REPUBLIQUE DU CAMEROON
PAIX-Travail-Patrie
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR

FACULTE DE L'ENGINERIE ET TECHGNOLOGIE

REPUBLIC OF CAMEROON
Peace-Work-Fatherland
MINISTER OF HIGHER EDUCATION

FACULTY OF ENGINEERING AND TECHNOLOGY

**********UNIVERSITY OF BUEA **********

DEPARTMENT OF COMPUTER ENGINEERING

COURSE: INTERNET PROGRAMMING (J2EE) AND MOBILE PROGRAMMING

COURSE CODE: CEF 440

TASK 6: DATABASE DESIGN AND IMPLEMENTATION

PRESENTED BY

GROUP 20

INSTRUCTOR: DR. VALERY

JUNE 2024

GROUP MEMBERS

S/N	**Name**	Matricule Number
1	DUESENBERRY MBIKANG AGBORTAR AKO	FE21A175
2	FOTABONG FUALEFAC PUISSANCE	FE21A195
3	ABANGMA ARRAH JESUS MCLOUIS	FE21A124
4	TABOT JOEL EBANGHA	FE21A308
5	EBOMESUMBE NGOLE DANDY BRADLEY	FE21A177

Contents

1. Introduction

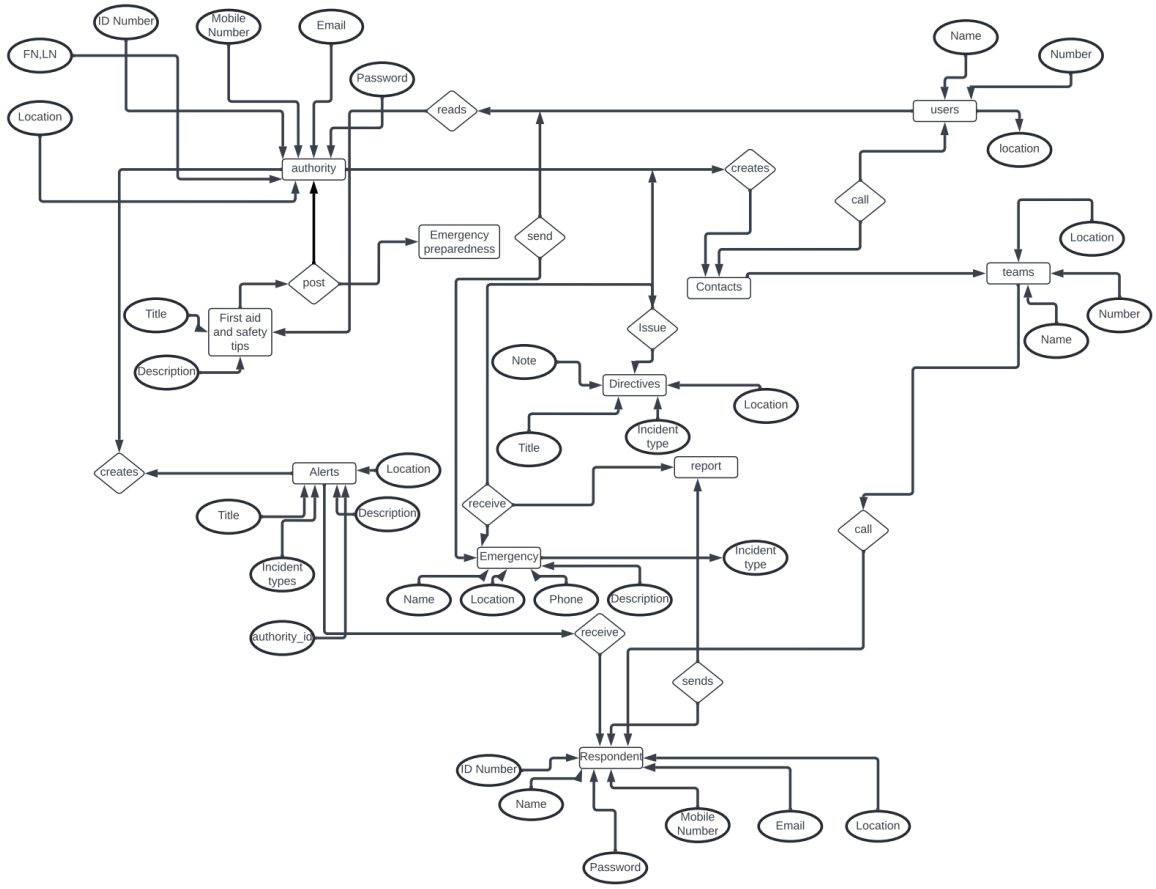
In the face of increasing disaster events, both natural and man-made, our team was tasked with the development of a robust disaster management system aimed at mitigating the impact of disasters through enhanced data coordination and response efficiency. This report outlines the database design and implementation phase, which is foundational to the system's operational success.

