

MAKERERE UNIVERSITY BUSINESS SCHOOL
COURSE WORK TWO EXAM FOR BACHELOR OF BUSINESS COMPUTING
OF MAKERERE UNIVERSITY, ACADEMIC YEAR 2024/2025

COURSE NAME: ADVANCED WEB APPLICATION DEVELOPMENT

COURSE CODE: BUC3131

SEMESTER: ONE

YEAR OF STUDY: III

GROUP COURSEWORK TWO

GROUP MEMBERS

NO.	NAME	REGISTRATION NUMBER	STUDENT NUMBER
1.	JJUUKO RONALD	22/U/5997	2200705997
2.	MUSINGUZI SIMON PETER	22/U/20505/EVE	2200720505

Instructions for use for Elites Academy Student Management System

1. Project Overview

The Student Management System is a web-based application built with the Laravel framework. It is designed to help administrators efficiently manage student records, courses, enrollments, and generate reports. This project supports CRUD operations (Create, Read, Update, Delete) for students, courses, and enrollments. It is built with user-friendly features to streamline administrative tasks.

2. Requirements

Before you begin setting up the project, ensure that the following software is installed on your system:

- **PHP** (Version 7.3 or above)
- **Composer**
- **Laravel Framework**
- **MySQL** (or compatible database server)

3. Project Setup

Step 1: Clone the Project

Clone the repository to your local machine using From:

<https://github.com/jjronnie/ronnieftcymo>

Step 2: Install Dependencies

Navigate into the project directory:

Install all the project dependencies

Step 3: Set Up the Environment

Copy the ``.env.example`` file to ``.env``

Configure your database credentials in the ``.env`` file:

`DB_CONNECTION=mysql`

`DB_HOST=127.0.0.1`

`DB_PORT=3306`

`DB_DATABASE=elite_db`

`DB_USERNAME=root`

`DB_PASSWORD=`

Step 4: Generate Application Key

Generate the application key by running:

`php artisan key:generate`

Step 5: Run Migrations

Run the database migrations to create the necessary tables:

`php artisan migrate`

Seed the database with sample data:

`php artisan db:seed`

Step 6: Start the Development Server

Start the Laravel development server by running:

```
php artisan serve
```

4. Project Structure

The structure of the project is as follows:

- **app/**: Contains core application logic, including controllers, models, and database migrations.
- **database/**: Contains migrations and seeders for the database.
- **resources/**: Includes views (Blade templates) and language files.
- **routes/**: Defines all routes for the application (web and API).
- **public/**: Contains publicly accessible assets such as images, stylesheets, and JavaScript files.
- **.env**: Environment configuration file for database and application settings.

5. Features and Functionality

- **Admin Login**

Admins can log in using their credentials (stored in the database).

- **Student Management**

Admins can:

- View a list of all students.
- Add, edit, or delete student records.

- **Course Management**

Admins can:

- View a list of all courses.
- Add, edit, or delete courses.

- **Enrollment Management**

Admins can:

- Enroll students in courses.
- View and manage student enrollments.
- **Report Generation**

Admins can generate reports based on:

- Students Per Level
- Student enrollments.
- Students Registered per month
- Detailed student records.

6. Usage Instructions

- **Logging In:**

- Use the admin login page to enter credentials. Modify the database to create a new admin user if necessary.

- **Managing Students:**

· After logging in, go to the **Students** section from the sidebar.

- To add a new student, click **Add New Student** and fill in the required details.
- To view or update student details, click on a student from the list.
- To delete a student, click **View** then you will see a delete button.

- **Managing Courses:**

- Go to the **Courses** tab.
- To add a new course, click **Add New Course** and fill in the course details.
- To update, delete or view course details, click on a course from the list.

- **Managing Enrollments:**

- To enroll a student in a course, click **View Student** for that particular student and select Enroll to courses then select the course.

Generating Reports:

- Go to the **student or Dashboard**.
- Choose the type of report you wish to generate (e.g., student enrollment statistics or student list by Level).
- Click **Generate Report** to view or download the report.

7. Database Tables

Below are the key tables in the Student Management System:

- **students:** Stores student information such as name, phone, email, and date of birth.
- **courses:** Stores course details, including name, description, and credits hours.
- **enrollments:** Tracks student enrollments in courses, with foreign keys referencing students and courses.
- **admins:** Stores admin user credentials for authentication.

8. Future Improvements

- **Enhanced Reporting:** Allow for more advanced filtering of reports (e.g., by date range, course).
- **Student Profile Page:** Enable students to view and update their profiles.
- **Email Notifications:** Integrate email notifications for enrollment and other significant actions.

9. Troubleshooting

Issue: Database connection errors

- **Solution:** Ensure that your .env file contains the correct database credentials and that MySQL is running.

Issue: 404 Error on accessing routes

Solution: Clear the route cache by running:

- php artisan route:clear
- php artisan cache:clear

10. Conclusion

The **Elite Student Management System** streamlines the management of student records, courses, and enrollments with a user-friendly interface and efficient backend. It's built to handle the administrative tasks of any educational institution, and it is ready for future enhancements and scalability.