**UNIVERSITY OF BUEA** **REPUBLIC OF CAMEROON**

P.O Box 63, PEACE-WORK-FATHERLAND

Buea, South West Region

[Tel: (237)](Tel:(237)) 674354327

Fax: (237) 3332 22 72

**FACULTY OF ENGINEERING AND TECHNOLOGY**

**DEPARTMENT OF COMPUTER ENGINEERING**

CEF 440: INTERNET AND MOBILE PROGRAMMING

**TASK 6: DESIGN AND IMPLEMENTATION OF THE DATABASE OF SAFEWAKA – A ROAD STATE AND ROAD SIGN NOTIFICATION APP**

Presented by

**GROUP 4**

COURSE MASTER

**Dr. Nkemeni Valery**

Academic year: 2023/2024

**GROUP MEMBERS**

|  |  |
| --- | --- |
| **NAMES** | **MATRICULE** |
| DJOUMESSI NGEUFACK IVAN | FE21A174 |
| TETUH WIBINAH ENGONWEI | FE21A322 |
| GATCHUINNE MOMO MURIELLE CYNTHIA | FE21A200 |
| DJOUMESSI DONGMO RONSARD ARNEGIE | FE21A172 |
| NJOYONG GODWILL EGENECHA JUNIOR | FE21A273 |

**TABLE OF CONTENTS**

Contents

[**Introduction** 4](#_Toc169788458)

[**Database Schema** 4](#_Toc169788459)

[**Implementation** 7](#_Toc169788460)

[**SQL codes** 7](#_Toc169788461)

[**ER Diagram** 9](#_Toc169788462)

[**Queries** 9](#_Toc169788463)

[**Database Security and Reliability** 10](#_Toc169788464)

[**Conclusion** 11](#_Toc169788465)

[**References** 12](#_Toc169788466)

**Introduction**  
In the modern era of smart transportation, the need for comprehensive road information and real-time updates has become increasingly vital for drivers. The road state and road sign notification app aim to address this need by providing a user-friendly mobile application that offers intuitive access to road sign information and up-to-date insights on road conditions. This report delves into the design and implementation of the database that powers the core functionalities of the app.

**Database Schema**  
The database schema for the road state and road sign notification app consists of the various tables in our database. Below are key tables in our database:

1. **Road State**

This table stores information about the current state of the road.

**Columns**

*stateID*: Unique identifier for the road state. Acts as the PK(primary key) of this table

*eventype*:Describes the current event of road at a given time.

*severity*: Describes the urgency of the road state event.

*location*: Gives the exact location of the road state event.

*timestamp*: Gives the time at which the driver encounters the road state.

1. **Navigation**

This table provides adequate navigation for the car drivers.

**Columns**

*navID*: Unique identifier for the navigation template. Acts as the PK(primary key) of this table

*appName*: Gives the name of the navigation provider.

*version*: Gives the version of the navigation provider.

*userID:* Describes the ID of the user.

*original\_location*: Gives the initial location of the driver before embarking on a journey.

*destination\_location*: Gives the destination location of the driver.

1. **RoadSign**

This table contains detailed information about the road signs located along the road network, including the sign type, text or symbols, GPS coordinates, and any additional metadata.

**Columns**

*signID*: Unique identifier for the sign template. Acts as the PK(primary key) of this table.

*sign\_type*: Describes the road sign the driver encounters on the road.

*interpretation*: Gives the meaning of the road signs.

*location*: Gives the location of the road sign.

*imageURl*: Provides the link to the images of the various road signs.

1. **User**

This table stores the individual preferences and notification settings for each user of the app(drivers) allowing them to customize the types of alerts and information they receive.

***Columns***

*userID*: Unique identifier for the user template. Acts as the PK(primary key) of this table.

*username*: Gives the name of the driver.

*setting*: Describes the personalized settings of the driver.

*location*: Gives the current location of the driver.

1. **Notifications**

This table logs all the notifications sent to users, including the type of notification, the recipient, the timestamp, and any relevant details.

