**Technical Review & Design document**

**Data Analysis & Cyber Security**

**Objective**: why you are building this system.

We are building this system because there is a power and water data collection station at the college, wanted to make an application that works with these data collections and use Artificial Intelligence to make predictions to water and power usage in correlation with the weather.

**Requirements**:

* + Functional Requirements – product features or functions
    - Use case diagram can capture functional requirements
  + Non-Functional Requirement
    - Quality constraints (e.g. performance, security portability, etc…)

As can be seen in figure 1 below, the user will log into the website, their login attempt will be validated by the system to make sure that they’re a valid user. They will then be taken to the Dashboard, from which they can choose to go to the Predict Page or the Graphs Page. The three main pages in the website will be accessible at all times through the use of a navbar. If the user decides to go to the Predict page, they will be met with some fields to enter information. Once the user enters the required details, a prediction will be made and displayed to the user.

Diagram

Description automatically generated

Figure 1: Use Case Diagram

**Architectural Design**

**User Interface** –



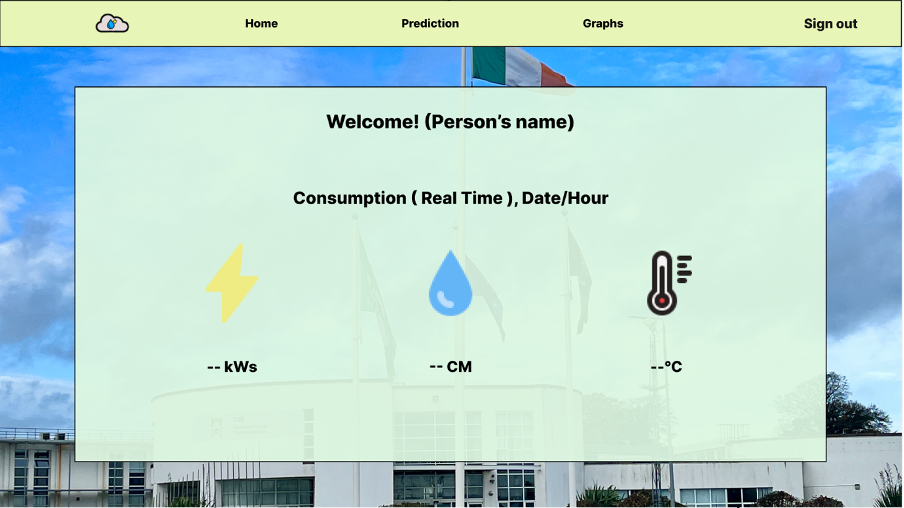
***Figure 2: Login Page***

Since our proposal few changes were made in the architecture, the main one is the background image, now it is our own photo of TUS campus in Thurley. This is the first page that the user is going to see.



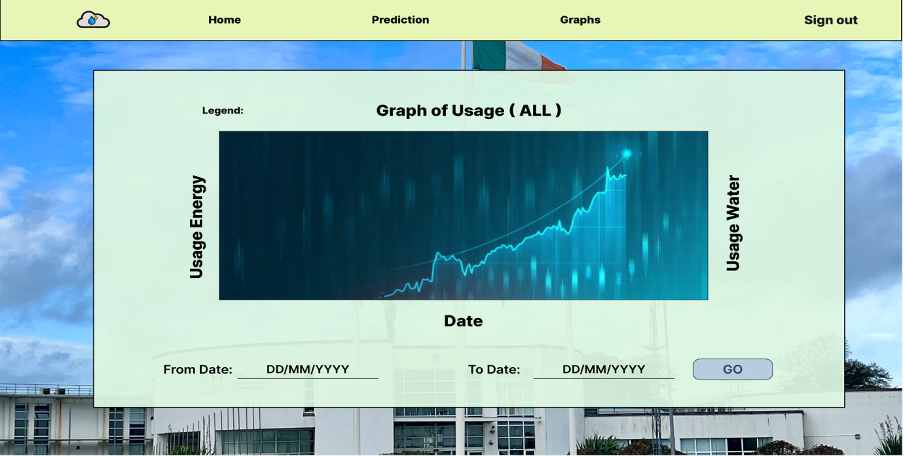
***Figure 3: Create an Account Page***

If the user has no account that is the page that they are going to access to create an account.