2. Conceptual Design

During the conceptual design phase, we developed an Entity-Relationship (ER) Diagram to visualize the data relationships crucial to disaster management:

Entities Identified:

- **Users**: Individuals using the system, including the general public and disaster victims.
- **Authority**: Government and agency officials responsible for disaster response.
- **Respondents**: First responders and emergency services.
- **Emergency**: Types and details of emergencies.
- **Contacts**: Contact details for all users.
- **Alerts, Calls, and Reports**: Communication and data generated during emergencies.



Key Relationships and Cardinalities:

- Authorities issue directives to Respondents.
- Users send and receive Alerts and make Calls.
- Reports are generated by Respondents and sent to Authorities.

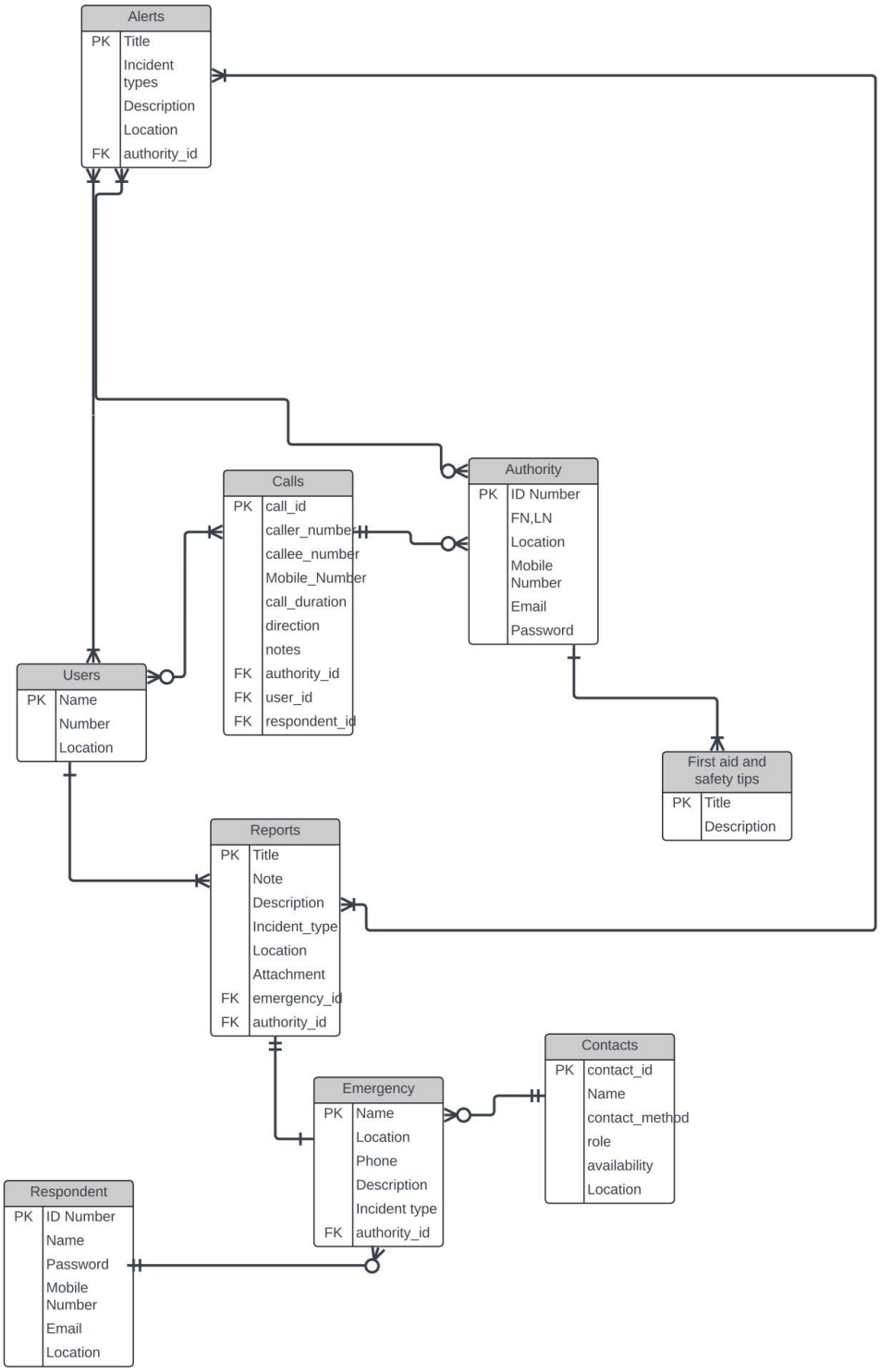
3. Logical Design

In this phase, we translated our conceptual ER diagram into a logical model, normalizing data to reduce redundancy and improve integrity:

Schema Definition:

- Created tables like Users, Authorities, Emergencies, each reflecting the system's entities.
- Defined attributes such as ID Number, Name, Location, and Email, using data types appropriate for the content (e.g., VARCHAR for text, INT for numerical data).

