



## School of Computer Sciences

### CAT405 Intelligent Computing Major Project

#### Final Report

*Automating Feedback Suggestion for Written Assignments  
(SemakLah)*

**Ching Jia Ying**

**153463**

**Supervisor: Dr Jasy Liew Suet Yan**

**Examiner: Associate Professor Dr. Wong Li Pei**

**Academic Session**

**2023/2024**

## DECLARATION

“I declare that the following is my work and does not contain any unacknowledged work from any other sources. This report was undertaken to fulfill the requirements of the Undergraduate Major Project for the Bachelor of Science in Computer Science (Honors) program at Universiti Sains Malaysia”.

Signature : 

Name : Ching Jia Ying

Date : 28/05/2024

## ABSTRAK

Usaha mengajar pelajar membuat tugas bertulis teknikal boleh menjadi tugas yang meletihkan terutamanya untuk kelas yang mempunyai ramai pelajar. Ramai pensyarah mungkin memilih untuk memberikan markah akhir tanpa sebarang maklum balas, manakala yang memberikan maklum balas mungkin tidak konsisten disebabkan bilangan besar tugas yang dihantar. Sebaliknya, pelajar sukar untuk memperbaiki tugas mereka apabila maklum balas tidak diberikan. Oleh itu, projek ini bertujuan untuk mencipta aplikasi web yang dapat menjalankan penilaian separa automatik untuk membantu instruktor dalam proses penilaian. Objektif sistem ini adalah untuk menyediakan bilik darjah dalam talian di mana pelajar boleh memuat naik tugas digital mereka sementara instruktor boleh memberikan maklum balas dan menilai tugas pelajar. Semasa proses penilaian, sistem melakukan ekstraksi teks, dan menerapkan pemrosesan bahasa tabii untuk analisis leksikal dan penghasilan cadangan maklum balas berdasarkan maklum balas sejarah yang diberikan oleh instruktor dengan pemadanan kemiripan teks. Sistem ini juga membolehkan instruktor memilih untuk mengedit maklum balas dan pelajar untuk melihat maklum balas tersebut. Hasil yang dijangka daripada projek ini adalah aplikasi web yang boleh menjalankan pengurusan bilik darjah, pengurusan tugas, penghasilan maklum balas, dan penilaian. Penyelesaian yang dicadangkan menawarkan beberapa faedah kepada instruktor dan pelajar, termasuk maklum balas yang tepat dan berkesan, beban kerja yang berkurang bagi instruktor, penilaian yang konsisten dan adil, dan hasil akademik yang lebih baik bagi pelajar.

---

*Kata Kunci: Penilaian Separuh Automatik, Penjanaan Maklum Balas, Pemrosesan Bahasa Tabii, Pemadanan Kemiripan Teks*

## ABSTRACT

It is tedious work for instructors to provide timely feedback on written assignments, particularly for classes with many students. Many instructors will choose to give the final grades without any feedback, while those who choose to give feedback may not be consistent due to the massive numbers of submitted assignments. On the contrary, students could not improve better in their assignments when feedback is not given. Hence, this project aims to create a web application that can perform semi-automatic grading to aid instructor in the grading process. The objectives of this system are to provide an online classroom where students can upload their digital assignments whilst instructors can provide feedback and grade students' assignments. During the grading process, the system performs text extraction, and applies natural language processing for lexical analysis, and the generation of feedback suggestion based on the historical feedback given by instructor using text-similarities matching. The system also allows instructors to choose to edit the feedback and students to view the feedback. The expected outcome of the project is a web application that can perform classroom management, assignment management, feedback generation, and grading. The proposed solution offers several benefits to both instructors and students, including timely and effective feedback, reduced workload for instructors, consistent and fair grading, and improved academic outcomes for students.

---

*Keywords: Semi-automatic Grading, Feedback Generation, Natural Language Processing, Text-Similarities Matching*

## ACKNOWLEDGEMENTS

In completion of SemakLah, I would like to extend my sincere gratitude to all who have supported me throughout the journey of preparing my final year project. “I can do all this through Him who gives me strength”. Praise God that by His grace, I managed to complete my final year project, and thank God for supporting me through different people and different ways.