**Columns**

*notifyID*: Unique identifier for the notification template. Acts as the PK(primary key) of this table.

*userID*: Gives the ID of the driver.

*stateID*:Gives the ID of the road state.

*priority*: Describes the priority of the notification.

*signID*: Gives the ID of the road sign.

1. **Developer**This table stores the feedbacks from the users regarding the app.

**Columns**

*developerID*: Unique identifier for the developer template. Acts as the PK(primary key) of this table.

*name*: Gives the name of the admin

*email*: Specifies the email of the admin that receives feedback about the app.

1. **Route**

This table is used to give the drivers the best route for a journey using the initial and destination locations of the driver.

**Columns**

*routeID*: Unique identifier for the route template. Acts as the PK(primary key) of this table.

*original\_location*: Gives the initial location of the driver before embarking on a journey.

*destination\_location*: Gives the destination location of the driver.

*duration*: Gives the duration of the journey using the optimized route.

*instruction*: Describes the directives given to the driver using the best route.

# **Implementation**

This section covers the implementation of our database using MySQL. The choice of the database is mainly because of consistency, reliability and performance.

**SQL codes**

**-- User Table**

CREATE TABLE App\_user (

userID INT PRIMARY KEY,

username VARCHAR(50) NOT NULL,

email VARCHAR(100) NOT NULL,

location VARCHAR(100) NOT NULL,

userType ENUM('Driver', 'Pedestrian', 'StateAgency') NOT NULL

);

-- **Roadsign Table**

CREATE TABLE Roadsign (

signID INT PRIMARY KEY,

signType VARCHAR(50) NOT NULL,

meaning VARCHAR(100) NOT NULL,

location VARCHAR(100) NOT NULL,

imageURL VARCHAR(200) NOT NULL

);

-- **Roadcondition Tabl**e

CREATE TABLE Roadcondition (

conditionID INT PRIMARY KEY,

eventType VARCHAR(50) NOT NULL,

severity INT NOT NULL,

location VARCHAR(100) NOT NULL,

conditiontime DATETIME NOT NULL,

userfeedback TEXT,

userID INT ,

FOREIGN KEY (userID) REFERENCES App\_user(userID)

);

-- **Notification Table**

CREATE TABLE Notification (

notifyID INT PRIMARY KEY,

userID INT NOT NULL,

signID INT NOT NULL,

conditionID INT NOT NULL,

priority INT NOT NULL,

FOREIGN KEY (userID) REFERENCES App\_user(userID),

FOREIGN KEY (signID) REFERENCES Roadsign(signID),

FOREIGN KEY (conditionID) REFERENCES Roadcondition(conditionID)

);

-- **Navigation Table**

CREATE TABLE Navigation (

navID INT PRIMARY KEY,

appName VARCHAR(50) ,

version VARCHAR(20) ,

userID INT NOT NULL,

originalLocation VARCHAR(100) NOT NULL,

destinationLocation VARCHAR(100) NOT NULL,

RouteID INT NOT NULL,

FOREIGN KEY (userID) REFERENCES App\_user(userID),

FOREIGN KEY (RouteID) REFERENCES Route(RouteID)

);

