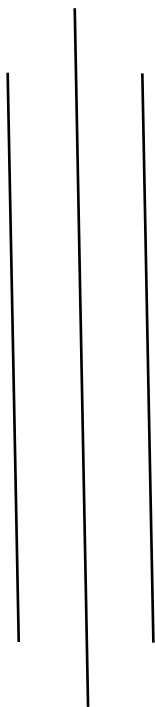# 4CS017 – INTERNET SOFTWARE ARCHITECTURE
Prototype 3

Nijmi Bajracharya
ID: 2508912

A Short Reflective Summary

Introduction

In this prototype3, I have extended this application by using HTML, CSS, JavaScript, PHP and MYSQL. In this prototype I have included data catching in order to optimize the performance and reduce the number of API calls. It also includes local storage to allow the limited offline access of the previously fetched weather data.
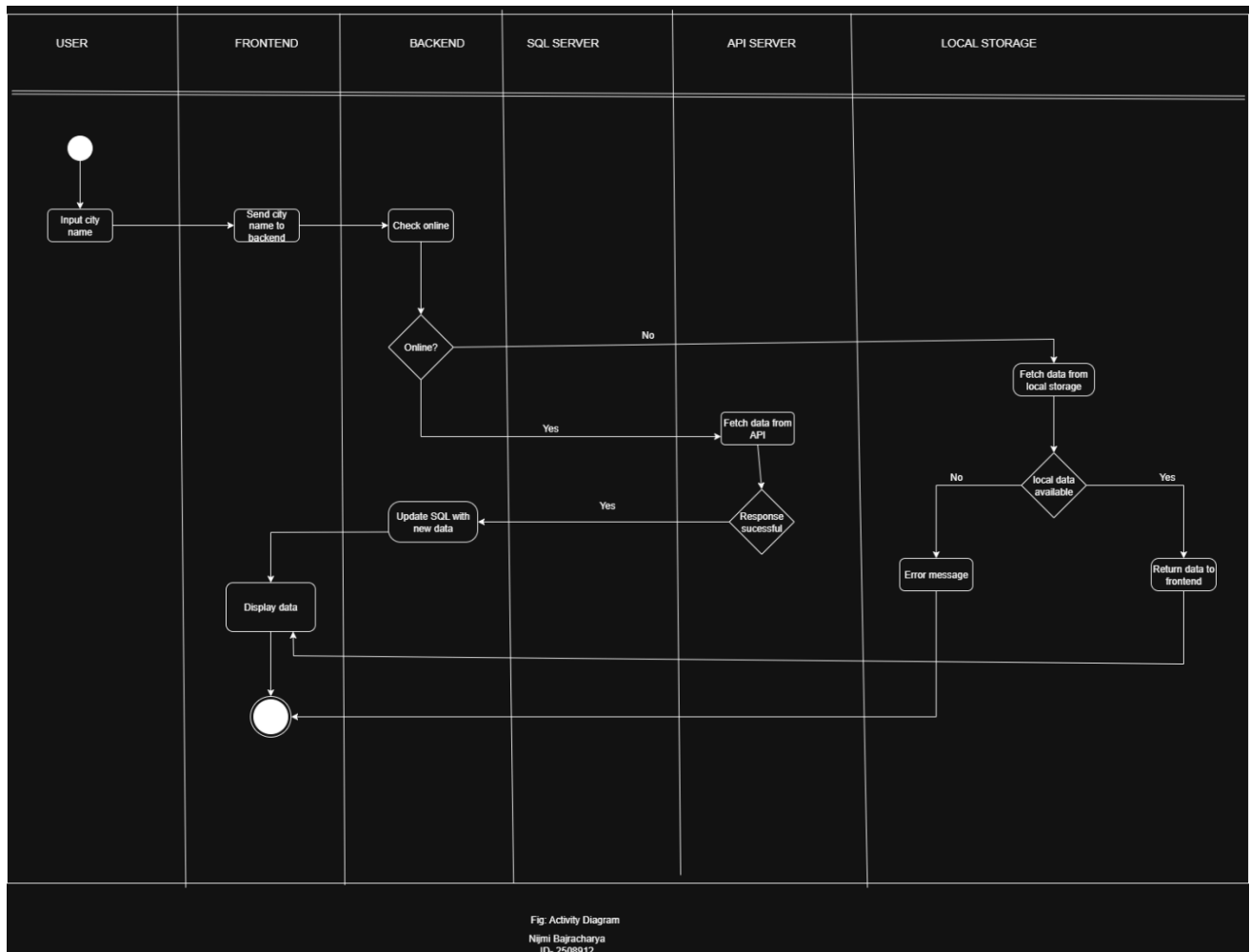
Strengths

a. Simple and easy to debug- As the codes are done in HTML, CSS, PHP and JavaScript the codes can be easily identified and debugged. Also, the codes are easy to understand and learn.

b. Updating of weather- By using JavaScript and MySQL, it helps to fetch real time update with the help of Open Weather Map API and also weather data less than 2 hours old is considered valid for display. If the existing data exceeds this time threshold, the system fetch updated information from the Open Weather API.

c. Proper storage- The SQL database helps to store weather data like temperature, humidity, etc and the time when it was recorded.

d. Offline access of weather data: Local storage allows users to access weather information even when they don't have an active internet connection.

e. Improvement in performance: Storing weather data reduces the need for repeated network requests and speeds up the app's performance. Likewise, users can also get the data quickly without waiting for it to load from the server each time.