First and foremost, I would like to thank the School of Computer Sciences for providing me with this invaluable opportunity to learn and grow in the field of computer science.

I extend my deepest appreciation to my FYP advisor, Dr. Jasy Liew Suet Yan, whose unwavering kindness, patience, and encouragement have illuminated my path throughout this journey. Dr. Jasy's mentorship has been nothing short of transformative, shaping not only the outcome of my project but also nurturing my growth as a student and as an individual. Her boundless knowledge and genuine care have been a constant source of inspiration and motivation. I am profoundly grateful for the privilege of being mentored by someone as exceptional as Dr. Jasy, whose influence will continue to resonate within me long after this project concludes.

To my beloved Christian Fellowship friends and my church members, your unwavering support and fervent prayers have been a source of strength and solace, especially during moments of stress and uncertainty. Your companionship has been a true blessing.

I am grateful to my Computer Sciences Friends who have generously shared their knowledge, insights, and encouragement throughout this journey. Your willingness to support and uplift me has made all the difference.

To my family, whose unwavering love and unwavering mental support have been my rock throughout this endeavor, I am deeply thankful. Your belief in me has been a constant source of motivation.

Lastly, I extend my gratitude to OpenAI, particularly ChatGPT and Copilot, whose innovative technologies have made my self-learning journey smoother and more efficient. Your assistance has been invaluable in overcoming various challenges encountered along the way.

## TABLE OF CONTENTS

DECLARATION .....	2
ABSTRAK.....	3
ABSTRACT.....	4
ACKNOWLEDGEMENTS .....	5
TABLE OF CONTENTS .....	6
LIST OF TABLES .....	11
LIST OF FIGURES .....	13
CHAPTER 1: INTRODUCTION .....	18
1.1. Project Background.....	18
1.2. Problem Statement .....	19
1.3. Motivation.....	19
1.4. System Objectives.....	20
1.5. Proposed Solution .....	21
1.5.1. Classroom Management.....	21
1.5.2. Assignment Management.....	21
1.5.3. Feedback Generation .....	22
1.5.4. Grading Module .....	22
1.6. Benefits and Uniqueness.....	23
1.7. SDG Alignment .....	24
1.8. Report Organization.....	24

CHAPTER 2: RELATED WORK .....	25
2.1. Existing Systems.....	25
2.1.1. FeedbackFruits .....	25
2.1.2. Gradescope.....	26
2.1.3. Formative.ai .....	27
2.1.4. CodePost .....	28
2.2. Competitor Analysis .....	29
2.3. Existing Algorithms .....	30
2.3.1. Lexical Analysis for spelling and grammar check.....	30
2.3.2. Text-Matching Solution .....	30
2.3.3. Existing Feedback Generation Method.....	31
CHAPTER 3: SYSTEM REQUIREMENTS AND DESIGN.....	32
3.1. Project Development Status.....	32
3.2. Project Scope .....	32
3.3. System Capabilities.....	33
3.4. System Limitations .....	34
3.5. Project Management .....	35
3.5.1. Work Breakdown Structure (WBS) .....	35
3.5.2. Gantt Chart.....	30
3.5.3. SWOT Analysis .....	31
3.6. Development Methodology .....	32

3.7. User Requirements .....	33
3.7.1. Requirement Gathering .....	33
3.7.2. Functional Requirements .....	34
3.7.3. Non-Functional Requirements .....	36
3.8. System Analysis .....	37
3.8.1. Use Case Diagram.....	37
3.8.2. Use Case Descriptions .....	38
3.8.3. Sequence Diagram .....	56
3.8.4. Entity Relationship Diagram (ERD) .....	64
3.9. Intelligent Components .....	65
3.9.1. Lexical Analysis (Spelling & Grammar Error).....	65
3.9.2. Feedback Suggestion Generation.....	66
3.10. Data Source .....	67
3.11. Technology Deployed .....	68
3.11.1. Hardware .....	68
3.11.2. Software .....	68
3.11.3. Programming Languages/Tools .....	68
3.11.4. Chosen algorithm .....	68
CHAPTER 4: SYSTEM DESIGN & IMPLEMENTATION .....	69
4.1. System Architecture.....	69
4.2. User Interface Design .....	70