-- **Route Table**

CREATE TABLE Route (

RouteID INT PRIMARY KEY,

originalLocation VARCHAR(100) NOT NULL,

destinationLocation VARCHAR(100) NOT NULL,

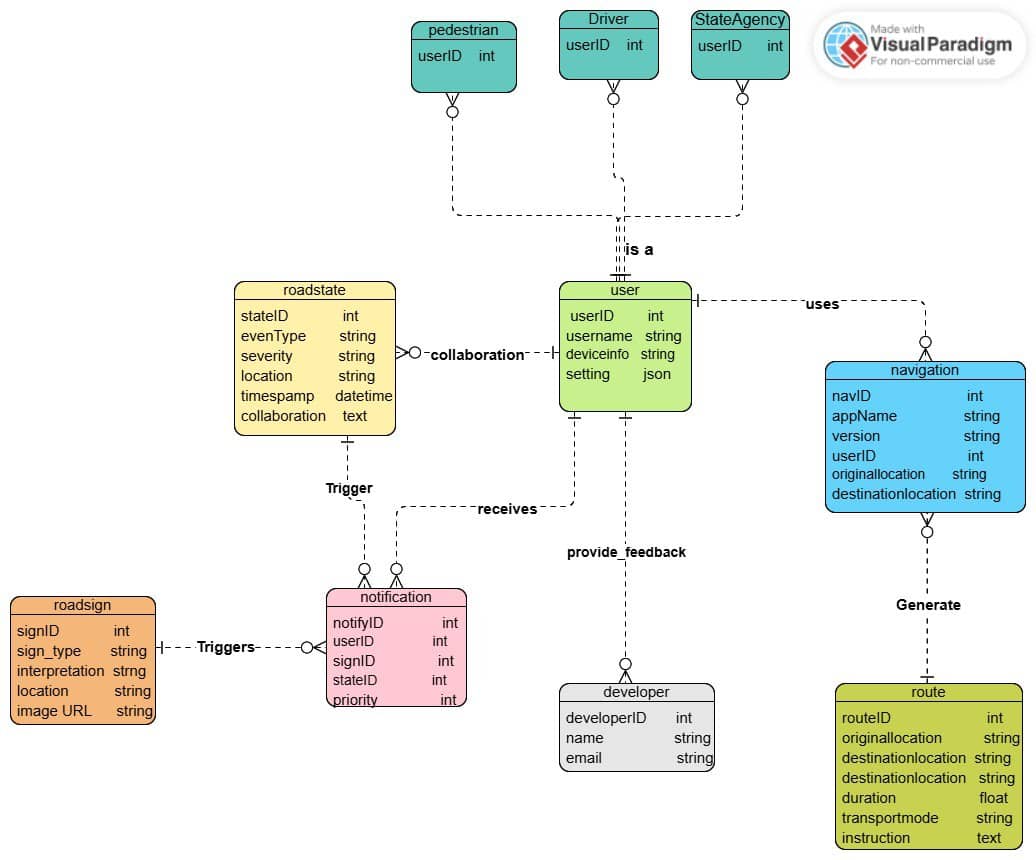
distance FLOAT NOT NULL,

duration INT NOT NULL

);

## **ER Diagram**

This diagram depicts the relationships between entities (tables and objects) in our database.

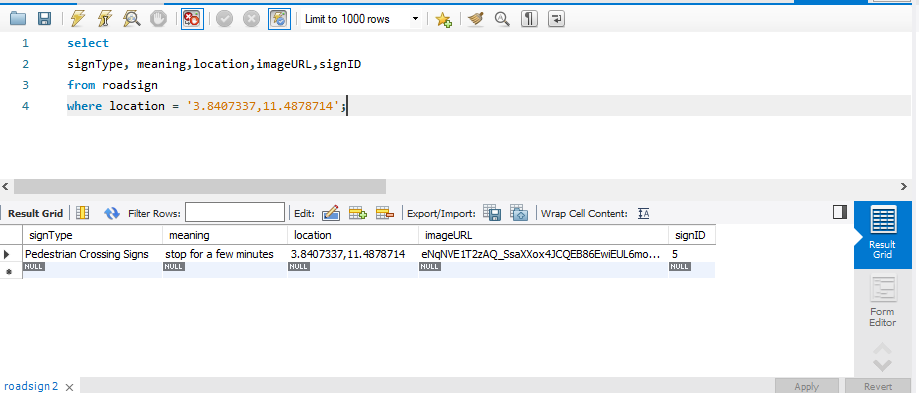


*Figure1: ER diagram of the database of SafeWaka*

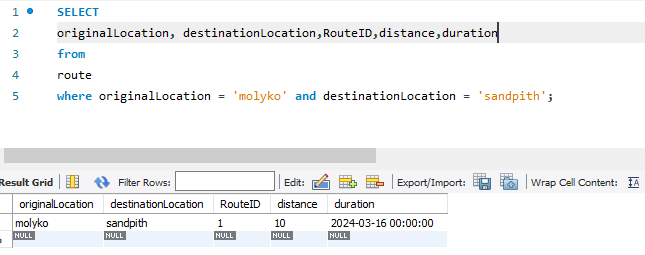
## **Queries**

Below are queries for some scenarios;

