

# Innovation and Extra Functionality

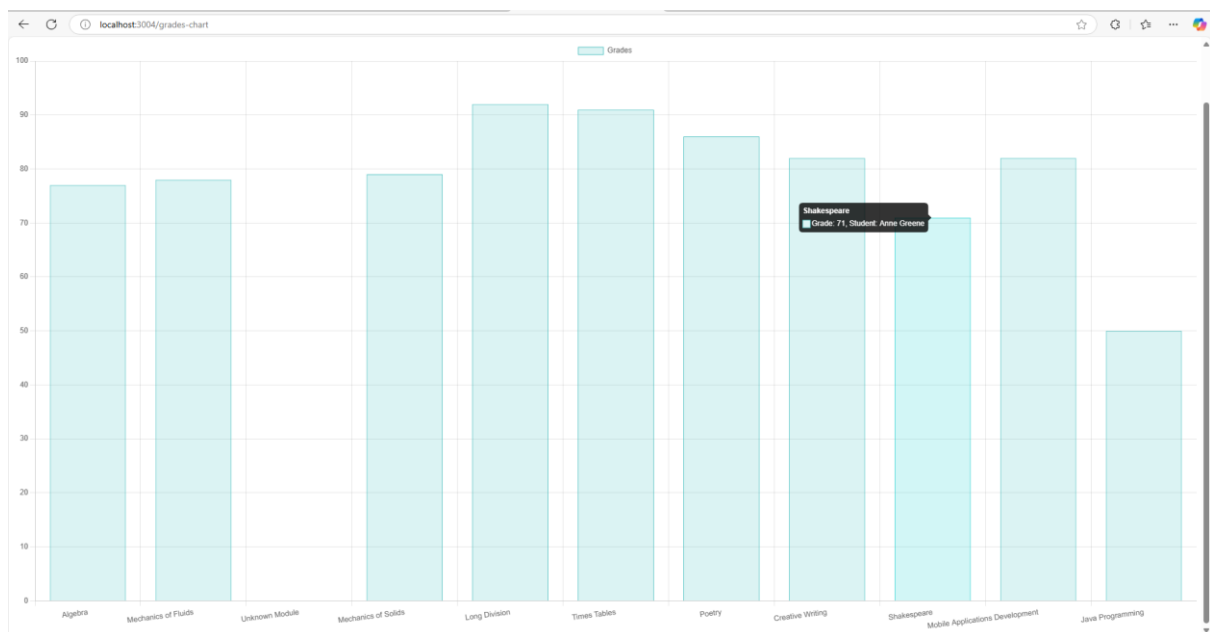
G00425067 – TiffanyNgikChee Yong

Feature1: Grade Chart Analysis .....	1
Feature2: better user experience for "Add New Student" and "Update Student" forms...	2
Feature3: Search Functionality for Students .....	3

This document highlights the innovative features and functionalities that go beyond the requirements of the assignment. The innovative part of the project include:

## Feature1: Grade Chart Analysis

- This feature utilizes the Chart.js library to create an interactive bar chart



Features of the Grade Chart:

- The system processes data to find and display the highest grades for each module, showing the student's name and their respective grade.

So, the users can understand grade distributions quickly.

Code Reference:

- The chart data is dynamically generated by the `/grades/chart-data` endpoint.
- The Chart.js library is used in the `/grades-chart` route for rendering the bar chart on the frontend.

## Feature2: better user experience for "Add New Student" and "Update Student" forms

The "Add New Student" and "Update Student" forms have been significantly improved to provide a better user experience **compared to the default implementation described in the problem sheet.**

Code Reference:

- Input validation logic is implemented in the `/students/add` and `/students/edit/:sid` routes.
- Custom error messages and dynamic form designs are integrated using both server-side and client-side techniques.

## Add Student

Student ID (4 characters):

Name:

Age:



Value must be greater than or equal to 18.

[Back to Students](#)

## Feature3: Search Functionality for Students

A search bar has been added to the "Students" page, enabling users to search for students by their names. This feature makes it easier to find specific students in large datasets and is implemented using **SQL's LIKE operator**.

← ↻ ⓘ localhost:3004/students?search=T

# Students

Search by Name:

[Add Student](#)

[Home](#)

StudentID	Name	Age	Actions
G003	Thomas Murphy	19	<a href="#">Update</a>
G005	Tom Riddle	27	<a href="#">Update</a>
G094	Tiffany Yong	21	<a href="#">Update</a>

[Back to Home](#)

Code Reference:

- The search parameter is passed via the GET method to the '/student' route.
- The DAO function dynamically constructs the query with SELECT \* FROM student WHERE name LIKE?.
- The front-end form submits the query and updates the table view with filtered results.