A picture containing text

Description automatically generated

**Fayoum University**

**Faculty of Science**

**Mathematics department**

**Mathematics and Computer Science**

**City Mall**

**(Application&&Security System)**

A graduation project document submitted to the Dep. of Computer Science as  
partial fulfillment for the Requirement for the Degree of Bachelor in  
Computer Science

ByAhmed Hamada Makhouf  
Amr Khaled  
Mohammed Ashraf  
Mustafa Hamdy

Supervised byDr. **Mohamed Abd-Elbaky**  
6/20

1. Introduction

- Briefly explain the purpose of the security system and its importance in a mall environment.

- Provide an overview of the backend architecture and its role in supporting the system.

2. Technologies Used

- Explain that the backend is built using Node.js, Kotlin, PostgreSQL, and Sequelize.

- Describe the reasons for choosing these technologies and their benefits for the project.

3. Database Design

- Describe the database schema and the tables required for the system.

- Explain the purpose and relationships of each table.

- Provide the table structures, including the column names, data types, and any constraints or indexes.

4. Authentication and User Management

- Explain how user authentication and authorization are implemented in the system.

- Describe the login and registration processes for users.

- Discuss the security measures, such as password hashing and encryption, used to protect user information.

5. Employee Attendance

- Explain how the system tracks employee attendance.

- Describe the process of recording employee check-ins and check-outs.

- Discuss any additional features related to employee attendance, such as leave management.

6. Fire Detection System

- Explain how the AI models are used to detect fires in the mall.

- Describe the data sources and sensors used to monitor for fire incidents.

- Discuss how the backend processes and analyzes the data received from the sensors to identify potential fires.

7. Car Tracking System

- Explain how the AI models are utilized to track cars entering and leaving the mall.

- Describe the data sources and sensors used for car tracking.

- Discuss how the backend processes and analyzes the data received from the sensors to identify incoming and outgoing cars.

8. Lost People Detection

- Explain how the AI models are employed to detect and find lost people in the mall.

- Describe the data sources and sensors used for tracking people.

- Discuss how the backend processes and analyzes the data received from the sensors to identify lost individuals.

9. Cinema Ticket Purchases

- Explain how users can purchase cinema tickets through the application.

- Describe the process of selecting movie shows, seat reservation, and payment handling.

- Discuss any integration with third-party payment gateways or services.

10. Conclusion

- Summarize the role of the backend in the overall security system for the mall.

- Highlight any notable challenges or achievements during the development process.

3. Database Design

The backend of the security system for the mall utilizes a PostgreSQL database managed using Sequelize. The following tables have been designed to support the functionalities of the system:

3.1. Car

The `cars` table stores information about the cars entering and leaving the mall. It includes columns such as `plateNum` (primary key) and `color`.

3.2. Cinema

The `cinemas` table stores details about the cinemas within the mall. It includes columns such as `id` (primary key), `name`, `location`, `openAt`, `closeAt`, `phone`, and `imageUrl`.

3.3. EmployeeAttendance

The `employeeAttendance` table tracks the attendance of the mall employees. It includes columns such as `id` (primary key), `loggedIn`, and `loggedOut`.

3.4. Employee

The `employees` table stores information about the mall employees. It includes columns such as `id` (primary key), `name`, `email`, `password`, `imageUrl`, and `role`.

3.5. Issue

The `issues` table records the issues reported within the mall. It includes columns such as `id` (primary key), `type`, `details`, `imageUrl`, and `state`.

3.6. ModelIssue

The `modelIssues` table stores information about the model issues reported within the mall. It includes columns such as `id` (primary key), `type`, `details`, `imageUrl`, and `state`.

3.7. Movie

The `movies` table stores information about the movies being shown in the mall's cinema. It includes columns such as `id` (primary key), `name`, `duration`, `release`, `description`, `genre`, `time`, `ticketPrice`, and `imageUrl`.

3.8. Offer

The `offers` table stores details about the offers available at various shops. It includes columns such as `id` (primary key), `discount`, `startAt`, and `endAt`.

3.9. Shop

The `shops` table stores information about the shops within the mall. It includes columns such as `id` (primary key), `name`, `location`, `openAt`, `closeAt`, `phone`, `imageUrl`, and `shopType`.

3.10. User

The `users` table stores information about the system users, including customers. It includes columns such as `id` (primary key), `name`, `phone`, `email`, `password`, `passwordConfirm`, `passwordChangedAt`, `passwordResetCode`, and `passwordResetExpire`.

3.11. Visit

The `visits` table tracks the visits of cars to the mall. It includes columns such as `id` (primary key), `timeIn`, `timeOut`, `section`, and `cost`.

3.12. Checkout

The `checkouts` table represents the checkouts made by users for movies. It includes columns such as `id` (primary key), `status`, `ticketPrice`, `ticketNum`, and `cost`.

3.13. CinemaMovie

The `cinema\_movie` table represents the relationship between cinemas and movies, indicating which movies are being shown in each cinema. It does not include any additional columns.

3.14. IssueEmployee

The `issue\_employee` table represents the relationship between issues and employees, indicating which employees are assigned to each issue. It does not include any additional columns.

3.15. ModelIssueEmployee

The `modelIssue\_employee` table represents the relationship between model issues and employees, indicating which employees are assigned to each model issue. It does not include any additional columns.

3.16. OfferShop

The `offer\_shop` table represents the relationship between offers and shops, indicating which shops have specific offers. It does not include any additional columns.

3.17. UserShop

The `user\_shop` table represents the relationship between users and shops, indicating which shops are visited by each user. It does not include any additional columns.

Note: The table definitions provided above assume the use of Sequelize models. Additional columns, relationships, or modifications may exist based on specific project requirements.

Please note that the code snippets provided may not reflect the complete implementation of the security system's database and might require further customization to suit your specific needs.