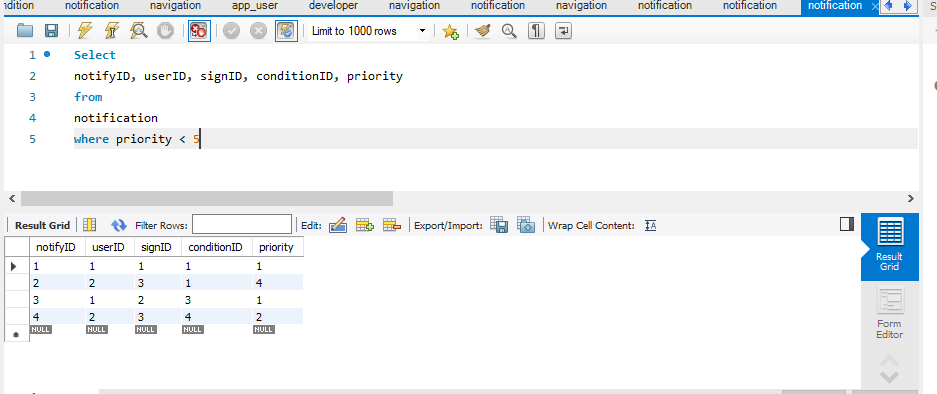
1. Display Road signs



1. Route optimization



1. Notification



**Database Security and Reliability**  
The database powering the road state and road sign notification app is designed with robust security and reliability measures, including:

* Secure authentication and authorization mechanisms to control access to the database
* Encryption of data at rest and in transit to protect sensitive information
* Automated backups and disaster recovery capabilities to ensure data integrity and availability
* Scalable infrastructure and load-balancing to handle increased user traffic and data growth
* Comprehensive logging and auditing to monitor database activities and detect any suspicious events

These security and reliability measures are carefully implemented to safeguard the app's data and provide a reliable service to users.

**Conclusion**  
By designing a comprehensive and well-structured database, the road state and road sign notification app can effectively store and manage the necessary data to deliver a user-friendly and informative experience for drivers. The app's ability to integrate with real-time data sources, provide customizable user preferences, and seamlessly integrate with navigation systems, positions it as a valuable tool for enhancing road safety and driver awareness.

# **References**

1. How To Design a MySQL Database for your Basic Notification System? - DEV Community. (n.d.). Retrieved from <https://dev.to/codehakase/how-to-design-a-mysql-database-for-your-basic-notification-system-4aic>
2. Designing a notification system | Notification database design | Medium. (n.d.). Retrieved from <https://medium.com/@amrit.kumar/designing-a-notification-system-notification-database-design-8b33b8169027>
3. Designing a Notification System in a Microservices Architecture. (2021, June 23). Retrieved from <https://www.freecodecamp.org/news/designing-a-notification-system-in-a-microservices-architecture/>
4. Notification System Design: Building a Scalable Notification Infrastructure. (2019, August 20). Retrieved from <https://www.callstack.com/blog/notification-system-design-building-a-scalable-notification-infrastructure>
5. Database Design for a Notification System. (n.d.). Retrieved from <https://www.codeproject.com/Articles/5291376/Database-Design-for-a-Notification-System>
6. Building a Scalable Notification System. (2018, April 23). Retrieved from <https://blog.bitsrc.io/building-a-scalable-notification-system-7e42c6b65e4d>
7. Designing a Robust Notification System. (2020, December 8). Retrieved from <https://engineering.shopify.com/blogs/engineering/designing-a-robust-notification-system>