- Established Primary Keys and Foreign Keys to maintain referential integrity across the relational database.



4. Physical Design

We selected MYSQL workbench as our Database Management System (DBMS) due to its robustness, scalability, and support for complex queries necessary for disaster management:

Physical Storage Considerations:

- Implemented indexing on frequently accessed fields like User ID and Emergency Type to speed up queries.
- Employed partitioning on historical data to improve performance and manageability.

Database Security Design:

- Implemented data encryption both at rest and in transit using mySQL workbench native support.
- Setup role-based access controls to ensure that data is only accessible to authorized personnel, crucial for protecting sensitive information.

5. SQL Database Implementation

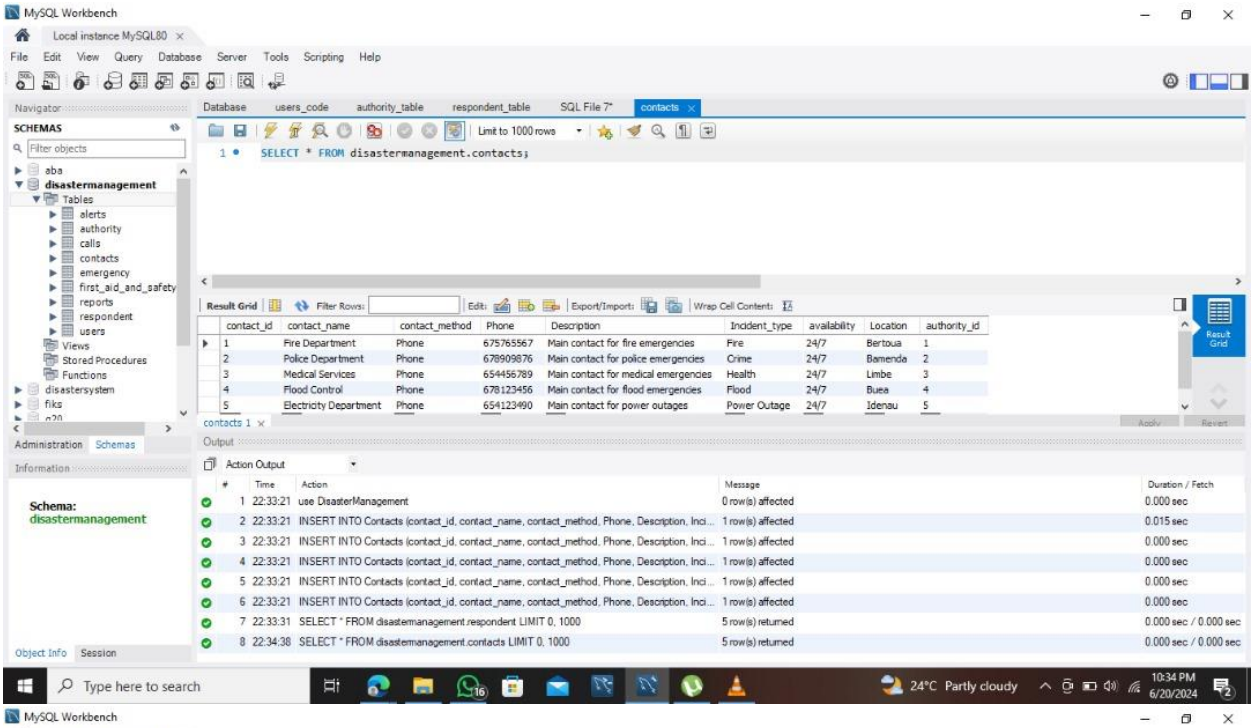
Our SQL implementation phase involved the actual creation of the database schema based on our design:

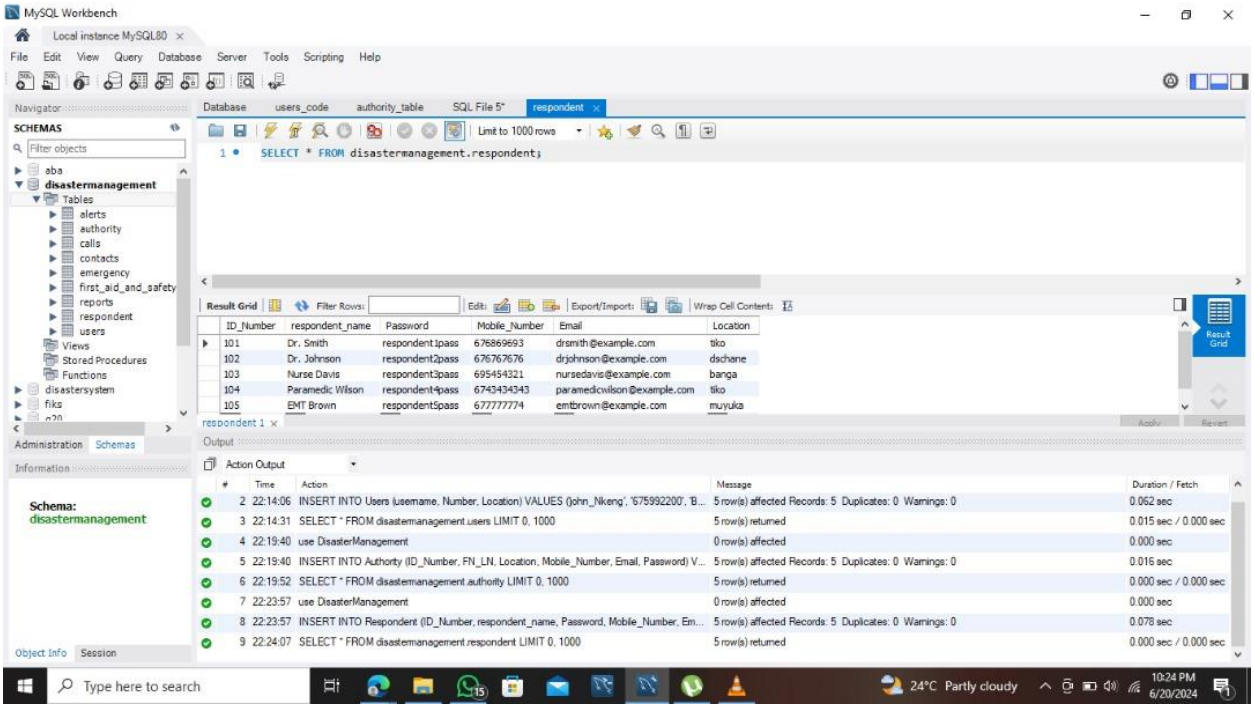
SQL Scripts:

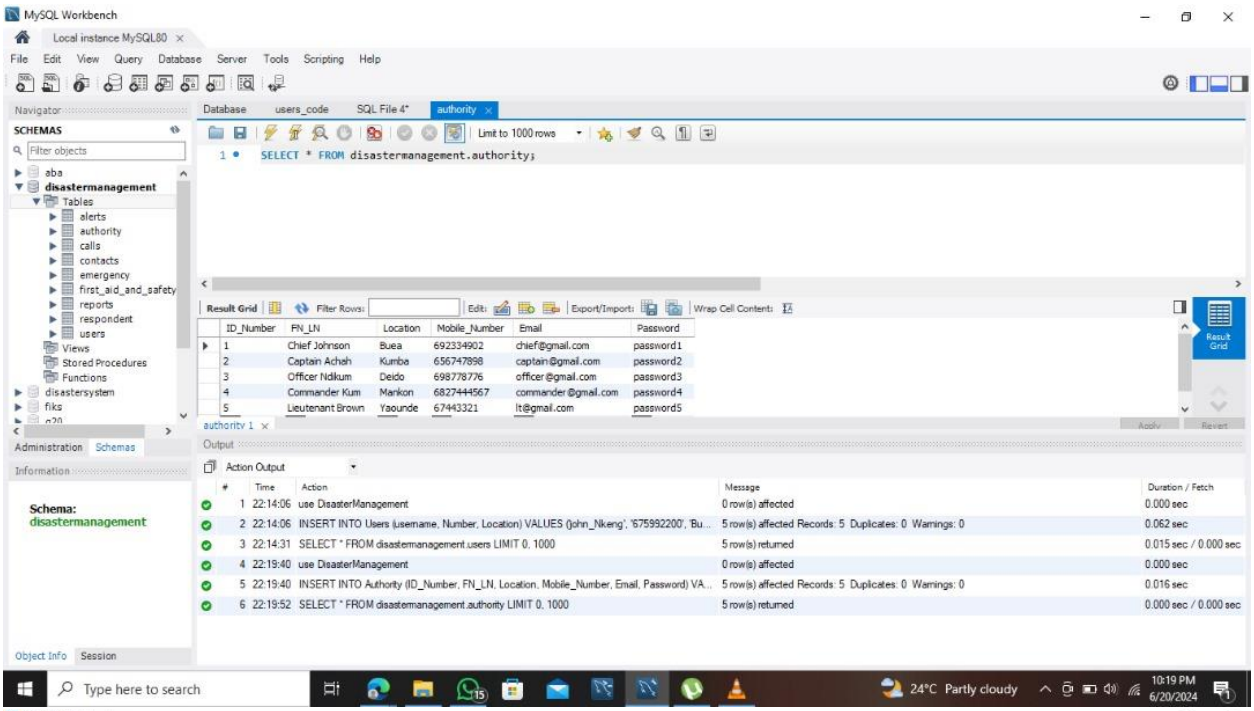
- Executed SQL scripts to create tables, define relationships, and set constraints.
- Example SQL for creating the Users table:
- Scripts can be found here: https://github.com/duesenberry55/CEF440-group-20/tree/main/task%206

