

Project - Traffic Monitoring

Bianca Mariciuc (2E1)

1 Introduction

Traffic Monitoring is an application designed to make traffic more organized and efficient, ensuring smoother journeys for drivers. Drivers are allowed to connect and receive traffic assistance. The main purpose for this application is to make the driver's life easier during driving. Drivers can receive information on the weather, fuel prices at nearby stations, and various sports events that are happening in the city they are driving through. Also, in this application, drivers can alert and be alerted by other registered users about accidents. In addition, drivers can receive warnings based on their speed, which is automatically updated every minute during their journey, helping them drive more safely and responsibly.

2 Applied Technologies

For this project, I chose to implement a concurrent **TCP (Transmission Control Protocol)** server that creates a thread for each connected client. I chose this protocol over others because TCP ensures accurate information transmission by establishing a virtual connection between the sender and receiver, which is crucial for my project, since drivers rely on precise and reliable data. Moreover, the implementation must be concurrent because the application needs to support multiple drivers simultaneously. This issue can be solved by creating a separate thread for each connected client, in this way the server can manage each client independently. Also, threads share the same memory space and resources of the parent process, making them more lightweight compared to processes.

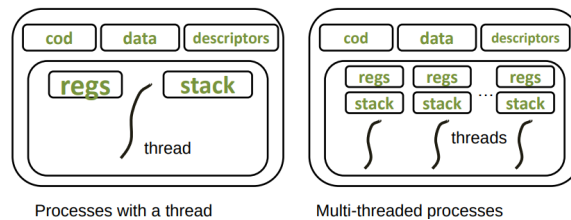


Figure 1: Threads share the memory space and resources of the parent process but maintain their own stack and registers.

3 Application Structure

The structure of the application is quite simple. Drivers are considered clients that communicate with the application server. Every client must follow a specific logic when requesting information from the server, this means that the client cannot receive any data unless they are logged in or registered. During a session, every client is allowed to send requests and wait for a response. Below is a diagram illustrating how communication between the client and server is made.

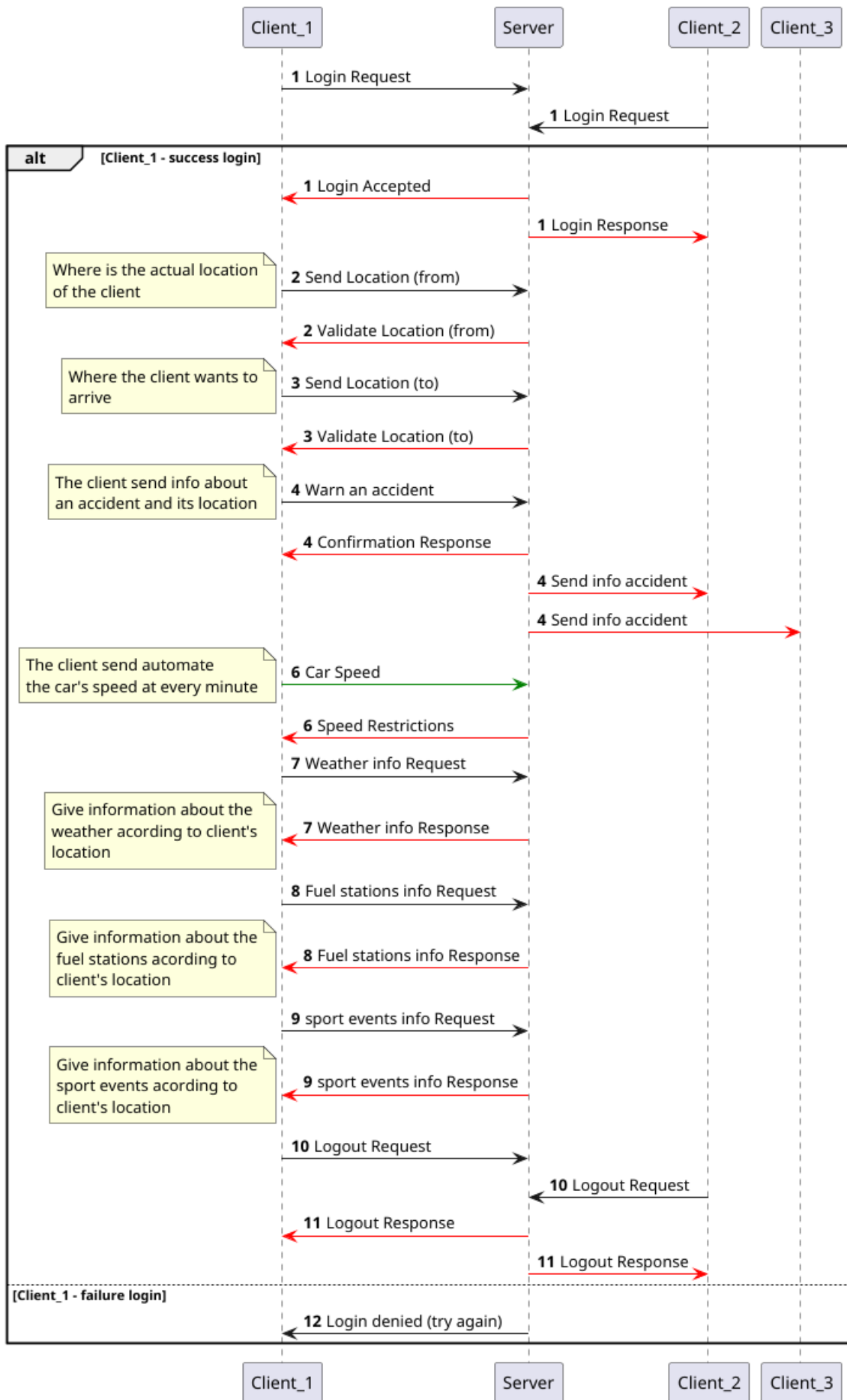


Figure 2: Sequence Diagram - Requests from points 1 to 3 must be executed in order, while requests from points 4 to 9 can be made in any order. 2

The diagram shows that certain information must be stored to provide a response to a request. To fix this problem, we will use a database to store all the necessary information.

