Capstone Proposal

Project Idea: ChatGPT 4 Website Builder

Project Name: Webbie

By: Alex Bates

**Topic**

I aim to create a website creation tool that leverages ChatGPT4 to speed up and lower the bar for the web development process. The idea is that you can select your framework, explain the kind of website you want, then ChatGPT will generate a lot of the code and my application will visualize the code live while also allowing you to make edits and continue development.

**Core Requirements**

* User Account System
  + Create/Delete Endpoints
  + Multiple Get Endpoints
  + Multiple Update Endpoints
* User can create projects, a project is a website
  + Boilerplate is non-ai, but the ai should fill out some initial pages
* User can instruct ChatGPT to create a react page and then edit it with a live preview
* User can instruct ChatGPT to create a react component and then edit it with a live preview
* User can preview generated code in my application
  + This entails:
    - Saving user and AI code to a gitlab server
    - Running a docker web container that pulls the gitlab files
    - Connecting to that container with an iframe on the frontend
* Run ChatGPT (and user?) code through linting process
* User can experience these things on a decent looking frontend
* Code editor has syntax highlighting and is nice to use
* User can delete pages, or projects
* User can do code editor things such as
  + Create Files
  + Rename Files
  + Delete Files
  + Save Files
  + Preview pages and components
  + CRUD Folders
  + Move Files and Folders
* An Editor API to handle all those editor things and forward

**Stretch Requirements**

* Advanced Options menu for code generation tweaks
* More frameworks
* Console access in preview
* Profile pictures
* Debug selected code
* Explain selected code
* Generate code here (cursor position on right click)
* Switch code between typescript and javascript
* Auto format code
* Make frontend have as much charm as old Nintendo UI
  + Make a mascot!!
    - Froggy!!
    - Rubber Duck!!
    - Spider!!
  + Make different themes for the site!

**Tech Stack**

* Python + assorted libraries
* Javascript + assorted libraries
* Fastapi
* React
* ChatGPT 4 API
* Docker Web container stuff
* Material UI
* Gitlab
* Kafka
* Kong or Ocelot Gateway
* Mongodb

**Proposed Schedule**

You have 10 weeks and are required to spend a minimum of 20 hours/week on your project. Map your core requirements onto a 10-week schedule showing which requirements you plan to complete each week.

Week 0 – Build a bunch of diagrams in preparation

Not actually week zero, but all the time leading up to week 1 I want to be planning the UI and the ways I will be interacting with ChatGPT. I want to have my entire predicted Trello to do list made and all of my page designs prepared before week 1 starts.

Week 1 – ChatGPT request foundations

This week I want to figure out a way to consistently get bootstrap code and more from ChatGPT while automatically previewing it. I want to send generated code through a linter, and possibly another instance of ChatGPT to verify that it doesn’t have any errors preventing it from running. A technique I plan to use to enhance consistency is to start every “session” with ChatGPT outlining the pseudocode format that it should respond with.

Week 2 – Frontend foundations

Week 2 I want to switch gears and work on the frontend, this will be built in react and will likely utilize the material UI component library to make creating the base of the frontend quite quick. Priority is getting an editor setup with ChatGPT generating code, then getting the live preview setup

Week 3 – Web Containers

This week will be dedicated to getting web containers running and updating based on user and possibly ChatGPT input. I predict this to be a challenging task, if that prediction is inaccurate then I will spend the rest of the week working on polishing the ways I ask ChatGPT for results.

Week 4 – ChatGPT communication polish

One of the crazy things about ChatGPT is how important your prompting is, we call the career “Prompt Engineering”, I plan to spend this week engineering the prompts used to generate websites, pages, and components.

Week 5 – Dockerize and setup any required architecture stuff/Touch Base

This week I want to make sure everything is dockerized and that any secondary servers I may need like a gateway, Kafka, or signalR are set up. The goal is to use Kong gateway so I can host from home. While this is something that I should be doing throughout the project, I would also like to formally set that at week 5 I want to compare my progress with my plans and make modifications to my schedule accordingly.