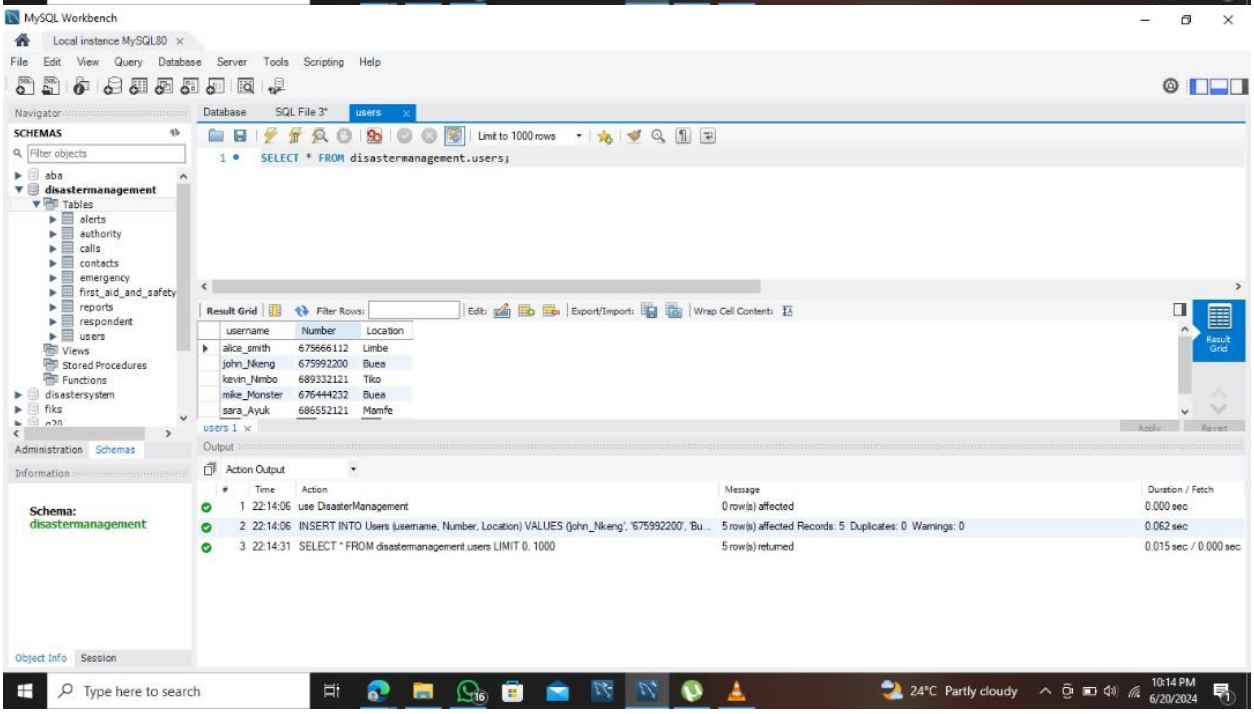
```
sql
CREATE TABLE Users (
  UserID INT PRIMARY KEY,
  Name VARCHAR(100),
  Location VARCHAR(100),
  Email VARCHAR(100)
);
```

