagrams

1. Class Diagram



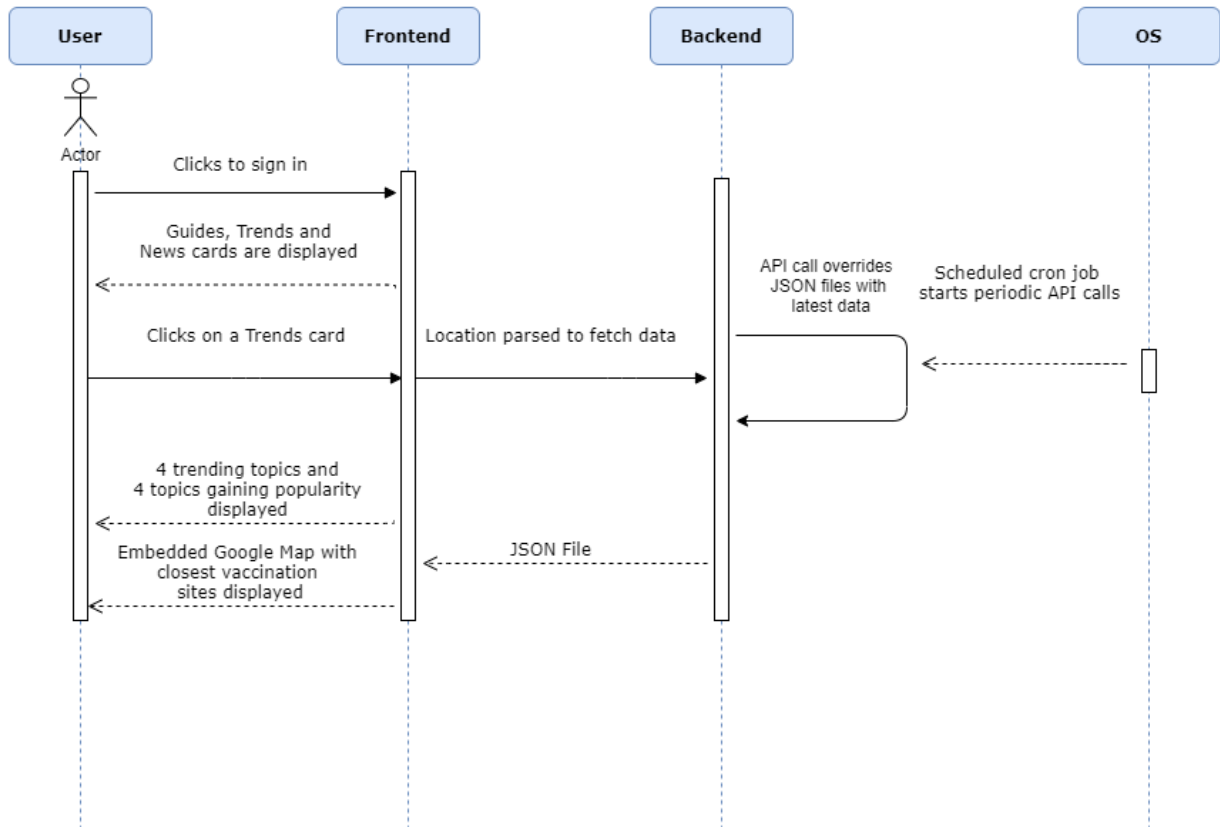
2. Use Case Diagrams



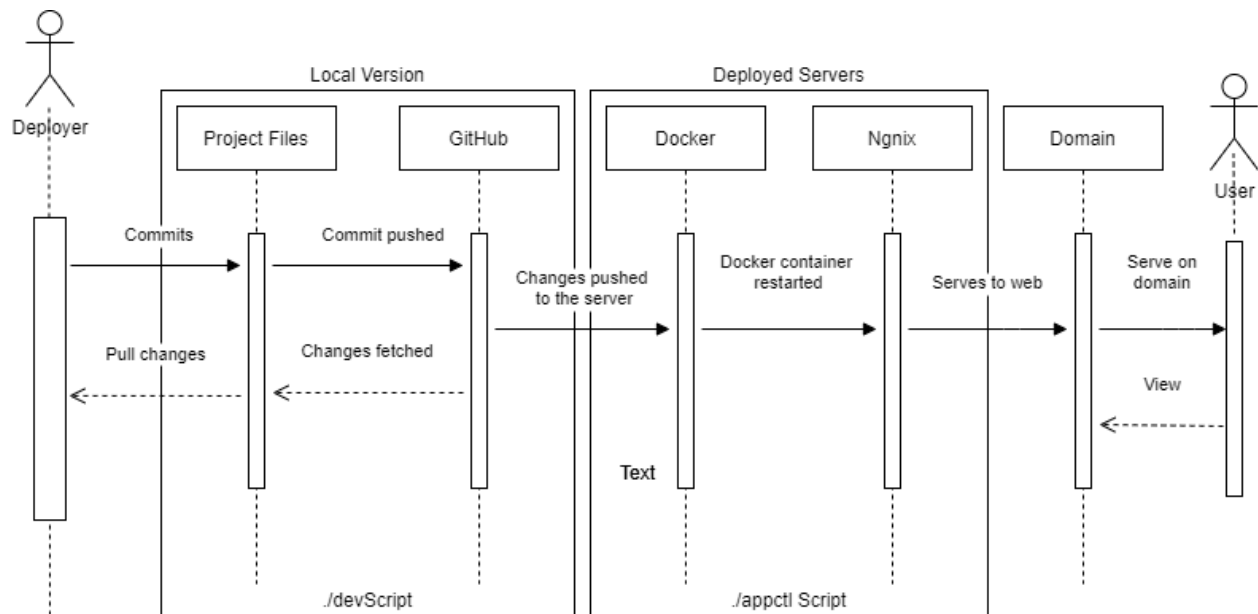


3. Sequence Diagram

Trends page sequence



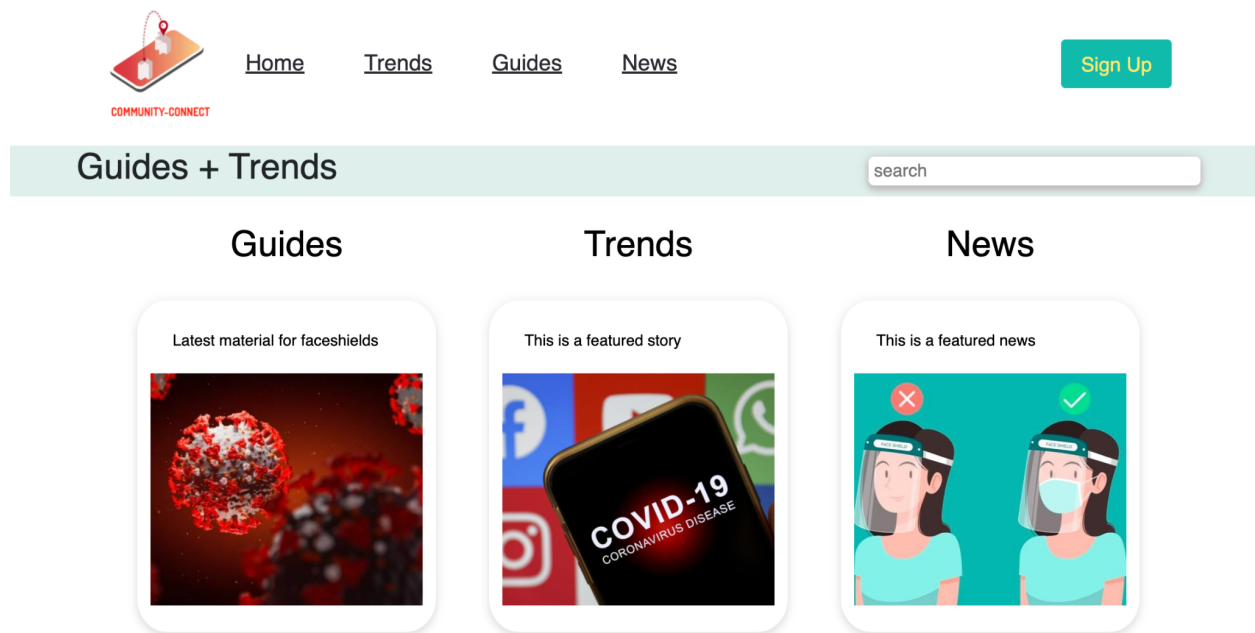
4. Development cycle sequence diagram

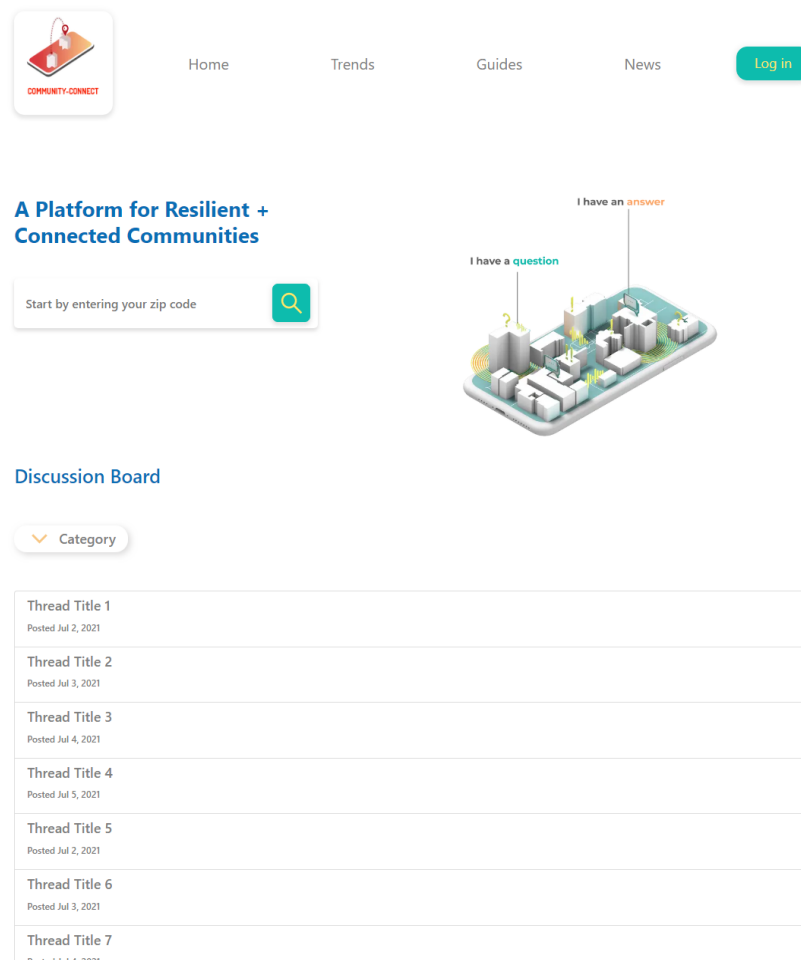


Appendix B - User Interface Design

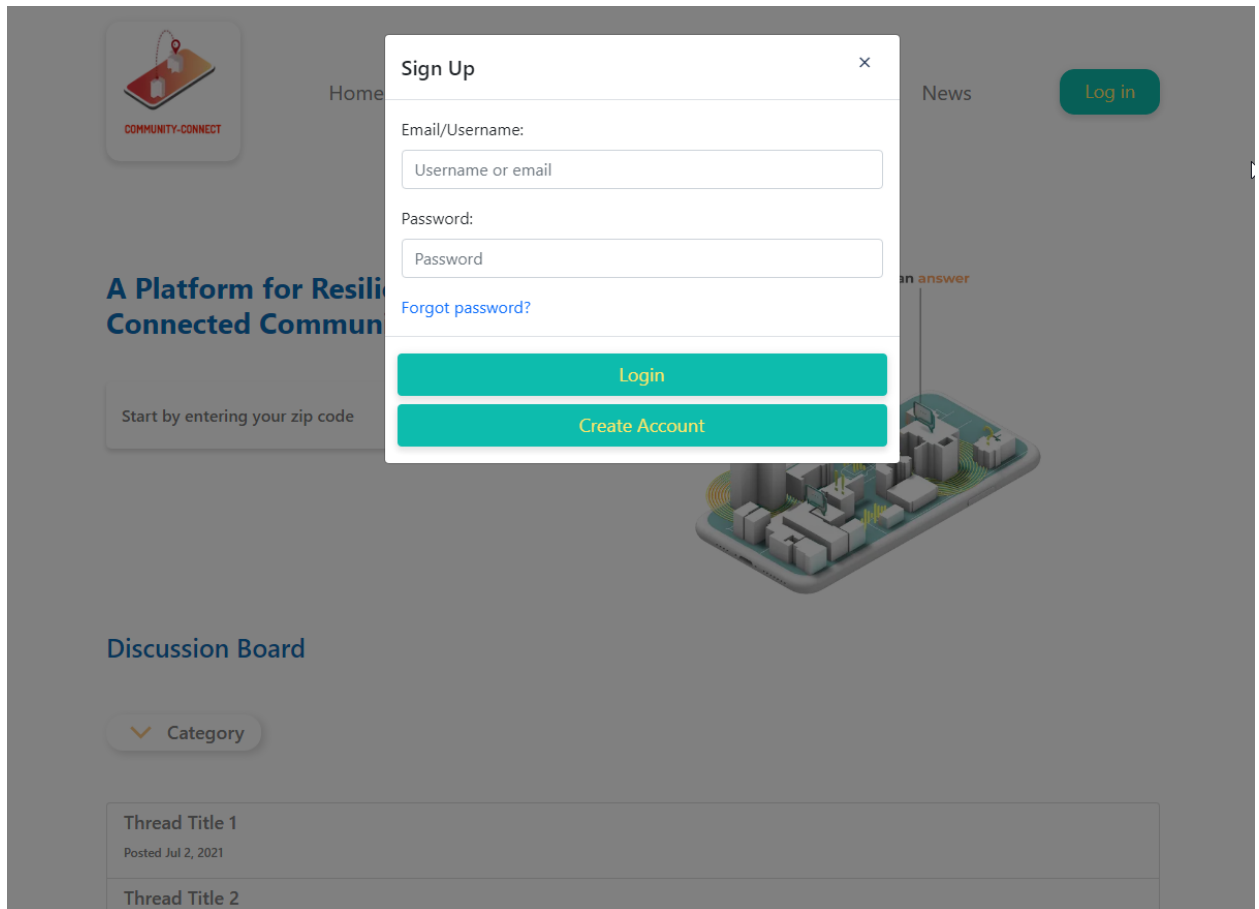
1. Home Page

Old design:



New design:

2. Sign Up Screen



The image shows a 'Sign Up' modal form overlaid on a website background. The modal is a white rectangle with a close button (X) in the top right corner. It contains the following elements:

- Sign Up** (Title)
- Email/Username:** A text input field with the placeholder text 'Username or email'.