4.2.1.	General .....	70
4.2.2.	Instructors .....	71
4.2.3.	Students.....	84
4.3.	Implementation of Intelligent Component.....	90
4.3.1.	Lexical Analysis (Spelling & Grammar Error).....	90
4.3.2.	Feedback Suggestion Generation.....	92
4.4.	Implementation Strategy .....	94
	CHAPTER 5: SYSTEM TESING & EVALUATION .....	95
5.1.	Testing Strategy .....	95
5.1.1.	Unit Testing .....	95
5.1.2.	Integration Testing .....	95
5.1.3.	System Testing.....	96
5.2.	Test Cases .....	97
5.2.1.	TC001: Login Account .....	97
5.2.2.	TC002: Register Account .....	97
5.2.3.	TC003: Manage Classroom .....	97
5.2.4.	TC004: Manage Student Enrolment .....	98
5.2.5.	TC005: Set Assignment Attributes .....	98
5.2.6.	TC006: Manage Grading Rubric .....	98
5.2.7.	TC007: Manage Assignment Submission.....	99
5.2.8.	TC008: Grade Student Assignment .....	99

5.3. Test Results .....	100
5.3.1. TC001: Login Account .....	100
5.3.2. TC002: Register Account .....	101
5.3.3. TC003: Manage Classroom .....	102
5.3.4. TC004: Manage Student Enrolment .....	103
5.3.5. TC005: Set Assignment Attributes .....	104
5.3.6. TC006: Manage Grading Rubric .....	106
5.3.7. TC007: Manage Assignment Submission.....	107
5.3.8. TC008: Grade Student Assignment .....	108
5.4. Model/Intelligent Component Evaluation.....	110
5.4.1. Lexical Analysis (Spelling & Grammar Error).....	110
5.4.2. Feedback Suggestion Generation.....	119
CHAPTER 6: CONCLUSION & FUTURE WORK .....	149
REFERENCES .....	150

## LIST OF TABLES

Table 2.1 Competitor Analysis .....	29
Table 3.1 Work Breakdown Structure (WBS).....	35
Table 3.2 SWOT Analysis .....	31
Table 3.3 Use Case Description for UC001 Login Account .....	38
Table 3.4 Use Case Description for UC002 Register Account.....	39
Table 3.5 Use Case Description for UC003 Manage Classroom.....	40
Table 3.6 Use Case Description for UC004 Manage Student Enrolment.....	41
Table 3.7 Use Case Description for UC005 Manage Assignment.....	42
Table 3.8 Use Case Description for UC006 Set Assignment Attributes .....	43
Table 3.9 Use Case Description for UC007 Manage Grading Rubric .....	44
Table 3.10 Use Case Description for UC008 Manage Student Submissions .....	45
Table 3.11 Use Case Description for UC009 Download Assignment.....	46
Table 3.12 Use Case Description for UC010 Manage Assignment Submission .....	47
Table 3.13 Use Case Description for UC011 View Grade .....	48
Table 3.14 Use Case Description for UC012 Review Feedback .....	49
Table 3.15 Use Case Description for UC013 Provide Feedback.....	50
Table 3.16 Use Case Description for UC014 Generate Feedback Suggestions .....	51
Table 3.17 Use Case Description for UC015 Perform Lexical Analysis .....	52
Table 3.18 Use Case Description for UC016 Grade Student Assignment .....	53
Table 3.19 Use Case Description for UC017 View Dashboard.....	54

Table 3.20 Use Case Description for UC018 Return Graded Assignment.....	55
Table 5.1 Confusion Matrix .....	111
Table 5.2 Spelling error for assignment 8.....	112
Table 5.3 Grammar error for assignment 8.....	113
Table 5.4 Error calculation for assignment 8.....	113
Table 5.5 Confusion Matix for assignment 8.....	114
Table 5.6 Spelling error for assignment 9.....	115
Table 5.7 Grammar error for assignment 9.....	115
Table 5.8 Error Calculation for assignment 9 .....	116
Table 5.9 Confusion Matrix for assignment 9 .....	116
Table 5.10 Spelling error for assignment 10.....	117
Table 5.11 Grammar error for assignment 10.....	117
Table 5.12 Error Calculation for assignment 10 .....	118
Table 5.13 Confusion Matrix for assignment 10 .....	118
Table 5.14 Summary of 3 new assignments .....	119
Table 5.15 Assignment 8 evaluation for feedback suggestion generation.....	121
Table 5.16 Assignment 9 evaluation for feedback suggestion generation.....	128
Table 5.17 Assignment 10 evaluation for feedback suggestion generation.....	137