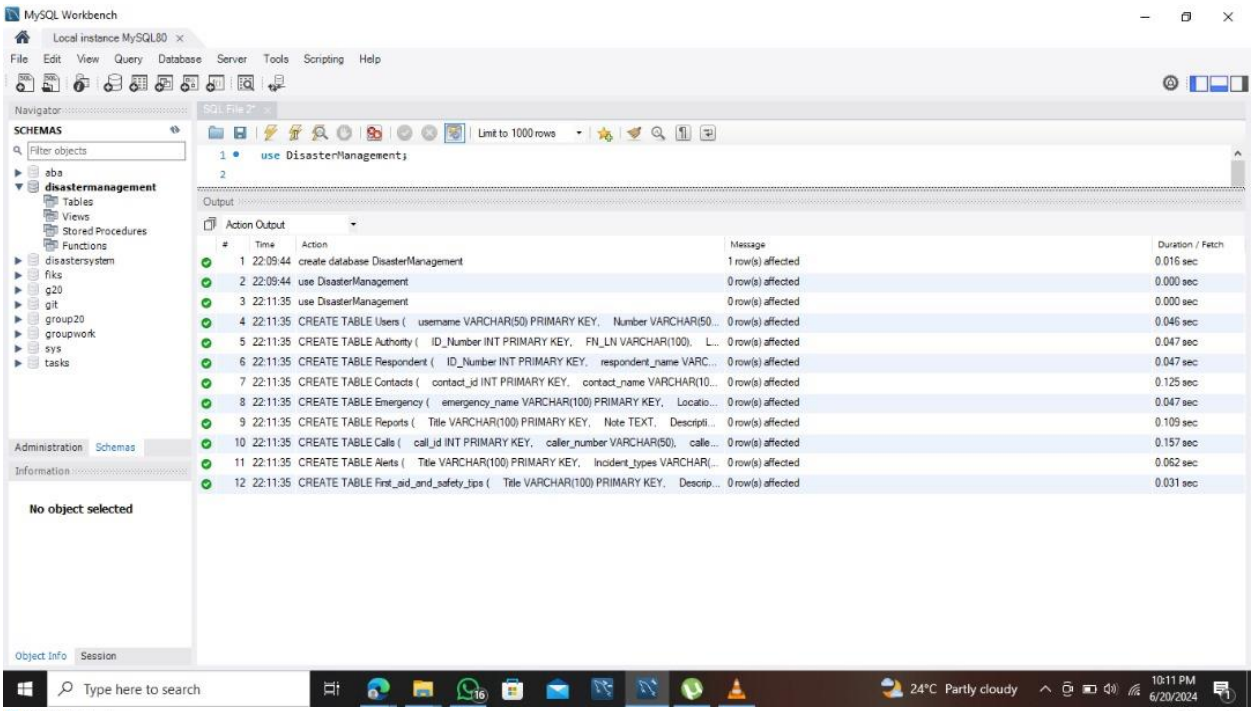
Results:

