UNIVERSITY OF GHANA

ACCRA, GHANA

SCHOOL OF ENGINEERING SCIENCES

DEPARTMENT OF COMPUTER ENGINEERING

CPEN 208 COURSE PROJECT 1 REPORT

NAME: SENA DELASE ANYOMI

STUDENT ID: 11292620

TITLE: A Report on the Development Task for the Computer Engineering Department Software

Introduction:

The objective of this task was to develop a relational database for the Computer Engineering Department and a web application using Next.js 14. The software aims to provide functionalities for managing student information, fee payments, course enrollments, and lecture assignments.

The task was divided into two main parts:

1. Database Development: Creating a PostgreSQL database with necessary schemas, tables, insert scripts, and a function to calculate outstanding fees.

2. Web Application Development: Building a Next.js 14 application with login, registration, and dashboard functionalities using Tailwind CSS templates.

Part 1: Database Development

1. Creating the Database and Schemas

- Database Name: `cpen_dept`

- Schemas Created: `public` (default schema)

`cpen` (additional schema)

2. Creating Tables

The following tables were created to implement the required functionalities:

- Students Table
  - Entities and Attributes:
  - `student_id`: Unique identifier for each student
  - `first_name`: First name of the student
  - `last_name`: Last name of the student
  - `date_of_birth`: Date of birth of the student
  - `email`: Email address of the student (unique)

- Fees Table
  - Entities and Attributes:
  - `fee_id`: Unique identifier for each fee record
  - `student_id`: Identifier linking to the student
  - `amount`: Amount of the fee
  - `payment_date`: Date when the fee was paid
  - `payment_method`: The mode in which the fee was sent

- Courses Table
  - Entities and Attributes:
  - `course_id`: Unique identifier for each course
  - `course_name`: Name of the course
  - `course_code`: Code of the course

- Enrollments Table
  - Entities and Attributes:
  - `enrollment_id`: Unique identifier for each enrollment record
  - `student_id`: Identifier linking to the student
  - `enrollment_date`: Date of enrollment

- Lecturers Table
    - Entities and Attributes:
    - `lecturer_id`: Unique identifier for each lecturer
    - `first_name`: First name of the lecturer
    - `last_name`: Last name of the lecturer
    - `email`: Email address of the lecturer

- Lecturer Assignments Table
    - Entities and Attributes:
    - `lecturer_assignment_id`: Unique identifier for each lecture assignment
    - `lecturer_id`: Identifier linking to the lecturer
    - `course_id`: Identifier linking to the course

- TAs Table
    - Entities and Attributes:
    - `ta_id`: Unique identifier for each TA
    - `first_name`: First name of the TA
    - `last_name`: Last name of the TA
    - `email`: Email address of the TA

- TA Assignment to Lecturer Table
    - Entities and Attributes:
    - `ta_assignment_id`: Unique identifier for each TA assignment
    - `lecturer_id`: Identifier linking to the lecturer
    - `course_id`: Identifier linking to the course

3. Insert Scripts for Sample Data

Sample data was inserted into each table to provide realistic data for testing and development purposes. The data included:

- Multiple students with various personal information.

- Fee records indicating student fees.

- Several courses with different credits.

- Enrollments linking students to their courses.

- Lecturers with their departmental affiliations.

- Assignments of lecturers to courses and TAs to courses.

4. Creating a Function to Calculate Outstanding Fees

A database function was created to calculate the outstanding fees for each student. The function aggregates the fee information and returns the result in a JSON array format, making it easy to integrate with web applications.

Part 2: Web Application Development

1. Project Initialization

The Next.js project was initialized using the `create-next-app` command, setting up the basic structure for the web application.

2. Tailwind CSS Integration

Tailwind CSS was integrated for styling the application. This involved:

- Installing Tailwind CSS and its dependencies.

- Configuring Tailwind to purge unused styles in production.

- Including Tailwind directives in the global CSS file.

3. Creating Templates and Modifying for the Project

Tailwind CSS templates were used and modified to create the following components:

- Login Page: A form for users to log into the application with fields for email and password.

- Register Page: A registration form for new users to sign up with fields for personal information.

- Dashboard: A user dashboard displaying relevant information and providing access to various functionalities of the application.

4. Project Submission

The complete project, including the Next.js application source code, database scripts, and a database backup, was pushed to a GitHub repository. The repository URL was shared for project review and assessment.

Conclusion:

This report details the steps taken to develop a relational database and a web application for the Computer Engineering Department. The project involved creating a PostgreSQL database with necessary tables and functions, integrating Tailwind CSS with a Next.js application, and ensuring the code is version controlled via GitHub. The resulting software provides essential functionalities for managing student information, fee payments, course enrollments, and lecture assignments.