## LIST OF FIGURES

Figure 2.1 Screenshot of FeedbackFruits.....	25
Figure 2.2 Screenshot of gradescope .....	26
Figure 2.3 Screenshot of Formative.AI.....	27
Figure 2.4 Screenshot of codePost.....	28
Figure 3.1 Gantt Chart .....	30
Figure 3.2 Use Case Diagram .....	37
Figure 3.3 Sequence Diagram for Login Account .....	56
Figure 3.4 Sequence Diagram for Register Account .....	56
Figure 3.5 Sequence Diagram for Manage Classroom .....	57
Figure 3.6 Sequence Diagram for Manage Student Enrolment .....	57
Figure 3.7 Sequence Diagram for Manage Assignment .....	58
Figure 3.8 Sequence Diagram for Set Assignment Attributes .....	58
Figure 3.9 Sequence Diagram for Manage Grading Rubric .....	59
Figure 3.10 Sequence Diagram for Manage Student Submissions.....	59
Figure 3.11 Sequence Diagram for Download Assignment .....	60
Figure 3.12 Sequence Diagram for Manage Assignment Submission .....	60
Figure 3.13 Sequence Diagram for View Grade.....	61
Figure 3.14 Sequence Diagram for Review Feedback.....	61
Figure 3.15 Sequence Diagram for Provide Feedback, Generate Feedback Suggestions .....	62

Figure 3.16 Sequence Diagram for Grade Student Assignment .....	62
Figure 3.17 Sequence Diagram for View Dashboard .....	63
Figure 3.18 Sequence Diagram for Return Graded Assignment .....	63
Figure 3.19 Entity Relationship Diagram (ERD) .....	64
Figure 3.20 Flow of language-tool-python library. ....	65
Figure 3.21 Flow of Feedback Suggestion Generation.....	66
Figure 4.1 System Architecture .....	69
Figure 4.2 User Login Page .....	70
Figure 4.3 User Sign Up Page.....	70
Figure 4.4 Instructor Homepage .....	71
Figure 4.5 Instructor create classroom.....	71
Figure 4.6 Instructor updates classroom. ....	72
Figure 4.7 Instructor deletes classroom. ....	72
Figure 4.8 Instructor classroom details page. ....	73
Figure 4.9 Instructor creates assignment. ....	73
Figure 4.10 Instructor updates assignment. ....	74
Figure 4.11 Instructor deletes assignment. ....	74
Figure 4.12 Instructor manages grading rubric.....	75
Figure 4.13 Instructor adds grading rubric .....	75
Figure 4.14 Instructor saves grading rubric .....	76
Figure 4.15 Instructor deletes grading rubric.....	76

Figure 4.16 Instructor enrols student .....	77
Figure 4.17 Instructor removes student .....	77
Figure 4.18 Instructor assignment details page.....	79
Figure 4.19 Instructor views submission page.....	80
Figure 4.20 Instructor view submission page after releasing submission .....	80
Figure 4.21 Instructor grading & feedback page (while giving feedback) .....	81
Figure 4.22 Instructor grading & feedback page. ....	82
Figure 4.23 Instructor edits comments.....	83
Figure 4.24 Instructor deletes comments .....	83
Figure 4.25 Student homepage .....	84
Figure 4.26 Student join a classroom.....	84
Figure 4.27 Student classroom details page .....	85
Figure 4.28 Student assignment details page (with feedback button) .....	85
Figure 4.29 Student assignment details page by default.....	86
Figure 4.30 Student submission page (Before due date, without submission) .....	87
Figure 4.31 Student submission page (After due date, without submission).....	87
Figure 4.32 Student submission page (after submission) .....	88
Figure 4.33 Student feedback review page. ....	89
Figure 4.34 Importing essential modules.....	90
Figure 4.35 Core function .....	90
Figure 4.36 json output .....	91