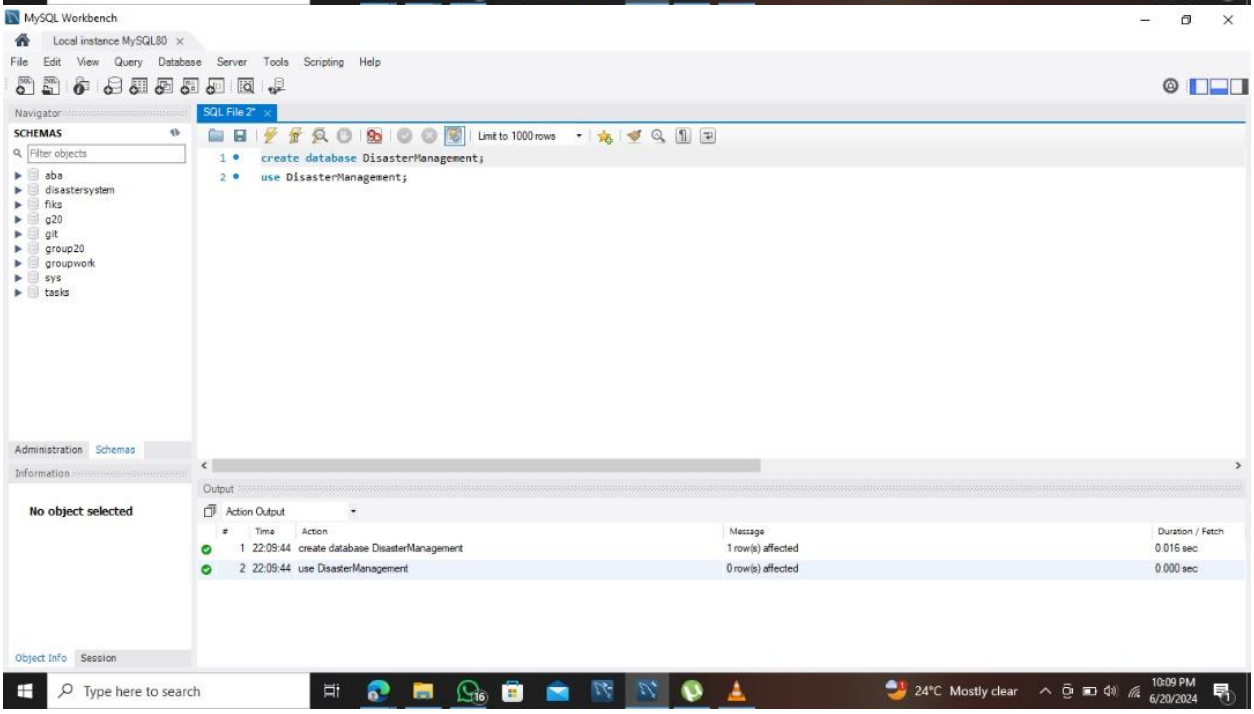

MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
aba
disastermanagement
Tables
alerts
authority
calls
contacts
emergency
first_aid_and_safety
reports
respondent
users
Views
Stored Procedures
Functions
disastersystem
fiks
Administration Schemas
Information
Schema:
disastermanagement
Object Info Session
Database users_code authority_table respondent_table SQL File 7* contacts
Limit to 1000 rows
1 • SELECT * FROM disastermanagement.contacts;
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
contact_id contact_name contact_method Phone Description Incident_type availability Location authority_id
1 Fire Department Phone 675765567 Main contact for fire emergencies Fire 24/7 Bertoua 1
2 Police Department Phone 678909876 Main contact for police emergencies Crime 24/7 Bamenda 2
3 Medical Services Phone 654456789 Main contact for medical emergencies Health 24/7 Limbe 3
4 Flood Control Phone 678123456 Main contact for flood emergencies Flood 24/7 Buea 4
5 Electricity Department Phone 654123490 Main contact for power outages Power Outage 24/7 Idenau 5
contacts 1
Output
Action Output
Time Action Message Duration / Fetch
1 22:33:21 use DisasterManagement 0 row(s) affected 0.000 sec
2 22:33:21 INSERT INTO Contacts (contact_id, contact_name, contact_method, Phone, Description, Inci... 1 row(s) affected 0.015 sec
3 22:33:21 INSERT INTO Contacts (contact_id, contact_name, contact_method, Phone, Description, Inci... 1 row(s) affected 0.000 sec
4 22:33:21 INSERT INTO Contacts (contact_id, contact_name, contact_method, Phone, Description, Inci... 1 row(s) affected 0.000 sec
5 22:33:21 INSERT INTO Contacts (contact_id, contact_name, contact_method, Phone, Description, Inci... 1 row(s) affected 0.000 sec
6 22:33:21 INSERT INTO Contacts (contact_id, contact_name, contact_method, Phone, Description, Inci... 1 row(s) affected 0.000 sec
7 22:33:31 SELECT * FROM disastermanagement.respondent LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
8 22:34:38 SELECT * FROM disastermanagement.contacts LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
Type here to search
24°C Partly cloudy
10:34 PM
6/20/2024


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
aba
disastermanagement
Tables
alerts
authority
calls
contacts
emergency
first_aid_and_safety
reports
respondent
users
Views
Stored Procedures
Functions
disastersystem
fiks
Administration Schemas
Information
Schema:
disastermanagement
Object Info Session
Database users_code authority_table SQL File 5* respondent
Limit to 1000 rows
1 • SELECT * FROM disastermanagement.respondent;
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
ID_Number respondent_name Password Mobile_Number Email Location
101 Dr. Smith respondent1pass 676869693 drsmith@example.com tiko
102 Dr. Johnson respondent2pass 676767676 drjohnson@example.com dschane
103 Nurse Davis respondent3pass 695454321 nursedavis@example.com banga
104 Paramedic Wilson respondent4pass 6743434343 paramedicwilson@example.com tiko
105 EMT Brown respondent5pass 677777774 emtbrown@example.com muyuka
respondent 1
Output
Action Output
Time Action Message Duration / Fetch
2 22:14:06 INSERT INTO Users (username, Number, Location) VALUES ('John_Nkeng', '675992200', 'B... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0 0.062 sec
3 22:14:31 SELECT * FROM disastermanagement.users LIMIT 0, 1000 5 row(s) returned 0.015 sec / 0.000 sec
4 22:19:40 use DisasterManagement 0 row(s) affected 0.000 sec
5 22:19:40 INSERT INTO Authority (ID_Number, FN_LN, Location, Mobile_Number, Email, Password) V... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0 0.016 sec
6 22:19:52 SELECT * FROM disastermanagement.authority LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
7 22:23:57 use DisasterManagement 0 row(s) affected 0.000 sec
8 22:23:57 INSERT INTO Respondent (ID_Number, respondent_name, Password, Mobile_Number, Em... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0 0.078 sec
9 22:24:07 SELECT * FROM disastermanagement.respondent LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
Type here to search
24°C Partly cloudy
10:24 PM
6/20/2024