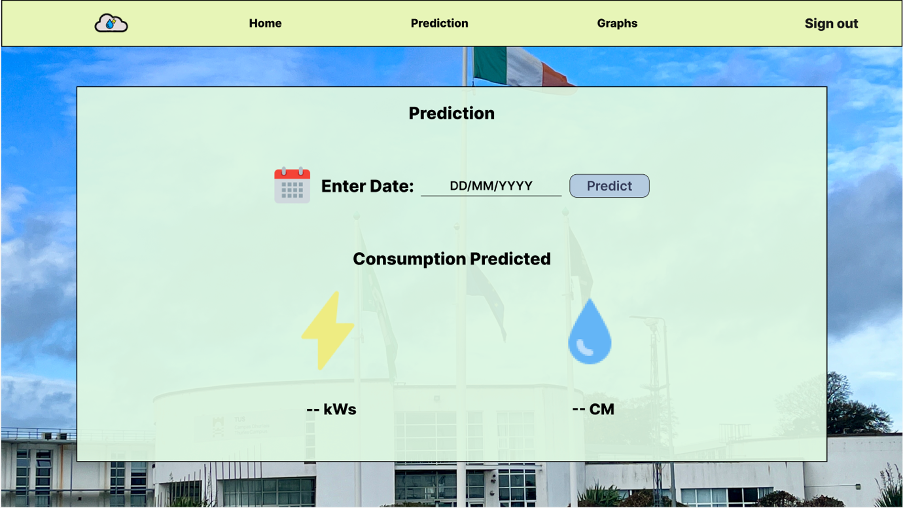
***Figure 4: Dashboard Page***

Once the user validates their account, that is the first page that they are going to see, the dashboard, this page will display the electricity and water usage and the temperature of Thurley in real time.



***Figure 5: Graphs Page***

The graphs page is going to show all time usage, since the end of 2013 in a line graph, both electricity and water. The user also can narrow this graph by the dates.



***Figure 6: Prediction Page***

Predict page is where the user will be able to see how much consumption the campus will have in the max two week time. The user will enter the date and the page will show the possible energy and water that will be consumed.

**Technical Review**

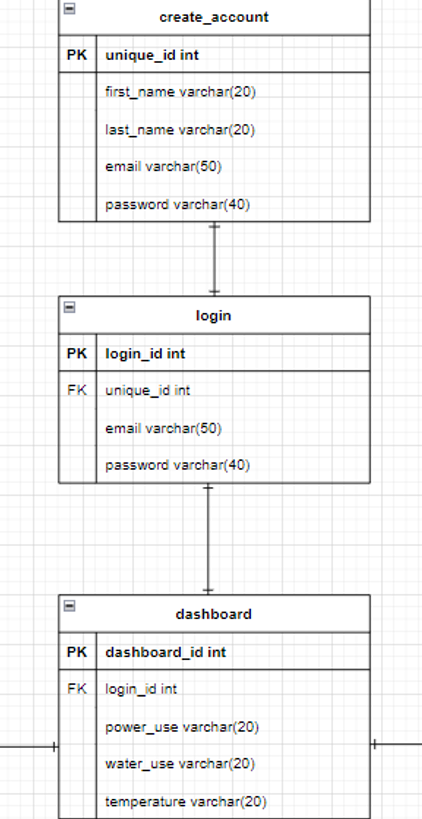
The technologies that will be used:

* **Front-end:** Angular.
* **Back-end:** Python and MySQL.
* **Authenticator:** AWS Amplify, Firebase can be an alternative.
* **Web-Host:** Amazon Web Services, Google Cloud Platform can be an alternative.
* **Interface design:** Figma.
* **Repository:** GitHub, Docker can be an alternative.
* **Communication:** Discord.
* **ERD-Diagrams:** Dia.