Figure 4.37 Code snippet .....	92
Figure 4.38 Cosine similarity code snippet.....	92
Figure 4.39 Sorting code snippet .....	92
Figure 5.1 Test case for Login Account.....	97
Figure 5.2 Test case for Register Account.....	97
Figure 5.3 Test case for Manage Classroom.....	97
Figure 5.4 Test case for Manage Student Enrolment.....	98
Figure 5.5 Test case for Set Assignment Attributes .....	98
Figure 5.6 Test case for Manage Grading Rubric.....	98
Figure 5.7 Test case for Manage Assignment Submission.....	99
Figure 5.8 Test case for Grade Student Assignment.....	99
Figure 5.9 Test result Login Account .....	100
Figure 5.10 Test result for Register Account.....	101
Figure 5.11 Test result for Manage Classroom.....	102
Figure 5.12 Test result for Manage Student Enrolment.....	103
Figure 5.13 Test result for Set Assignment Attributes .....	104
Figure 5.14 Test result for Manage Grading Rubric.....	106
Figure 5.15 Test result for Manage Assignment Submission .....	107
Figure 5.16 Test result for Grade Student Assignment.....	108
Figure 5.17 Before evaluation.....	120
Figure 5.18 After evaluation.....	120

**LIST OF ABBREVIATIONS AND SYMBOLS**

ERD	- Entity Relationship Diagram
IDE	- Integrated Development Environment
LMS	- Learning Management System
WBS	- Work Breakdown Structure
SMC	- Spelling Mistake Correction
TF-IDF	- Term Frequency and Inverse Document Frequency
OS	- Operating System
UML	- Unified Modeling Language
HTML	- HyperText Markup Language
CSS	- Cascading Style Sheets
NLTK	- Natural Language Toolkit

# CHAPTER 1: INTRODUCTION

## 1.1. Project Background

Over the past decade, the enrollment in higher education in Malaysia has seen a significant increase with over 400,000 new students enrolling in higher education institutions (HEIs) every year. The gross higher education enrollment rate in Malaysia reached 48% in 2012, representing a 70% increase in enrollment from 2004 to 2014, with 1.2 million students enrolled in both public and private HEIs. According to the Ministry of Education in Malaysian there are currently 20 public universities, 36 polytechnics, and 105 community colleges [1]. The number of students enrolled strikes up to the highest in 2016 with 1,346,858 students. In addition, the general statistic of students and instructor ratio at public universities was approximately 18.93 students for every instructor available in 2022 [2]. However, the number of students in class can be more than a few hundred depending on courses.

With the massive number of students, it is challenging for instructors to provide quality feedback to each student due to the increasing workload and a growing demand for personalized feedback on technical written assignments. This leads to the results of dissatisfaction and limited engagement between instructor and students [3]. Other than that, the grading process can be repetitive especially for open-ended technical written assignments. It is a tedious and time-consuming process for instructors to provide personalized feedback. As for students at university level, they usually do not get any feedback from the instructors on their assignments, which is not helpful for the improvement of students. According to Harvard Graduate School of Education, feedback should be given in a timely manner to help students understand what they are doing correctly and incorrectly, with the focus on what they are doing in their assignments [4].

Therefore, SemakLah, is proposed that aims to create a web application that can perform semi-automatic grading and suggest feedback based on components in a grading rubric on technical written assignments submitted by university level students. On the contrary, students can upload their assignments in PDF format, and they can review the feedback from the instructors.

## 1.2. Problem Statement

The problem statement is that instructors face difficulties in providing timely and consistent feedback on technical written assignments, particularly when dealing with many students in a classroom. This is due to the increasing workload of the instructor, especially during the feedback and grading process. Grading technical written assignments is a tedious and time-consuming process for instructors. Plus, providing personalized feedback, and going through the assignment for each student in detail is even more challenging. Instructor would not know the lexical performance of the student by having a glance at their assignments.

Additionally, instructor might want to reuse the same feedback for the similar mistakes made by different students. With that being said, providing feedback and grading the same questions repeatedly from different students can be monotonous for instructors, and they could have utilized this time to work on more meaningful tasks such as preparing lessons or conducting research.