- Password:** A text input field with the placeholder text 'Password'.
- [Forgot password?](#) (Link)
- Login** (Button)
- Create Account** (Button)

The background website is a greyish-blue color. It features a logo in the top left corner that says 'COMMUNITY-CONNECT' with an icon of a smartphone and a location pin. The main heading is 'A Platform for Resilient Connected Communities'. Below this is a text input field for 'Start by entering your zip code'. To the right, there is a 'Log in' button. At the bottom, there is a 'Discussion Board' section with a 'Category' dropdown menu and two thread titles: 'Thread Title 1' (Posted Jul 2, 2021) and 'Thread Title 2'.

3. Create Account Screen

COMMUNITY-CONNECT

Home News Log in

A Platform for Resilient Connected Communities

Start by entering your zip code

Create Account [X]

Username:

E-mail:

Zipcode:

Password:

Register Cancel

Discussion Board

Category

Thread Title 1
Posted Jul 2, 2021

Thread Title 2
Posted Jul 3, 2021

4. Guides Page

Old design:

Wear a mask

Description

[Read more](#)



Wash your hands often

Description

[Read more](#)



Stay 6 feet away

Description

[Read more](#)



Edit

Delete

Edit

Delete

Edit

Delete

Clean and disinfect

Description

[Read more](#)



Avoid crowds

Description

[Read more](#)



Get a healthy Sun

Description

[Read more](#)