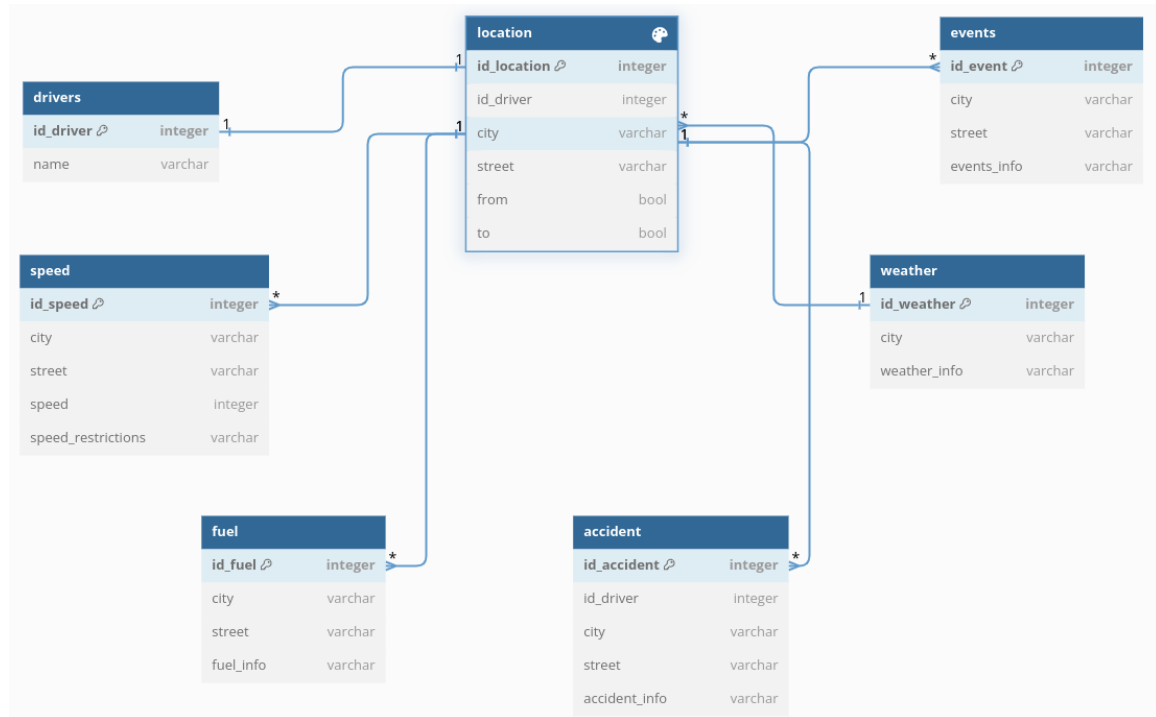


Figure 3: Information Organized In A Database

In this diagram we have 7 tables: drivers, location, events, speed, weather, fuel and accident. The "drivers" table stores client information. The "location" table stores the current location of the client and their destination. The "events" table contains details about events occurring in a city. The "speed" table holds the speed limits for specific streets in a city and their reasons. The "weather" table provides weather data for each city. The "fuel" table contains information about fuel stations in a city, and the "accident" table lists active accidents registered in specific city.

4 Implementation Aspects

Firstly, to display the automated speed of a car, I implemented a function that utilizes some primitives from the *time.h* library. These primitives help me generate and print a new speed every minute using a function that incorporates *rand()*.

Here is the function that generates the speeds:

```

1 void random_speed() {
2     int speed = rand() % 81;
3     printf("Actual speed: %d\n", speed);
4     fflush(stdout);
5 }
  
```

And this condition is checked during every iteration of the main while loop.

```

1 currentTime = time(NULL);
2     if (difftime(currentTime, lastTime) >= 60) {
3         random_speed();
4         lastTime = currentTime;
5     }
  
```

5 Conclusion

The Traffic Monitoring application is created to improve traffic organization and make driving easier by giving drivers real-time updates about weather, fuel prices, sports events, and accident alerts. It also helps drivers stay safe by sending speed warnings every minute, encouraging more responsible driving.

The application uses a concurrent TCP server to ensure reliable data transfer and smooth communication between the server and multiple drivers. By assigning each client to its own thread, the server can handle multiple users at the same time without being affected by slower or problematic clients.

To make the application better, adding data encryption could increase security and protect drivers' information. Also, using machine learning for traffic predictions could provide more advanced and helpful advice for drivers.

6 Bibliography

1. <https://www.geeksforgeeks.org/what-is-transmission-control-protocol-tcp/>
2. <https://www.geeksforgeeks.org/thread-in-operating-system/>
3. Lecture 7 - Computer Networks UAIC
4. <https://plantuml.com/sequence-diagram>
5. <https://docs.dbdiagram.io/>
6. <https://www.overleaf.com/learn>