Week 6 – Develop frontend, Polish Frontend, and get user feedback from people I know

First priority is getting core features out so I can start testing and getting feedback

Second priority is polishing the site and making it look nice

This includes a homepage and some of the fluff stuff

Third priority is addressing feedback from “playtests”

Week 7 – Polish the backend, make sure security is up to snuff, and more user feedback

This week is the week to formally check back on the backend api to bug test and make sure I’m doing security right. I will also spend this week addressing feedback and trying to get more of it. Contents of this week may get pushed back to week 8 for frontend development time.

Week 8 – Polish Polish Polish, a pretty frontend is crucial for a pretty presentation

Either stretch items start, or crunch time starts, either way this application better be shaping up by this point. Before I start working on stretch items I would like to create my presentation materials. A diagram of my architecture, a slide for web containers, a slide or 2 for the ChatGPT prompt engineering, some other stuff idk.

Week 9 – Crunch

Whether it’s working on stretch goals or finishing the 1.0 version, I will be hard at work and making sure that I always have a fully functional and closely up to date presentation branch on github that never has broken code.

Week 10 – Rest/Prepare for presentation

This week development should be over, last-minute changes are how you mess up a presentation. This week is only for prepping that presentation and making sure I have a backup recording.

**Diagrams**

**Service Architecture Diagram 1**

**A computer screen shot of a diagram

Description automatically generated**

