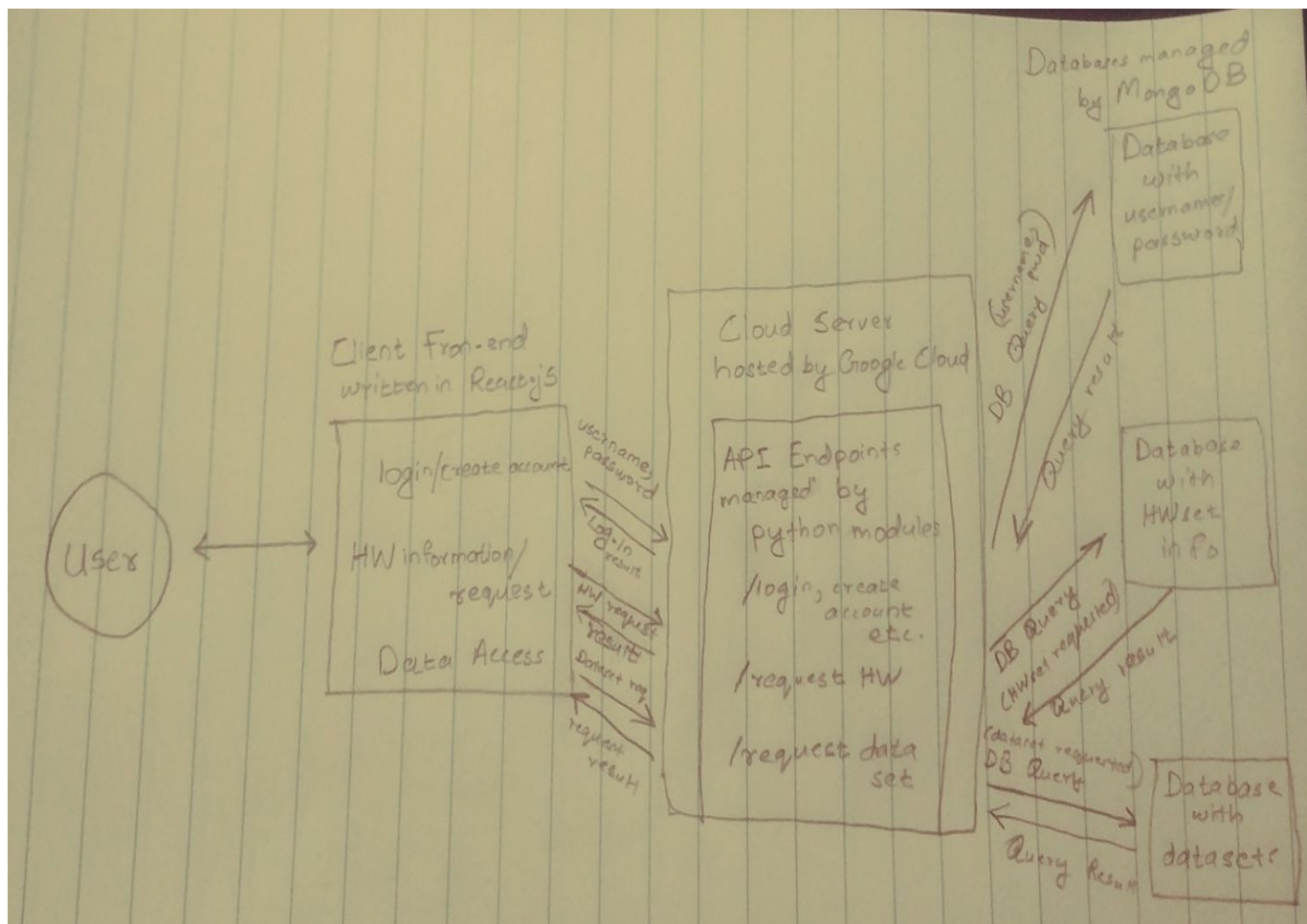


Project Plan

Project summary

The purpose of this project is to build and design a website that allows users to request hardware resources for their usage for a designated time interval. The user interacts with a website which sends and receives information about the available resources to and from a server. The server will have a database which keeps track of all available resources and interfaces with an API that updates and allocates the resources real-time. The server will store user information such as encrypted username and password and current hardware allocation to that user.

Sketch



Tools and Approaches

Coding will be done in Python because of its flexibility and the plethora of available libraries for the backend. Testing will be done using PyTest. PyTest allows for writing and sequential testing of original test cases with detailed feedback not provided by other

testing software. PyTest will allow us to thoroughly test our API and backend systems even as they interact with the frontend. PyTest also has many library extensions available should we run into a problem that is difficult to test with the base library alone. All these factors make it the best choice for testing our system over the duration of the project.

The website will be hosted using Google Cloud. Google Cloud allows a strong control and security of the cloud platform. Google Cloud machines also have enhanced execution allowing several visitors at any time. The live migration feature would also make it easier to host the server side code.

The front end will be designed using React. React allows for streamlined user interface development. With React, we can design a website that creates smooth animations, eye catching visuals, and uses a modern, neat design.

Feature Summaries

The front end will provide a login service for all users. The user will be able to access their resources by providing a UserID and a password. These usernames and passwords will be encrypted and stored in the database. All logins will encrypt the provided username and password and check for a matching pair in the database unless it is a new registration. The user will be made aware of the currently available hardware resources and have the option to request new hardware resources should it be less than the currently available total. In the event that a user requests more than available or their request fails for another reason, the front end web application will prompt the user to enter a lower request after notifying them of their request's failure. Any and all hardware that is reserved by the user will be exclusively used by the user until the reservation expires. The database will keep track of the currently allocated resources to the encrypted account information and will keep all data organized and secure. Finally, there will be an API which interfaces between the front and back ends and accepts and fulfills requests from the users.

Github Usage

GitHub repository will be used to keep track of all the code written and will be updated frequently. The issue tracker will be used to record bugs and additional features as the project is developed. The project board will be used to keep track of the tasks that need to be completed, in progress, and completed as well the user stories for the project. Any technical debt that arises will be noted and put back in the in-progress column for improvement. Each of the project members will have their own branch and commit and pull frequently to keep branch current. All changes to the master branch will only come after thorough testing.

Conventions

We will use snake_case for our variables, functions, methods, and classes. We will not employ global variables unless it is a global constant. All requests from the users will be routed and handled through the API. Try-catch for error handling. We will use 1 tab for any required whitespace. We will end every statement with a semicolon, “;”. All conditional statements will contain parentheses. In JavaScript code, we will only use function expressions, unless the function should only execute one statement, i.e. an add function that simply adds multiple numbers, in which case it will use an arrow function.