The lack of timely feedback on technical written assignments can lead to several problems. Firstly, students do not receive feedback on their work, which means they cannot learn from their mistakes and improve their performance. This can lead to dissatisfaction for both instructors and students, as well as a lack of engagement and motivation for students to improve their work [3]. Secondly, students may not be able to identify their weaknesses and strengths, which can hinder their learning progress.

## 1.3. Motivation

This project is driven by the need to overcome the challenges faced by the instructor in the current grading process. The motivation of this project is to create a web application that will encourage and help instructors to provide timely, effective, and consistent feedback to students as well as reducing the workload for the instructors. Other than that, the desire of students to learn from their mistakes through receiving feedback for their assignments also motivates the creation of this project.

## 1.4. System Objectives

The objectives of this system are:

- To provide an online assignment management system in which students can upload their written assignments whilst instructors can semi-automatically review student assignments.
- To perform lexical analysis to access overall language and grammatical quality of the writing assignments.
- To provide instructors with automatic feedback suggestions based on student solutions.
- To provide instructors with grading features to grade students' assignments.

## 1.5. Proposed Solution

SemakLah, is the proposed web application that can perform semi-automatic grading and generate feedback suggestions based on historical feedback given by instructor in consideration of components in a grading rubric.

SemakLah consists of 4 modules, which are Classroom Management, Assignment Management, Feedback Generation and Grading. Instructor can set up a classroom with assignments for students. Students can download the assignment and they will upload their assignment in PDF for the assignment submission. When the instructor is reviewing an assignment, the system will perform text extraction for lexical analysis using the natural language processing techniques. The instructor can create feedback for the students, and the system will generate top 3 feedback suggestions (based on the historical feedback given by instructor) when the instructor highlights a similar text. The instructors can choose to accept the suggested feedback as it is, edit or overwrite the suggested feedback. Other than providing feedback, instructors can also grade students' assignments. Once grading is completed and the instructor publishes the feedback and grades, students can view the feedback through SemakLah.

Module Features:

### 1.5.1. Classroom Management

1. Log in for instructors and students.
2. Registration for new users.
3. Management of online classrooms with different courses.
4. Management of student enrolment to online classrooms.

### 1.5.2. Assignment Management

1. Creation and editing of assignments.
2. Setting assignment attributes (title, description, instructions, start date, end date, file upload)
3. Management of the list of student submissions.
4. Downloading assignments for students.
5. Uploading, editing, and reviewing assignment submissions by students.

### **1.5.3. Feedback Generation**

1. Extraction of digital text from assignment files.
2. Lexical analysis for spelling and grammatical errors.
3. Similarity matching with feedback history.
4. Top 3 feedback suggestions based on history and highlighted text at the character level.
5. Review of the summary of lexical analysis.
6. Highlighting text to provide feedback.
7. Selection of feedback suggestions and editing.

### **1.5.4. Grading Module**

1. Management of grading rubrics with different weightage.
2. Entry of scores according to grading rubrics.
3. Viewing generated final scores and grades for each student assignment.
4. Reviewing the dashboard of grading progress.
5. Returning feedback and grades for students to review.
6. Reviewing feedback and grades given by instructors for students.

## 1.6. Benefits and Uniqueness

SemakLah, the proposed web application for semi-automatic grading and feedback generation offers several benefits to both instructors and students.

First, instructors can provide timely and effective feedback to students. This is particularly important in classes with many students, where providing feedback on written assignments can be a tedious task. With SemakLah, instructors can save time on giving feedback for students' assignments, allowing them to focus on other important tasks.

Second, the workload for instructors can be reduced. By automating the grading and feedback generation process, instructors can spend less time grading assignments and more time on other important tasks, such as lesson planning and research. This can lead to increased job satisfaction and a better work-life balance for instructors.

Third, students can receive feedback that is consistent, fair, and objective. With the use of grading rubrics and natural language processing techniques, the feedback provided will be more consistent and objective, ensuring that all students are graded fairly. This can help to improve student engagement and motivation, as they will be able to see how they are progressing and what they need to do to improve their work.

Fourth, students can view their feedback through the application, which can help them improve their work. By providing students with access to their feedback through the application, they can review their work and identify areas for improvement. This can help to improve the overall quality of assignments and reports submitted by students, leading to better academic outcomes.