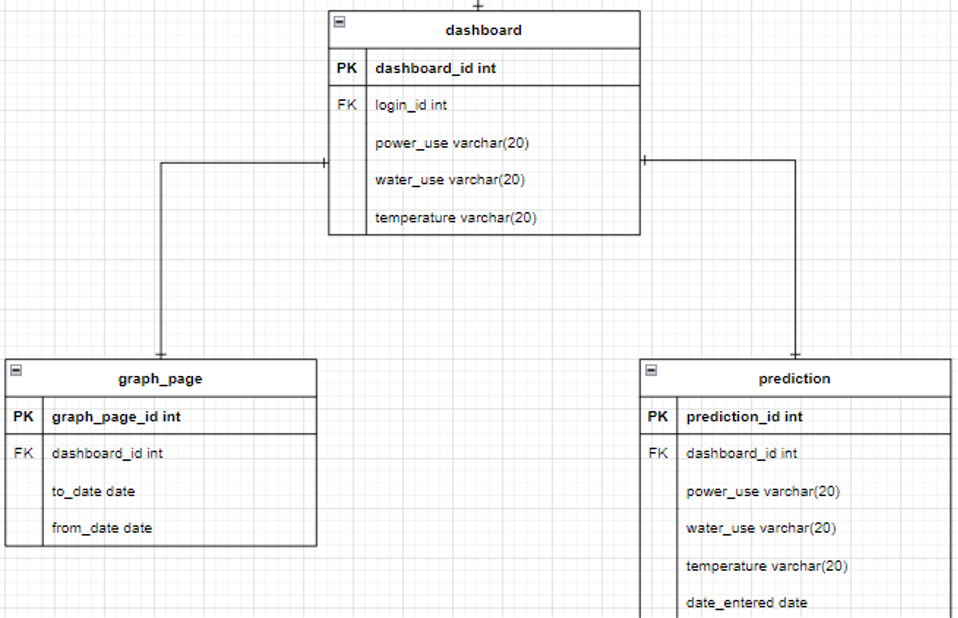
**Data Storage – Model ERD**

A design of your database schema (ERD), Mapping of the ERD to a set of relations (tables).

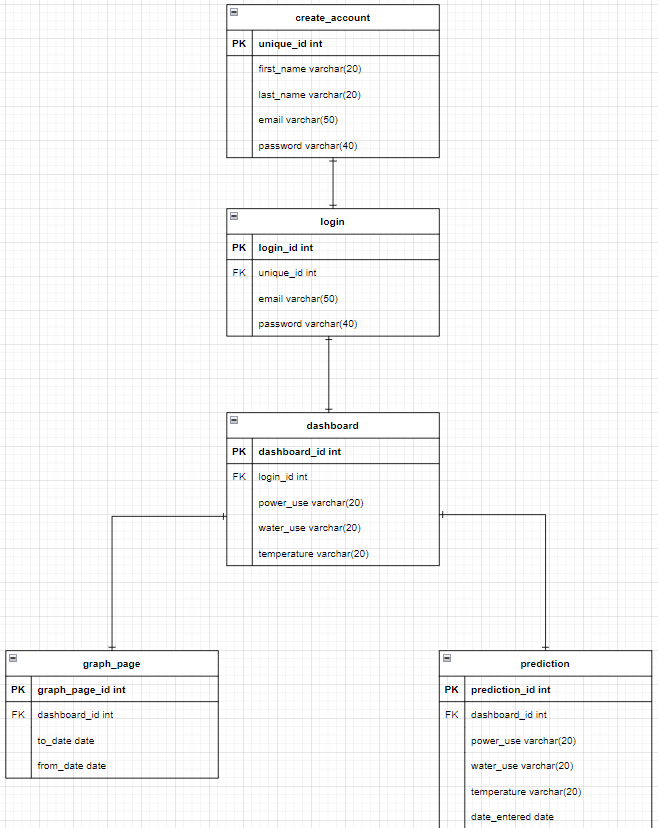
As shown in Figure 2, Figure 3 and Figure 4 all the tables will have 1 to 1 relationship with each other to keep the database as simple as possible, all the primary and foreign keys will be integer values to keep uniformity while using varchar to store the power, water and temperature. In Figure 3 the graph and prediction tables are connected through the dashboard to maintain structural integrity, both have the dashboard unique id as foreign keys.



***Figure 2: Creating an Account, Login and Dashboard***



***Figure 3: Dashboard linking to Graph and Prediction pages***



***Figure 4: Full ERD Diagram***