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
aba
disastermanagement
Tables
alerts
authority
calls
contacts
emergency
first_aid_and_safety
reports
respondent
users
Views
Stored Procedures
Functions
disastersystem
fiks
Administration Schemas
Information
Schema:
disastermanagement
Object Info Session
Database users_code SQL File 4* authority
Limit to 1000 rows
1 • SELECT * FROM disastermanagement.authority;
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
ID_Number FN_LN Location Mobile_Number Email Password
1 Chief Johnson Buea 692334902 chief@gmail.com password1
2 Captain Achah Kumba 656747898 captain@gmail.com password2
3 Officer Ndikum Deido 698778776 officer@gmail.com password3
4 Commander Kum Mankon 6827444567 commander@gmail.com password4
5 Lieutenant Brown Yaounde 67443321 lt@gmail.com password5
authority 1
Output
Action Output
Time Action Message Duration / Fetch
1 22:14:06 use DisasterManagement 0 row(s) affected 0.000 sec
2 22:14:06 INSERT INTO Users (username, Number, Location) VALUES ('john_Nkeng', '675992200', 'Bu... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0 0.062 sec
3 22:14:31 SELECT * FROM disastermanagement.users LIMIT 0, 1000 5 row(s) returned 0.015 sec / 0.000 sec
4 22:19:40 use DisasterManagement 0 row(s) affected 0.000 sec
5 22:19:40 INSERT INTO Authority (ID_Number, FN_LN, Location, Mobile_Number, Email, Password) VA... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0 0.016 sec
6 22:19:52 SELECT * FROM disastermanagement.authority LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
Type here to search
24°C Partly cloudy
10:19 PM
6/20/2024


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
aba
disastermanagement
Tables
alerts
authority
calls
contacts
emergency
first_aid_and_safety
reports
respondent
users
Views
Stored Procedures
Functions
disastersystem
fiks
Administration Schemas
Information
Schema:
disastermanagement
Object Info Session
Database SQL File 3* users
Limit to 1000 rows
1 • SELECT * FROM disastermanagement.users;
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
username Number Location
alice_smith 675666112 Limbe
john_Nkeng 675992200 Buea
kevin_Nimbo 689332121 Tiko
mike_Monster 676444232 Buea
sara_Ayuk 686552121 Mamfe
users 1
Output
Action Output
Time Action Message Duration / Fetch
1 22:14:06 use DisasterManagement 0 row(s) affected 0.000 sec
2 22:14:06 INSERT INTO Users (username, Number, Location) VALUES ('john_Nkeng', '675992200', 'Bu... 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0 0.062 sec
3 22:14:31 SELECT * FROM disastermanagement.users LIMIT 0, 1000 5 row(s) returned 0.015 sec / 0.000 sec
Type here to search
24°C Partly cloudy
10:14 PM
6/20/2024





6. Technologies Used

- **Database Management System (DBMS):** mySQl was chosen for its advanced features, reliability, and open-source nature.
- **Development Environment:** We utilized mySQL workbench, providing us with a graphical interface to facilitate database design and manipulation.

- **Data Modeling Tools:** Lucid chart was used to create detailed ER diagrams, supporting our data modeling and schema design processes.
- **Security Technologies:** We implemented AES encryption standards through PostgreSQL for data at rest and TLS for data in transit.
- **Testing Tools:** For load testing, we used mySQL workbench built-in benchmarking tool, to measure the system's performance under various load conditions.

7. Testing and Validation

Extensive testing was conducted to ensure the integrity and performance of the database:

Testing Strategies:

- Performed unit testing on individual tables.
- Conducted integration testing to ensure that table relationships were properly enforced.
- Load testing was implemented to simulate real-world usage scenarios.

8. Importance of Database Management in the Disaster Management System

The structured organization of our database has significantly improved data retrieval times and accuracy, which is critical during ongoing disaster events. The system supports dynamic reporting and analysis, enabling decision-makers to quickly understand disaster impacts and coordinate appropriate responses. Additionally, the database is designed to scale, accommodating an increasing volume of disaster data and evolving management requirements.

9. Conclusion

The database design and implementation phase has successfully laid down the backbone for the disaster management system. Our choices in technology and design principles have created a scalable, secure, and robust database that meets the critical demands of disaster management.

10. Future Considerations

Moving forward, we will continue to enhance security measures, refine our backup and recovery procedures, and explore integration opportunities with other emergency response systems to further extend the capabilities of our disaster management system.