## 

## 

## 

## 

## **ChatMaps Deliverable 2**

Software Architecture

Stephen Goodridge, Clark LaChance, Nicholas Pease, Joseph Gallant, Aidan Bradley

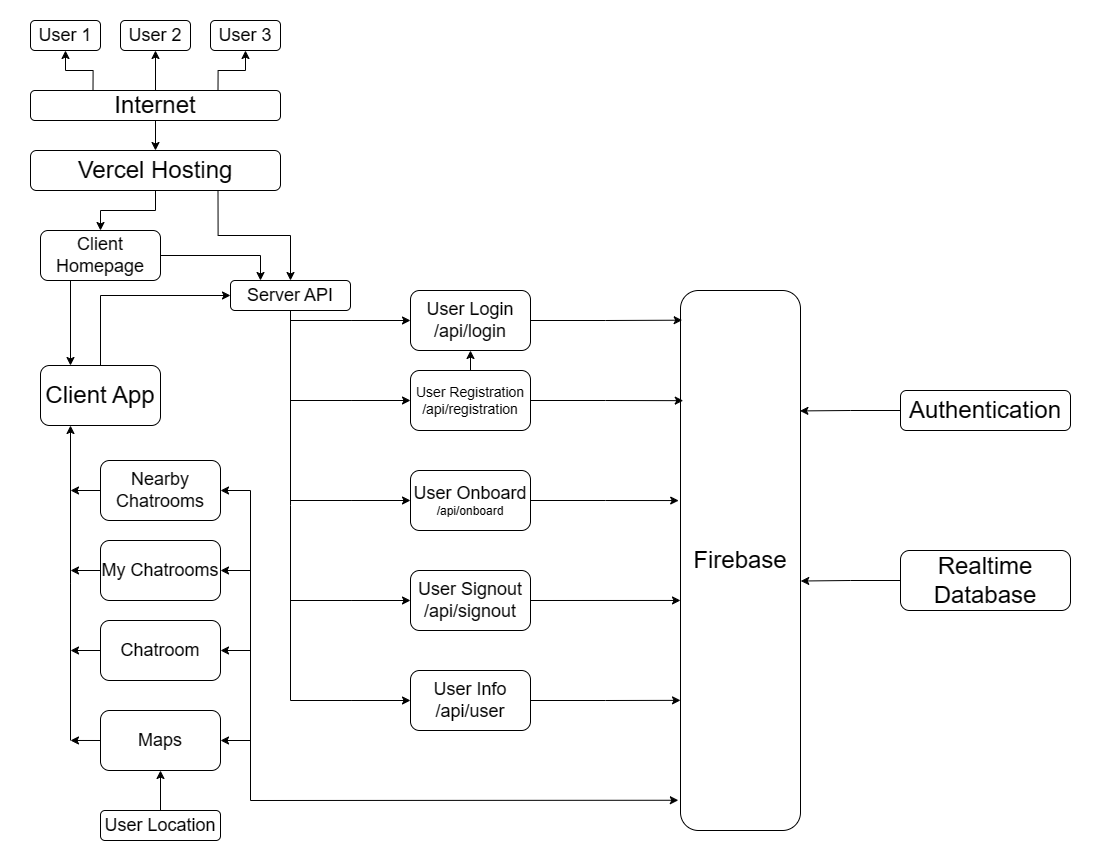
COS420

3 March 2024

Github: <https://github.com/ChatMaps/ChatMaps>

Kanban: <https://trello.com/b/TaygvBv7/chatmaps>

**Architectural Design Document**



For ChatMaps, we chose a client-server architecture as depicted in the diagram above. In a client-server architecture, all systems in an application are accessible at differing endpoints or via different servers. At the top you can see very little is rendered on the users end, and most of the app is hidden behind our hosting provider Vercel.

We chose to design our app this way to maximize the performance on the user’s end, thus improving the user experience. Due to the structure of React, on the server side we have a client and a backend area. The client area consists of most everything that the user directly interacts with, whereas the backend area (labeled as the Server API) exists as a layer between our application and our external authentication and database provider Firebase. When using our application, we perform more of a handoff process where after authentication is verified, control is shifted to the Client App box, which then utilizes all of the subsystems listed underneath to provide interactivity to the user.