CSC 554 2.0 Net-centric Computing and System Administration (Intake VI) – Assignment Report

Name - W.M.K.N. Wijekoon

Reg No. - GS/COMP/230

Intake - VI

Project Title: - Student Management App

Links

GitHub - https://github.com/KavinduNuwandika/CSC 554.git

Docker Hub - https://hub.docker.com/r/kavib/student_managment

Docker composer to run with test data -

https://drive.google.com/file/d/13C7r0DuTtaCqbTqchKTbk4Uy8Smc -lj/view?usp=sharing

Project Overview

The goal of this project is to develop a user management and grade tracking system using Spring Boot, where users can register and access different functionalities based on their roles (Admin, Teacher, Student). The system is designed to handle three main user types:

1. Admin:

 The admin has overall control of the system. They can manage users, assign roles (Admin, Teacher, Student), Enroll Students to classes and oversee all activities within the application.

2. Teacher:

o Teachers can create classes and assign grades to students.

The grading system allows teachers to input scores for students.

3. Student:

- Students can view their own results. They can track their performance over time, but do not have the ability to modify their own grades or class data.
- Students can also manage their personal account details and view relevant class information.

Core Features

- User Registration and Login: Users can create an account with their personal information (name, email, password), and their default role will be Student
- Role-based Access Control: Different users have different permissions.
- Class Management: Teachers can create classes. Admins can assign/reassign students to classes.
- Grading System: Teachers can input marks for students in their classes.
- **Grade Viewing:** Students can view their Marks and performance history within the system.

Technologies Used

Backend:

- **Spring Boot:** A powerful framework for building Java-based applications. It simplifies the development of RESTful web services and provides robust support for authentication and authorization, making it ideal for managing different user roles (Admin, Teacher, Student).
- JPA (Java Persistence API): Used for database interaction, allowing seamless integration with MySQL to store and retrieve data related to users, classes, and grades.

Frontend:

• **HTML:** A simple, lightweight approach for presenting the application's user interface. Basic forms and pages will allow users to interact with the system (e.g., user registration, login, viewing grades).

Database:

• MySQL: A relational database system used to store user information (Admin, Teacher, Student), class data, grades, and other relevant information. The system uses JPA to interact with MySQL, ensuring easy CRUD operations.

Version Control:

• **GitHub:** GitHub is used for version control, allowing for easy code management, and maintaining a history of changes. Regular commits and meaningful comments will ensure smooth progress tracking and enable you to revert to previous versions if necessary.

Deployment:

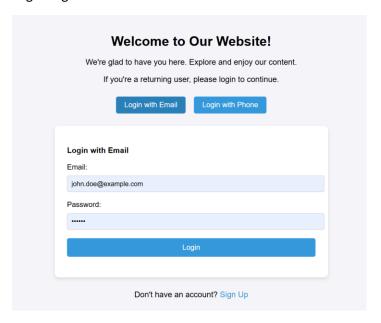
Docker: Docker will be used to containerize the application, ensuring that it can be easily
deployed across various environments. It simplifies the process of packaging the
application along with all necessary dependencies, allowing for consistent deployment on
different machines or cloud services.

Testing:

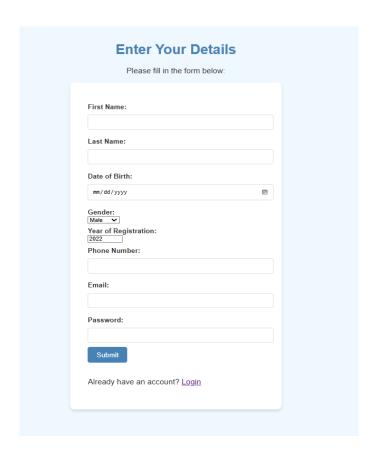
Postman: Postman is a powerful tool used for testing and interacting with the RESTful APIs
of the application. It allows developers to easily send HTTP requests, check responses, and
automate tests to ensure that the application's backend behaves as expected. Postman is
particularly useful for testing CRUD operations, ensuring proper authentication, and
validating error handling.

System Demonstration

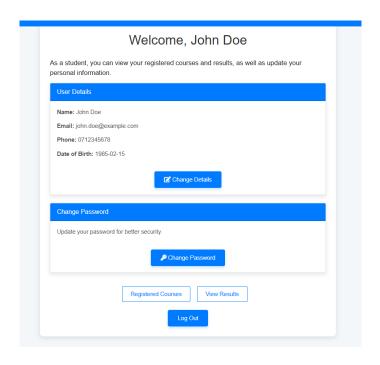
Login Page:



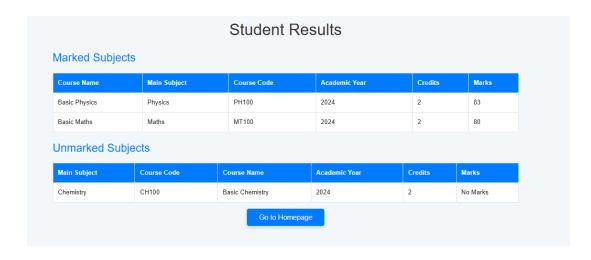
Sign Up:



Student View:

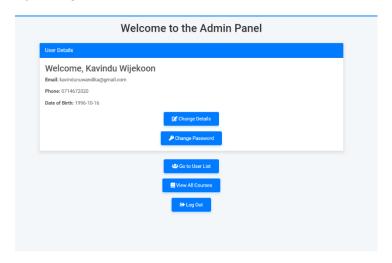


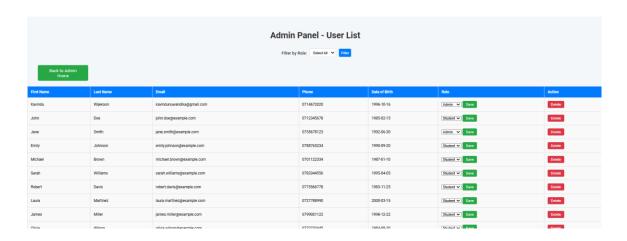


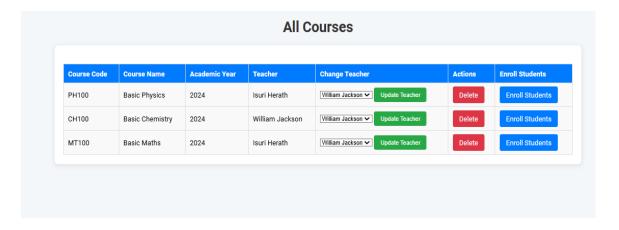


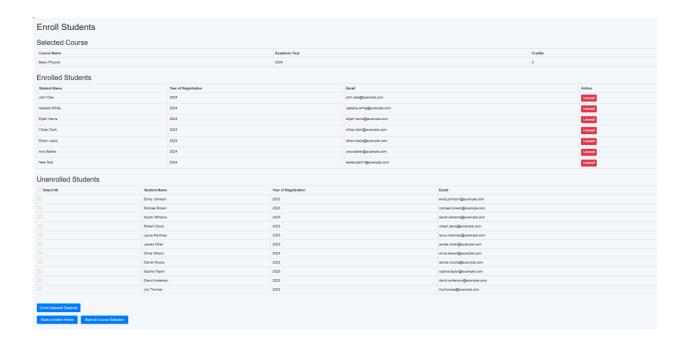
Teacher View:

Admin View:







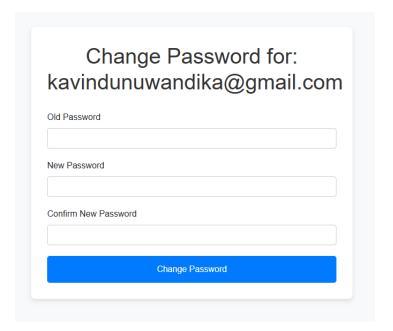


Change details:

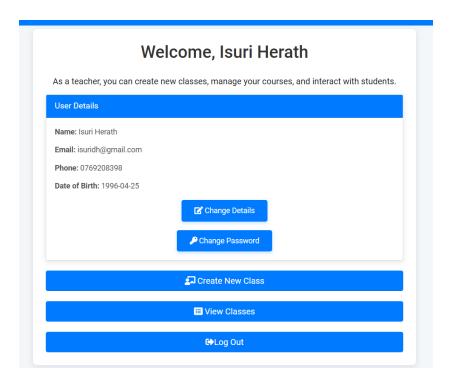
Edit User



Change Password:



Teacher View:



Create a New Class

lain Subject
Physics
lass Code
ourse Name
cademic Year
redits
Create Class

MY CLASSES

#	Course Code	Course Name	Academic Year	Credits	Main Subject	Grade Class
1	PH100	Basic Physics	2024	2	Physics	Grade
2	MT100	Basic Maths	2024	2	Maths	Grade

Go to Teacher Home

Create New Class

Selected Course

Course Name	Academic Year	Credits
Basic Physics	2024	2

Marked Students

Student Name	Year of Registration	Email	Marks	Grade
Isabella White	2024	isabella.white@example.com	60	С
Elijah Harris	2024	elijah harris@example.com	60	С
Chice Clark	2024	chioe.clark@example.com	60	С
Ethan Lewis	2024	ethan.lewis@example.com	70	В
Ava Walker	2024	ava.walker@example.com	80	A
Johnyyy Doeet	2024	john.doe@example.com	83	A



Cott Marks Delete Marks Unmarked Students

Student Name	Year of Registration	Email	Marks	Grade
New Test	2024	teststudent1@example.com	No Marks	No Grade