Edit

Delete

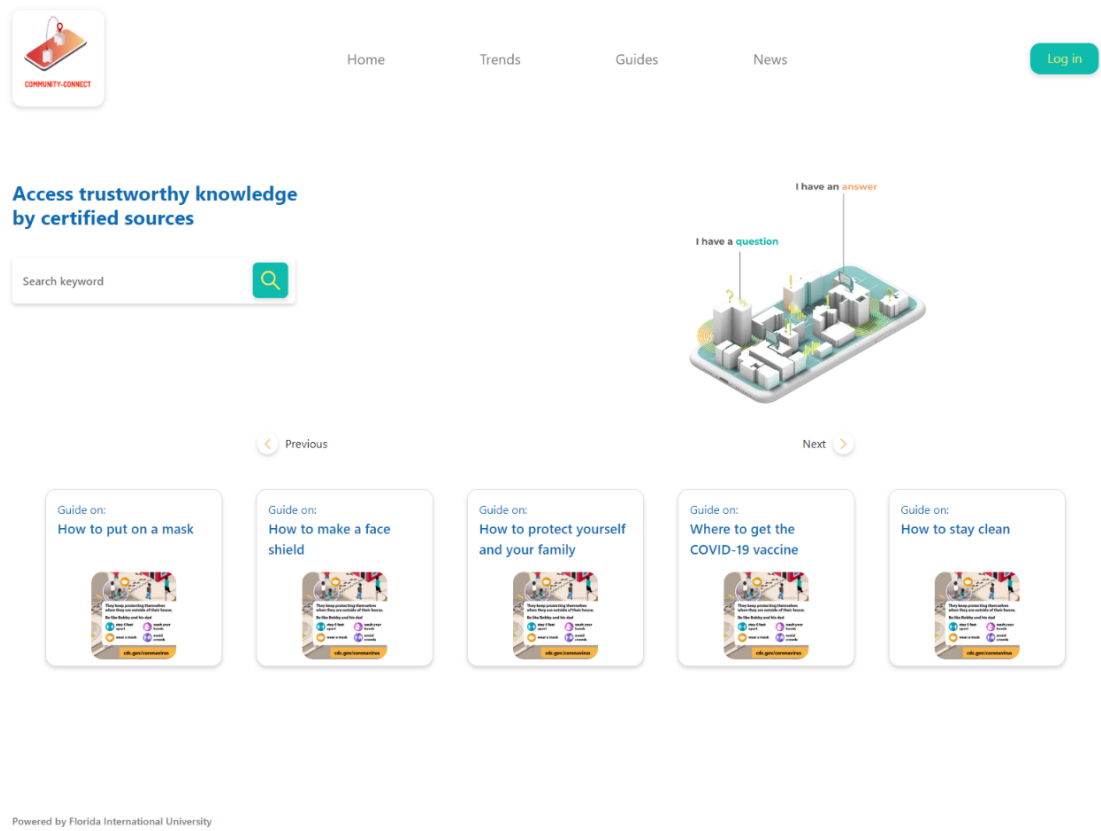
Edit

Delete

Edit

Delete

New design:



5. News Page

Old design:

news articles about "Coronavirus"

by The Washington Post on MSN.com



[In countries where vaccines are available, offers of free airline tickets and apartments for the fully inoculated](#)

In an effort to get ahead of vaccine hesitancy, some countries and companies are getting creative with material incentives.



by Houston Chronicle

[The Latest: UK official says variant may delay lockdown lift](#)

Britain's health secretary says the delta variant, which is fast becoming the dominant coronavirus variant in the U.K., is 40% more transmissible compared to the country's existing strains. Matt Hancock acknowledged Sunday that the rise in Optimism that Britain can vaccinate its way out of the pandemic took a dent in recent weeks with growing concern that infections are again rising,

by The Washington Post on MSN.com



[In countries where vaccines are available, offers of free airline tickets and apartments for the fully inoculated](#)

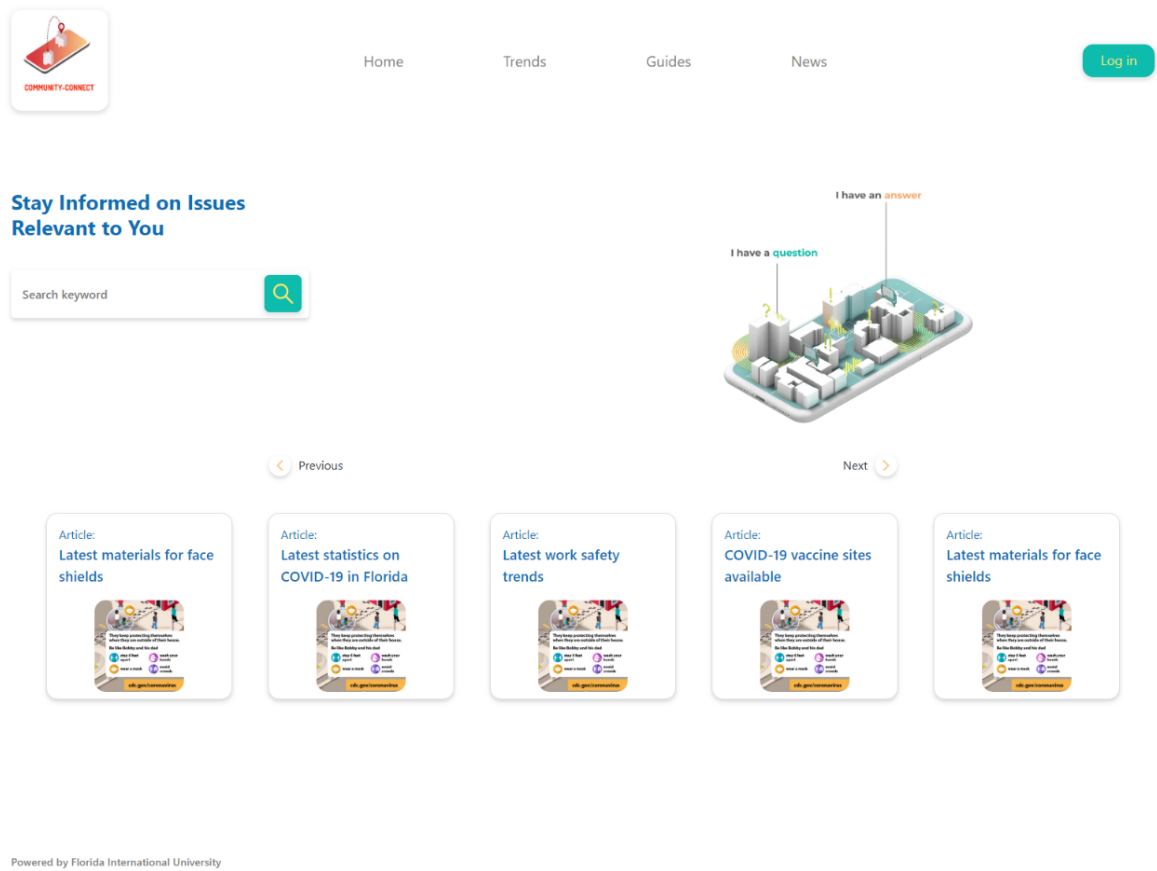
In an effort to get ahead of vaccine hesitancy, some countries and companies are getting creative with material incentives.



by Houston Chronicle

[The Latest: UK official says variant may delay lockdown lift](#)

Britain's health secretary says the delta variant, which is fast becoming the dominant coronavirus variant in the U.K., is 40% more transmissible compared to the country's existing strains. Matt Hancock acknowledged Sunday that the rise in Optimism that Britain can vaccinate its way out of the pandemic took a dent in recent weeks with growing concern that infections are again rising,

New design:

6. Trends Page

Old design:

Covid Vaccination sites near you



4 Trending topics in Florida

Coronavirus disease 2019

Vaccine

COVID-19 vaccine

COVID-19 testing

4 Topics gaining popularity in Florida

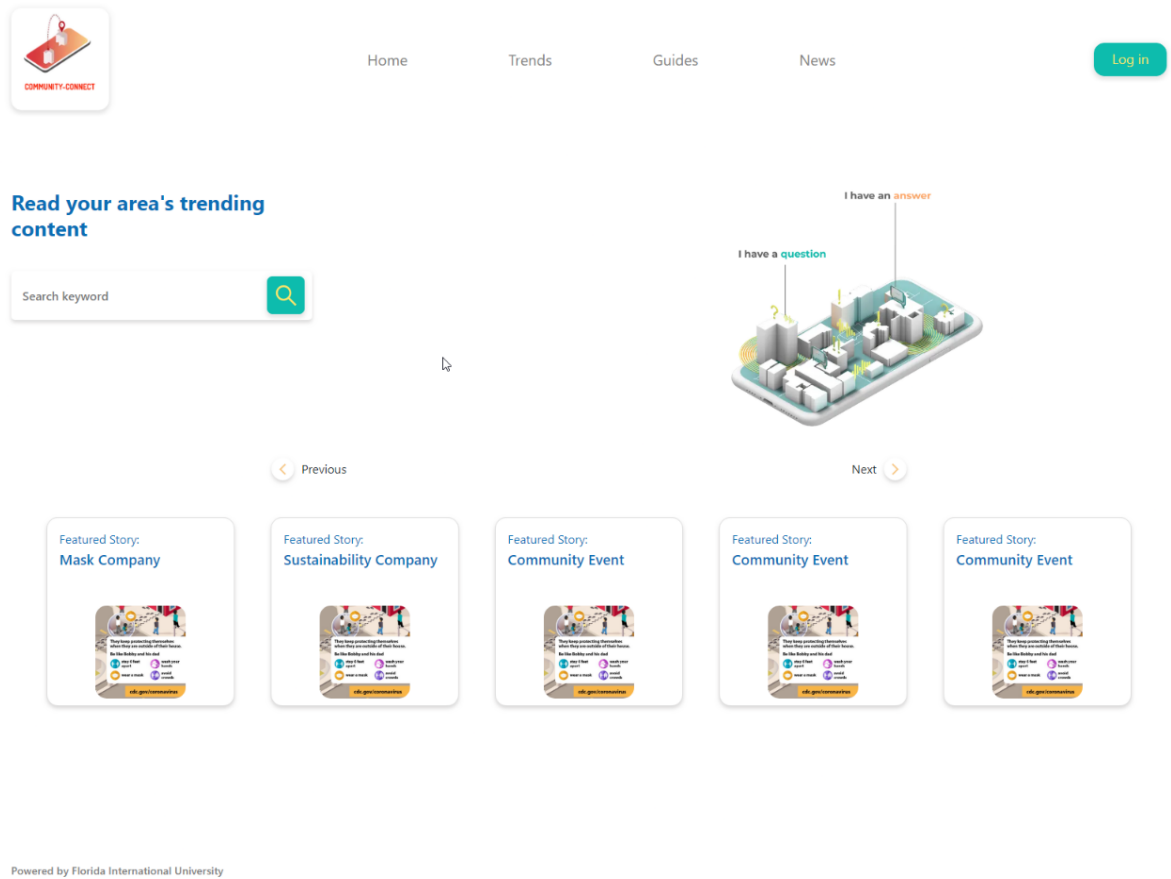
Walgreens Healthcare Clinic

Healthcare Clinic

COVID-19 vaccine

COVID-19 testing

New design:



7. Carbon Footprint Calculator

[Home](#)[Trends](#)[Guides](#)[News](#)[Log in](#)

[Get Started](#)[Travel](#)[Home](#)[Food](#)[Shopping](#)

Start with a quick estimate

Where do you live?

How many people live in your household?

1 2 3 4 5+

What is your gross annual household income?

<10k 10k 20k 30k 40k 50k 60k 80k 100k 120k+

Graph placeholder

Powered by Florida International University

Appendix C - Sprint Review Reports

Sprint Review - Sprint 1

Attendees:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

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

- **As a developer I would like to redo the News page based on the new design**
- **As a developer I would like to redo the Guides page based on the new design**
- **As a developer I would like to redo the Trends page based on the new design**

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.

- Implement search bar functionality.
 - How this should be reflected on the user story definition in Mingle:
 - This user story will be moved to pending
- As a developer I would like the admin to update the images on the cards (conditionally render a edit when the user is an admin), when edit is clicked a modal opens to edit the the card

Attached below is the **copy** of the team's **Sprint Planning Meeting Minutes**:

Sprint Planning Meeting Minutes

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 5:00 PM

End time: 6:00 PM

After discussion, the velocity of the team was estimated to be 7 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 complete the project installation guide. (1 hour)
- User Story 2 - As a developer I would like to familiarize with the project structure (1 hour)
- User Story 3 - As a developer I would like to familiarize with React (1 hour)
- User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows (2 hours)
- User Story 5 - As a developer I would like to implement a better User Interface for the Home Page (2 hours)

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

- Niurbelis Gonzalez
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- Sebastian Gomez
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- Shaoting Wu

- User Story 1 - As a developer I would like to complete the project installation guide.
- User Story 2 - As a developer I would like to familiarize with the project structure.
- User Story 3 - As a developer I would like to familiarize with React.
- Adrian Gonzalez
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- Chábeli Castano Arango
 - User Story 1 - As a developer I would like to complete the project installation guide.
 - User Story 2 - As a developer I would like to familiarize with the project structure.
 - User Story 3 - As a developer I would like to familiarize with React.
- Narmeen Kibria
 - User Story 2 - As a developer I would like to familiarize with the project structure (1 hour)
 - User Story 3 - As a developer I would like to familiarize with React (1 hour)
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page (2 hours)
- Manuel Kamboykos
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- David Garcia
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 1 - As a developer I would like to complete the project installation guide. (1 hour)
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows (2 hours)

Sprint Review - Sprint 2

Attendees:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

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

- **As a developer I would like to implement search bar functionality.**
- **As a developer I would like the admin to update the images on the cards (conditionally render a edit when the user is an admin), when edit is clicked a modal opens to edit the the card**
- **As a developer I would like to add stories to make remaining pages for the nav links**
- **As an administrator I would like to collect analytics with Google Analytics about the most popular / trending statement**
- **As a user I would like to have an analytics page with useful covid-19 data such as cases, vaccination rates and country comparisons.**
- **As a user I want to be able to see what the most trending COVID 19 topics are based on my location(COVID, PPE, FDA approved, faceshield, mask)**

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.