**Service Architecture Diagram 2** (Displays frameworks when possible)

A diagram of a diagram of a person

Description automatically generated

**Schema Diagram**

**A diagram of a computer

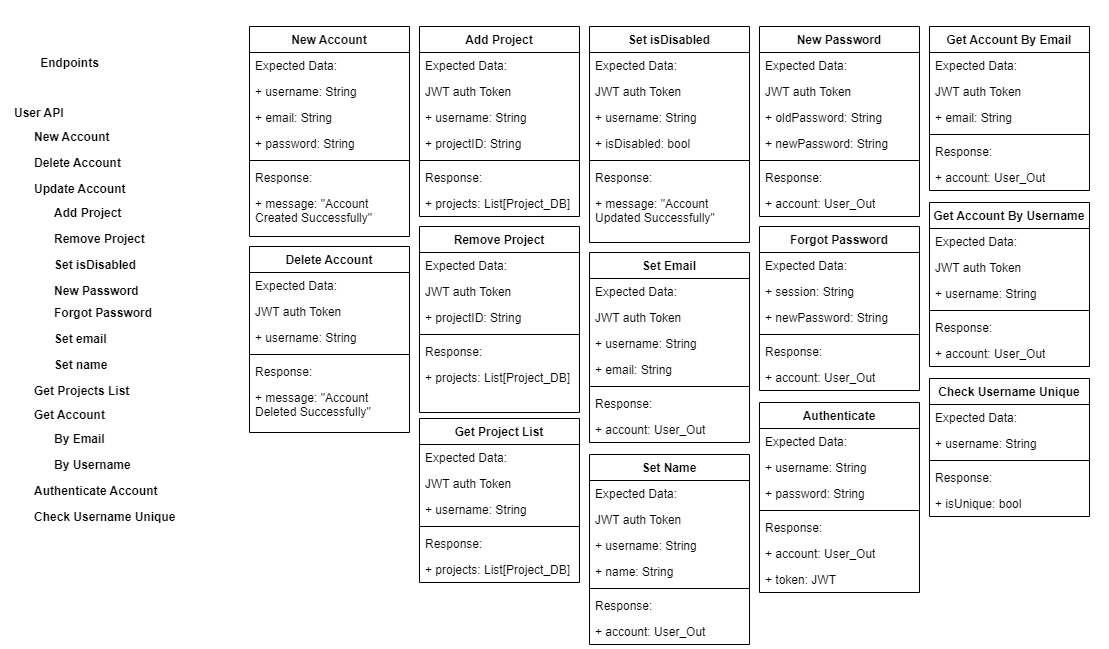
Description automatically generated**

**Editor API Group Endpoint Diagram**

**A group of black and white text

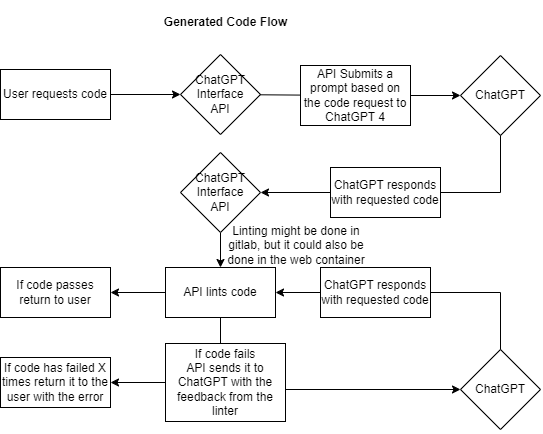
Description automatically generated**

**User API Endpoint Diagram**

**ChatGPT 4 Interface API**

**A screenshot of a computer

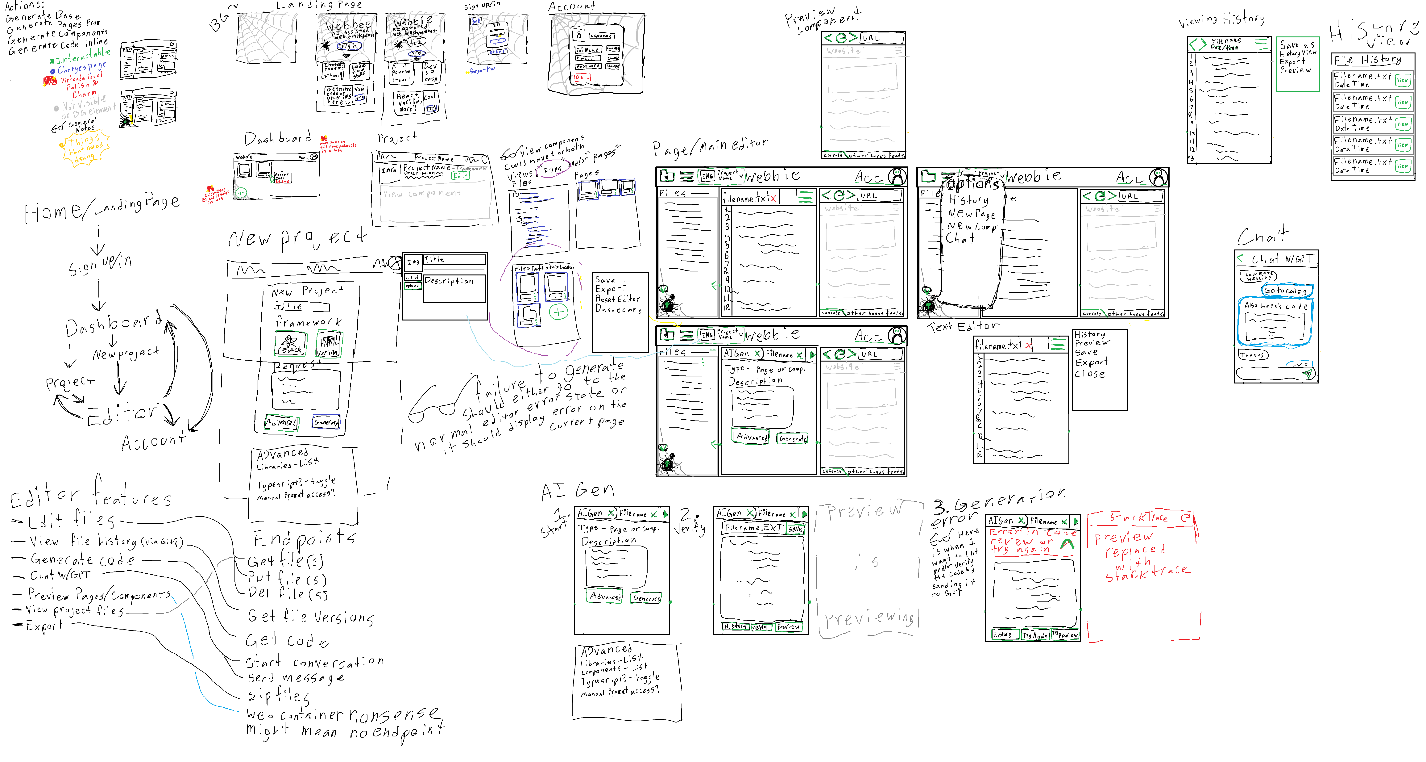
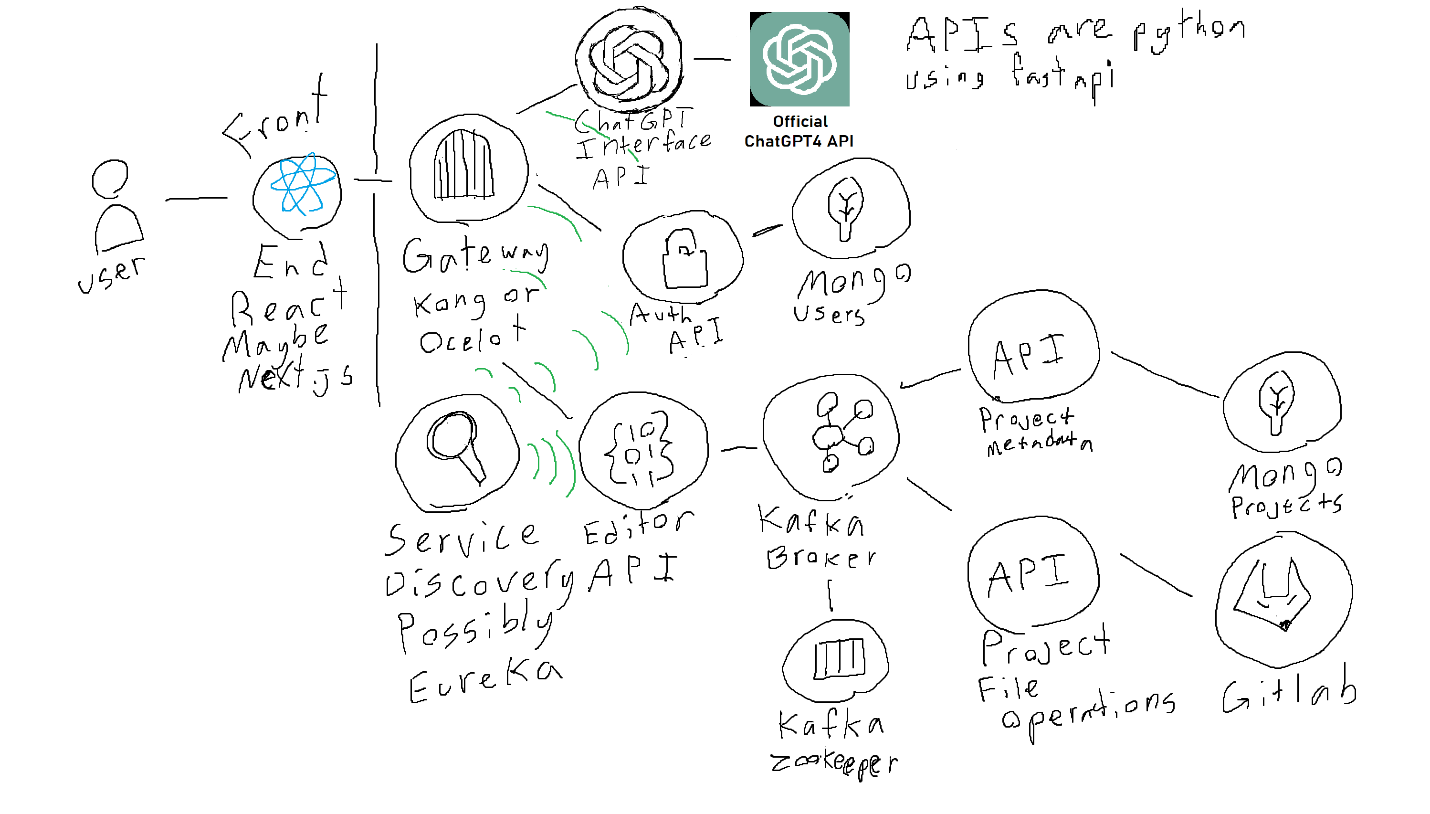
Description automatically generated**

**Generated Code Flow**

**Page Flow Diagram**

**A diagram of a software project

Description automatically generated**

**MS Paint Diagrams, UI and Service Diagram**