Weakness and Challenges

a. Lack of database performance- It makes our system slower when we search for weather in many cities.

b. Old Data: Since local storage saves data for offline use, there is a risk of displaying outdated weather information if the app isn't able to fetch new data due to connectivity issues.

c. Accuracy of data- The weather conditions might not be accurate for fast changing weather conditions.

d. Limited storage: The local storage is usually limited to a certain amount of space (around 5MB per domain in most browsers), so it may not be suitable for storing large sets of weather data, especially with detailed forecasts for multiple locations.

This prototype was the most complex one while connecting JavaScript with PHP and MYSQL databases logics. In this prototype I have learnt server side scripting using PHP and also implementing the offline fallback using local storage. This version of weather app is more advanced, efficient and user friendly.

UML Activity Diagram:

Fig: Activity Diagram
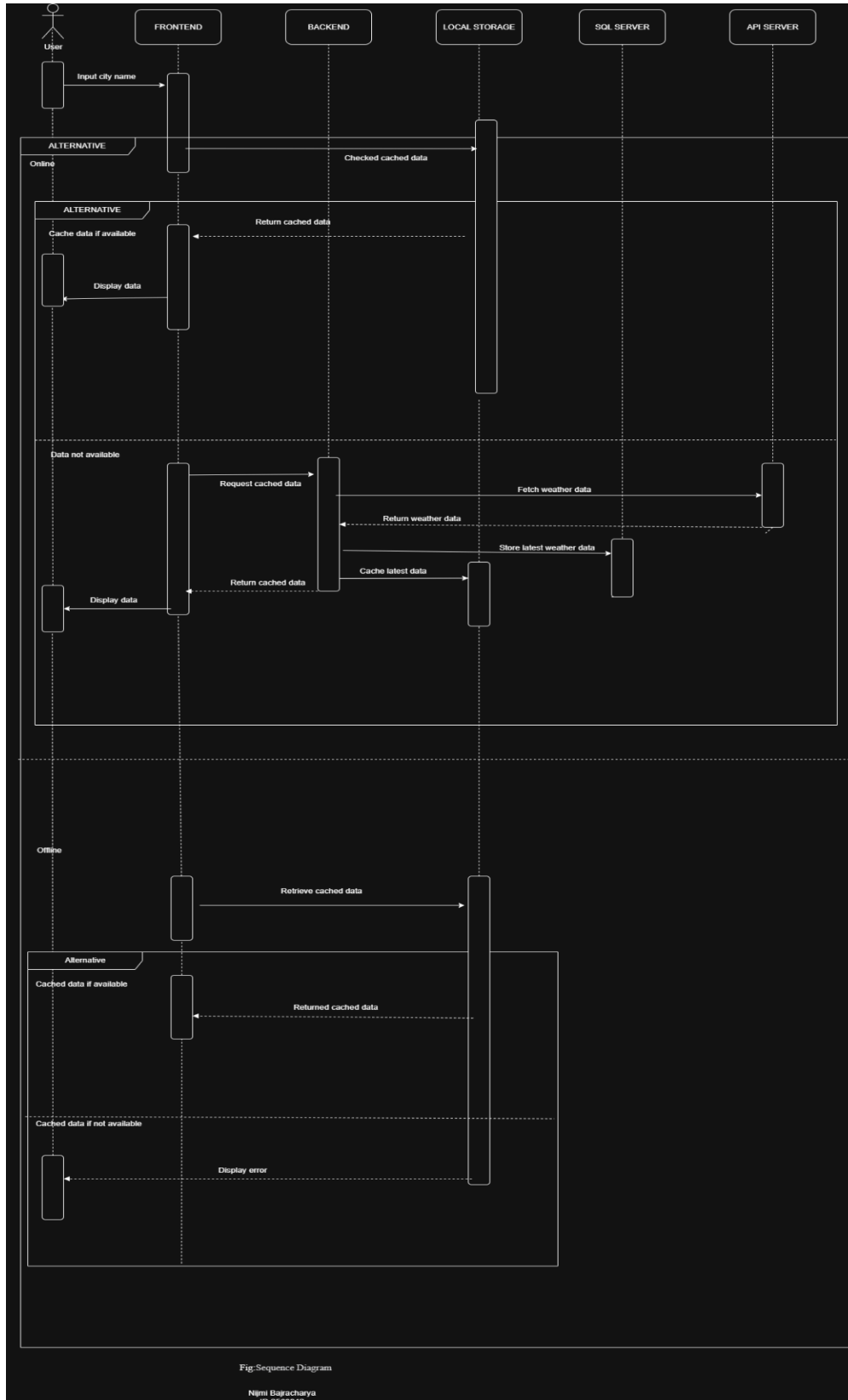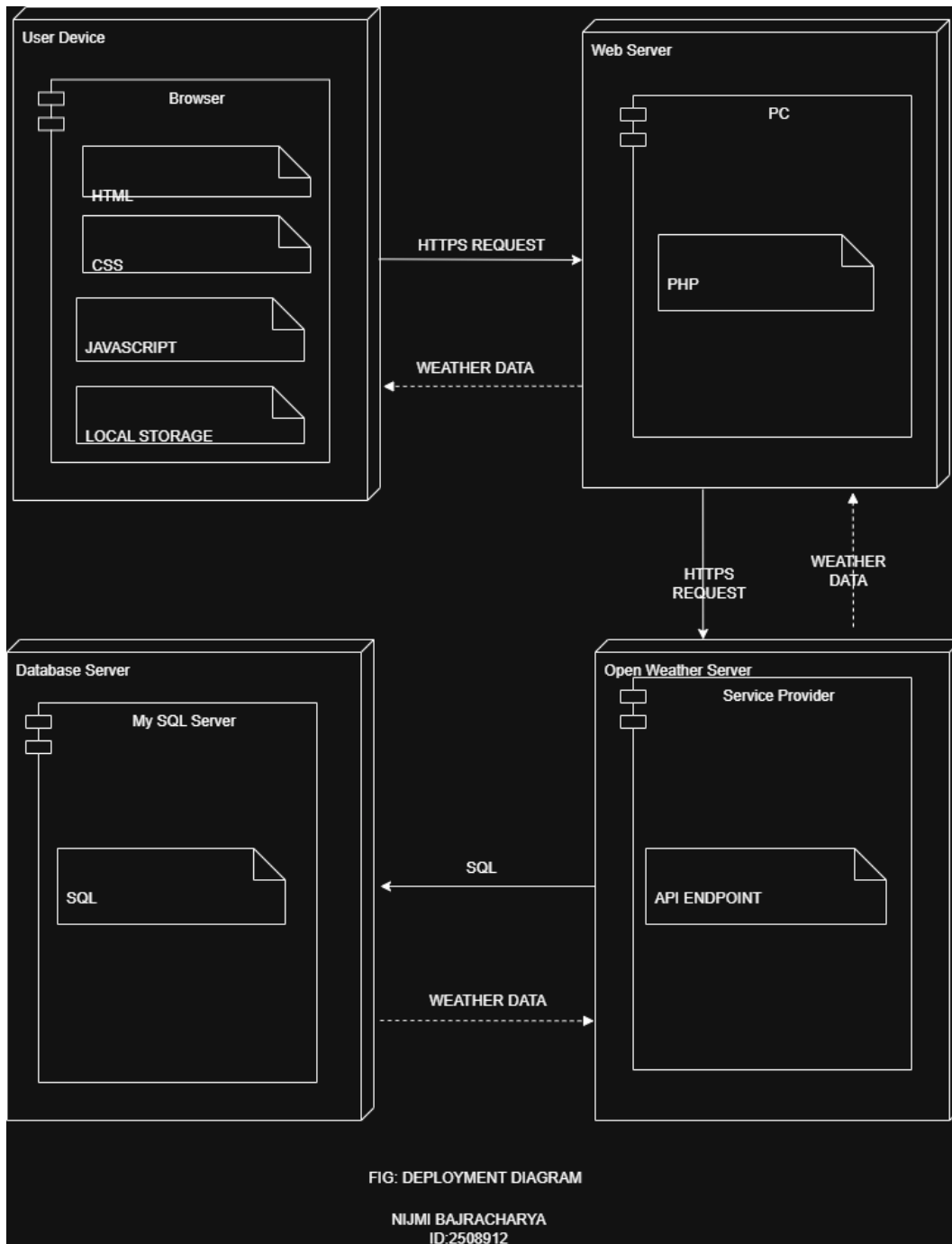Nijmi Bajracharya
ID- 2508912

UML Sequence Diagram:

Fig:Sequence Diagram

Nijmi Bajracharya
ID:2508912

UML Deployment Diagram:

FIG: DEPLOYMENT DIAGRAM

NIJMI BAJRACHARYA
ID:2508912

Data stored in browser:

YOUTUBE LINK- https://youtu.be/m1USxsQ5Knc

WEBSITE LINK- http://nijmi.free.nf/prototype3/nijmi.html