- As a developer I would like to implement an ai bot service. This can be through selenium or through the backend..
 - How this should be reflected on the user story definition in Mingle:
 - This user story will be moved to pending
- As a developer, I would like to have a database of the Zip codes that will map to appropriate states.
- As a user I want to have a bot answer my questions based on keywords and syntax. Can you make a sustainable faceshield? Can I find a 3D printing service in my neighborhood to make face shields? What are the benefits of wearing a face shield?

Attached below is the **copy** of the team's **Sprint Planning Meeting Minutes**:

Sprint Planning Meeting Minutes:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 15 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 redo the Guides page based on the new design (5 hours)
- User Story 2 - As a developer I would like to redo the Trends page based on the new design (5 hours)
- User Story 3 - As a developer I would like to redo the News page based on the new design (5 hours)

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

- Niurbelis Gonzalez
 - User Story 3 - As a developer I would like to redo the News page based on the new design
 - User Story 1 - As a developer I would like to redo the Guides page based on the new design

- User Story 2 - As a developer I would like to redo the Trends page based on the new design
- Sebastian Gomez
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- Shaoting Wu
 - User Story 3 - As a developer I would like to redo the News page based on the new design (5 hours)
- Adrian Gonzalez
 - User Story 1 - As a developer I would like to redo the Guides page based on the new design
 - User Story 3 - As a developer I would like to redo the News page based on the new design
 - User Story 2 - As a developer I would like to redo the Trends page based on the new design
- Chábeli Castano Arango
 - User Story 3 - As a developer I would like to redo the News page based on the new design
 - User Story 1 - As a developer I would like to redo the Guides page based on the new design
 - User Story 2 - As a developer I would like to redo the Trends page based on the new design

- Narmeen Kibria
 - User Story 1 - As a developer I would like to redo the Guides page based on the new design
- Manuel Kamboykos
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- David Garcia
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 1 - As a developer I would like to complete the project installation guide. (1 hour)
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows (2 hours)

Sprint Review - Sprint 3

Attendees:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

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

- **As a developer I would like to create a REST API for the Thread class as described in the documentation.**
- **As a developer I would like to create a REST API for the User class as described in the documentation.**
- **As a developer I would like to create a REST API for the Polls class as described in the documentation.**
- **As a developer I would like to create a REST API for the Category class as described in the documentation.**
- **As a developer I would like to create a REST API for the Post class as described in the documentation.**

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.

- As an administrator I would like to collect analytics with Google Analytics about the most popular / trending statement
 - How this should be reflected on the user story definition in Mingle:
 - This user story will be moved to pending
- As a user I would like to have an analytics page with useful covid-19 data such as cases, vaccination rates and country comparisons.
- As a developer I would like to implement a drop down for the guides, news and trends.

Attached below is the **copy** of the team's **Sprint Planning Meeting Minutes**:

Sprint Planning Meeting Minutes:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 35 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 create a REST API for the Thread class as described in the documentation. (7 hours)
- User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation. (7 hours)
- User Story 3 - As a developer I would like to create a REST API for the Polls class as described in the documentation. (7 hours)
- User Story 4 - As a developer I would like to create a REST API for the Category class as described in the documentation. (7 hours)
- User Story 5 - As a developer I would like to create a REST API for the Post class as described in the documentation. (7 hours)

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

- Niurbelis Gonzalez

- User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation.
- User Story 1 - As a developer I would like to create a REST API for the Thread class as described in the documentation.
- User Story 4 - As a developer I would like to create a REST API for the Category class as described in the documentation.
- User Story 3 - As a developer I would like to create a REST API for the Polls class as described in the documentation.
- User Story 5 - As a developer I would like to create a REST API for the Post class as described in the documentation.
- Sebastian Gomez
 - User Story 5 - As a developer I would like to create a REST API for the Post class as described in the documentation.
- Shaoting Wu
 - User Story 5 - As a developer I would like to create a REST API for the Post class as described in the documentation. (7 hours)
- Adrian Gonzalez
 - User Story 1 - As a developer I would like to create a REST API for the Thread class as described in the documentation.
 - User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation.
 - User Story 3 - As a developer I would like to create a REST API for the Polls class as described in the documentation.

- Chábeli Castano Arango
 - User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation.
 - User Story 1 - As a developer I would like to create a REST API for the Thread class as described in the documentation.
- Narmeen Kibria
 - User Story 2 - As a developer I would like to create a REST API for the User class as described in the documentation. (7 hours)
- Manuel Kamboykos
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows
 - User Story 5 - As a developer I would like to implement a better User Interface for the Home Page
- David Garcia
 - User Story 2 - As a developer I would like to familiarize with the project structure
 - User Story 3 - As a developer I would like to familiarize with React
 - User Story 1 - As a developer I would like to complete the project installation guide. (1 hour)
 - User Story 4 - As a developer I would like to research a work around solution to run Docker in Windows (2 hours)

Sprint Review - Sprint 4

Attendees:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

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

- **As a developer I would like to create a frontend service that can consume the Django REST API for Threads.**
- **As a developer I would like to create a frontend service that can consume the Django REST API for Posts.**
- **As a developer I would like to create a frontend service that can consume the Django REST API for User.**

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.

- As a developer I would like to implement search bar functionality.
- As a developer I would like the admin to update the images on the cards (conditionally render a edit when the user is an admin), when edit is clicked a modal opens to edit the the card
- As an administrator I would like to collect analytics with Google Analytics about the most popular / trending statement

Attached below is the **copy** of the team's **Sprint Planning Meeting Minutes**:

Sprint Planning Meeting Minutes:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 24 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 create a frontend service that can consume the Django REST API for Threads. (8 hours)
- User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- User Story 3 - As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)

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

- Niurbelis Gonzalez
 - User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for User.
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts.
 - User Story 1 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads.

- Sebastian Gomez
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Shaoting Wu
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Adrian Gonzalez
 - User Story 1 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
 - User Story 3 - As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Chábeli Castano Arango
 - User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for User (8 hours)
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Narmeen Kibria
 - User Story 1 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
- Manuel Kamboykos
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- David Garcia
 - User Story 2 -As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)

Sprint Review - Sprint 5

Attendees:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

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

- **As a developer I would like to create a frontend service that can consume the Django REST API for Threads.**
- **As a developer I would like to create a frontend service that can consume the Django REST API for Posts.**
- **As a developer I would like to create a frontend service that can consume the Django REST API for User.**

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.

- As a developer I would like to add all the relationships in the REST API.
- As a developer I would like the admin to update the images on the cards (conditionally render a edit when the user is an admin), when edit is clicked a modal opens to edit the the card
- As a developer I would like to implement search bar functionality

Attached below is the **copy** of the team's **Sprint Planning Meeting Minutes**:

Sprint Planning Meeting Minutes:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 24 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 redo the Home page based on this design
- User Story 2 - As a User I would like forgot password in the Login Page.
- User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
- User Story 4 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- User Story 5 - As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)

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

- Niurbelis Gonzalez
 - User Story 1 - As a developer I would like to redo the Home page based on this design
 - User Story 5 - As a developer I would like to create a frontend service that can consume the Django REST API for User.

- User Story 4 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts.
- User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads.
- User Story 2 - As a User I would like forgot password in the Login Page.
- Sebastian Gomez
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- Shaoting Wu
 - User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
- Adrian Gonzalez
 - User Story 1 - As a developer I would like to redo the Home page based on this design
 - User Story 3 - As a developer I would like to create a frontend service that can consume the Django REST API for Threads. (8 hours)
 - User Story 4 - As a developer I would like to create a create a frontend service that can consume the Django REST API for User. (8 hours)
 - User Story 5 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
 - User Story 2 - As a User I would like forgot password in the Login Page.
- Chábeli Castano Arango
 - User Story 1 - As a developer I would like to redo the Home page based on this design
 - User Story 5 - As a developer I would like to create a frontend service that can consume the Django REST API for User.
- Narmeen Kibria
 - User Story 1 - As a developer I would like to redo the Home page based on this design

- Manuel Kamboykos
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- David Garcia
 - User Story 2 -As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)

Sprint Review - Sprint 6

Attendees:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

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

- As a developer I would like to create a REST API for the Polls class as described in the documentation.
- As a developer I would like to create a REST API for the Category class as described in the documentation.
- As a user I would like to have a page for the Carbon Footprint Calculator.

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.

- As a developer I would like to create a frontend service that can consume the Django REST API for Posts.
- As a developer I would like the admin to update the images on the cards (conditionally render a edit when the user is an admin), when edit is clicked a modal opens to edit the the card
- As a developer I would like to implement search bar functionality
- As a developer I would like to add stories to make remaining pages for the nav links

Attached below is the **copy** of the team's **Sprint Planning Meeting Minutes**:

Sprint Planning Meeting Minutes:

Capstone II: Niurbelis Gonzalez, David Garcia, Sebastian Gomez,.

Capstone I: Manuel Alejandro Kamboykos, Chabeli Castano Arango, Narmeen Kibria, Shaoting Wu.

Senior Project: Adrian Gonzalez

Start time: 3:30 PM

End time: 4:30 PM

After discussion, the velocity of the team was estimated to be 24 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 create a REST API for the Polls class as described in the documentation. (8 hours)
- User Story 2 - As a developer I would like to create a REST API for the Category class as described in the documentation. (8 hours)
- User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages) (8 hours)

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

- Niurbelis Gonzalez
 - User Story 2 - As a developer I would like to create a REST API for the Category class as described in the documentation.
 - User Story 1 - As a developer I would like to create a REST API for the Polls class as described in the documentation.
 - User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages)

- Sebastian Gomez
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
 - User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages)
- Shaoting Wu
 - User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages) (8 hours)
- Adrian Gonzalez
 - User Story 3 - As a user I would like to have a page for the Carbon Footprint Calculator. (Model it off the other pages)
 - User Story 2 - As a developer I would like to create a REST API for the Category class as described in the documentation.
 - User Story 1 - As a developer I would like to create a REST API for the Polls class as described in the documentation.
- Chábeli Castano Arango
 - User Story 2 - As a developer I would like to create a REST API for the Category class as described in the documentation.
 - User Story 1 - As a developer I would like to create a REST API for the Polls class as described in the documentation.
- Narmeen Kibria
 - User Story 1 - As a developer I would like to create a REST API for the Polls class as described in the documentation. (8 hours)
- Manuel Kamboykos
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)
- David Garcia
 - User Story 2 - As a developer I would like to create a frontend service that can consume the Django REST API for Posts. (8 hours)

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

Local Development Environment Setup Guide

The preferred way to run development instances of COVID Innovation Network on your local machine is with [Docker](#), which makes it easy to spin up an arbitrary number of instances running different code with separate databases and dependencies besides each other.

For New Prototype:

Frontend:

Downloads list

- Chrome or Firefox (do not use Edge or Safari)
- React Dev Tools Chrome extension
- iTerm2 (if using Mac)
 - <https://iterm2.com>
 - Make sure to move the application to the applications folder
- Xcode Command Line Tools
 - This is installed with iTerm2
 - Verify install using `$ xcode-select -p`
- Homebrew (if using Mac)
 - <https://brew.sh>
 - Check install using `$ brew doctor`
- ZSH (if using Mac)
 - This should already be installed with the latest version of MacOS
 - Verify using `$ zsh --version`
- NVM (if using Mac)
 - Run `$ mkdir ~/.nvm`
 - Run `$ brew install nvm`
 - Update (create if the file doesn't exist) `~/.zshrc` with the commands found in <https://formulae.brew.sh/formula/nvm>
 - Close and reopen terminal
 - Verify install `$ nvm --version`

- Node
 - Run `$ nvm install node`
 - Check install `$ node -v`
 - Check NPM install `$ npm -v`
- Docker
 - Follow instructions <https://docs.docker.com/get-docker/>

Backend:

Once Docker is installed and the repository cloned:

Make sure that Docker is running. Open terminal in VS code and make sure you are in the root directory of the project, the same directory where the README.md is. Run the following commands to build and start the Docker containers:

```
docker-compose build
```

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

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

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

```
docker-compose up
```

From that point use `docker-compose up` to start the backend server and `docker-compose down -v` to stop the server.

For Older Version of Community Connect:

Once Docker is installed, clone the repository and run `./dev init` command in your terminal. This will build necessary docker containers, install python dependencies and initialize the database. After the command does its magic, you will be able to start the development server using the `docker-compose up` command.

The `./dev` utility implements other features besides the `init`. Run it without any arguments to get the list of available actions.

Running the Local Development Server

To start the development server at any time, regardless of OS enter `docker-compose up` into your terminal (for Windows use PowerShell to run this command).

After the development server starts, visit <http://localhost:8000> in your browser to see your Misago installation.

The Admin Control Panel is available at <http://localhost:8000/admincp/> in your browser. To log in to it use Admin username: 'admin' and password: 'password'.

Deployed Development Server

SSH Login Information for Deployed Development Server

Hostname: misago.live

Port: 22

Username: `root`

Password: `%^56TYtyGHghBNbn`

File Protocol: `SFTP`

Deployed Development Server Usage

This server is hosted by [Digital Ocean](#), it is a dual-core droplet with 2GB of ram and 50GB of storage (keep this in mind).

Because the server only has 2GB of memory, if too many days (less than a week) pass by without a proper shutdown, the server will break. Please use these steps to properly start and shut down the server:

Starting the server:

1. SSH into the `misago_docker` folder
2. In the corresponding terminal run the command: `./appctl start` or `docker-compose-up`

Shutting down the server:

1. SSH into the `misago_docker` folder

2. In the corresponding terminal run the command: `./appctl stop` or `docker-compose-down`
-

Deployed Production Server

SSH Login Information for Deployed Production Server

Hostname: community-connect.cs.fiu.edu

Port: 22

Username: Your FIU Username (If you are new to the project you must email request@cis.fiu.edu and cc the PO, and your professor to gain access to the server, make sure to ask for SUDO permissions)

Password: Your FIU CS lab password

File Protocol: SFTP

Deployed Production Server Note (IMPORTANT)

This server is critical to the success of the project and as such should always be running, always have a stable version of Community Connect running on it, and always have tight security around it.

This server has safeguards in place to ensure changes are meant, so make sure every change made here has been tested and approved on both local dev versions and the [dev server](#).

For `./appctl` changes to work `sudo` must be used. This is also the case for `docker-compose` commands. Also, to ensure server stability when the server has to be started it should be done using `sudo docker-compose up -d` NOT `./appctl` and its start-up functions.

`./appctl` start-up functions are disabled on this server for stability and security reasons. ex. to restart the server: `sudo ./appctl restart` will shut down the server; then to start up again `sudo docker-compose up -d` must be run.

The `-d` argument in the `docker-compose` command is to run it in daemon mode.

Production Email Configuration

- Production has an email address configured: `community-connect@cs.fiu.edu`
- This address is an alias so it forwards to the PO's email and we can send emails out through SMTP forwarding with `smtp.cs.fiu.edu`
- If changes have to be made to the email used `sudo ./appctl email` can facilitate those changes.

For `@cs.fiu.edu` emails to send emails correctly these settings must be configured:

- Host: `smtp.cs.fiu.edu`
 - User: `community-connect`
 - Port: 25
 - SSL: NO
 - TLS: NO
-

Frontend Changes (CSS/JS)

Install Gulp.js and any other dependencies it requires.

<https://gulpjs.com/docs/en/getting-started/quick-start>

Once you edit any of the `.less` or `.js` files. You need to build the changes using the terminal with the command `gulp build`. NOTE: other NPM dependencies may need to be installed, so simply search online for the missing ones according to the terminal. Your built files will be stored in the `misago/static/misago` directory.

To make these changes live in the server, you will need to SSH into the server or use a program like WinSCP and copy the `misago/static/misago` directory into the `misago-docker misago_docker/misago/theme/static/` directory. When the pop-up window appears for permissions, click on the drop-down icon next to 'Yes' and select "Yes To All"

CMD

- `ssh root@misago.live`
- Same Password

- `cd misago_docker`

After deploying your new file, you must `cd` into the `misago_docker` folder and in the terminal run the command `./appctl collect static` to make Misago "collect" this file and replace the default with it.

Then return to the root folder and run the command `docker-compose build` to build changes.

- `'./appctl restart'`

Frontend Changes (HTML Templates)

You must create a duplicate version of the `.html` template and move it over to the `misago-docker theme/templates/misago/jumbotron.html` and customize them there.

Then run `./appctl restart` to make the changes live.

Authentication Issues

There may be times when a user attempts to sign in to their account, where they are not redirected to a new page with their account logged in. In this case, the Redis cache needs to be cleared. This can be achieved by following these steps:

1. Open the Docker Desktop Application
2. Make sure that an app is currently running and click on the running app to view the containers.
3. Hover over `rootfoldername_redis_1` container.
4. Click on the first blue circle button that reads CLI when hovered over.
5. In the Redis terminal window run the command `redis-cli FLUSHDB`, the terminal should read 'OK' after successfully running the command.
6. In the redis terminal window run the command `redis-cli -n DB_NUMBER FLUSHDB`, the terminal should read 'OK' after successfully running the command.
7. In the redis terminal window run the command `redis-cli -n DB_NUMBER FLUSHDB ASYNC`, the terminal should read 'OK' after successfully running the command.
8. In the redis terminal window run the command `redis-cli FLUSHALL`, the terminal should read 'OK' after successfully running the command.

9. In the redis terminal window run the command `redis-cli FLUSHALL ASYNC`, the terminal should read 'OK' after successfully running the command.
10. Return the root folder of the project and run `docker-compose-build` and now the user should be able to authenticate.

Shortcomings/Wishlist Document

Among the possible improvements that should be made in the next iteration of this projects are:

1. Implement a graph with Chart.js for the Carbon Footprint Calculator page.
2. Connect the Carbon Footprint Calculator to the Carbon Interface API.
3. Finish implementing the API's for both frontend and backend.
4. Fixing mobile view for Guides/Trends/News pages.
5. Chat-bot/Search integration and connection with elastic search.
6. Admin/management application to manage Guides page.
7. Location prompt on a page-load to inform users as well as allow them to choose their location/privacy preferences.

References

<https://docs.docker.com>

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

<https://reactjs.org>

<https://nextjs.org/>