Finally, SemakLah is unique because it combines the use of grading rubrics with natural language processing techniques to generate feedback at two different levels of analysis: lexical and rubric. The system also allows instructors to choose to accept the suggested feedback as it is, edit or overwrite the suggested feedback.

## 1.7. SDG Alignment

SemakLah is aligned with the SDG 4: Quality Education, as it aims to improve the quality of education by providing timely and effective feedback to students.

## 1.8. Report Organization

This report is organised into 4 sections:

Chapter 1 – Introduction:

- This chapter includes project background, problem statements, motivation, system objectives, proposed solutions, and benefits and uniqueness of the proposed solution.

Chapter 2 –Related Work:

- This chapter includes the existing systems, competitor analysis and existing algorithms/methods.

Chapter 3 – System Requirements and Designs:

- This chapter includes the project scope, system capabilities, system limitations, project management, development methodology, requirements, and detailed analysis of the proposed solution.

Section 4 – System Design & Implementation

- This chapter includes the system architecture, user interface design, implementation of intelligent component and implementation strategy.

Section 5 – System Testing & Evaluation

- This chapter includes the testing strategy (unit testing, integration testing and system testing), test cases, test results and intelligent component evaluation.

Section 6 – Conclusion & Future Work:

- This chapter concludes this final report and includes some of the future work.

## CHAPTER 2: RELATED WORK

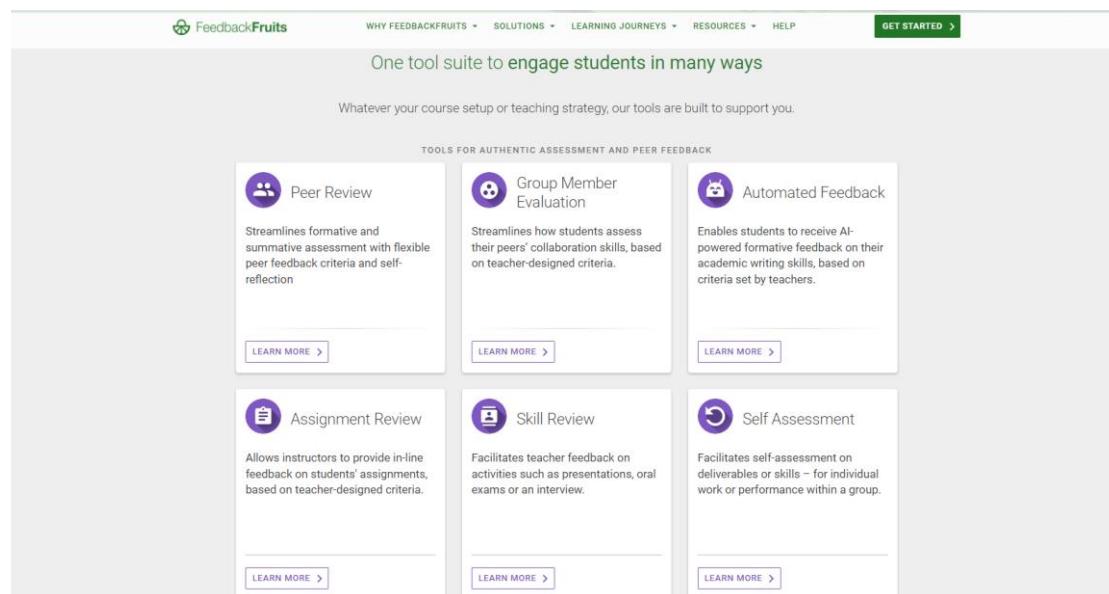
### 2.1. Existing Systems

There are a few existing systems that have similar scope including Feedbackfruit, gradescope, formative.AI and codePost.

#### 2.1.1. FeedbackFruits

FeedbackFruits is a digital teaching tool for higher education that enhances student engagement, collaboration, and feedback in both asynchronous and synchronous learning. FeedbackFruits allows teachers to structure and customize feedback criteria with options such as comment criterion, (Likert) scales, rubrics, or a combination of multiple types. The platform allows teachers to reuse any feedback comment they post within an assignment, saving time in cases where there is recurring feedback that must be given on multiple students' deliverables. However, it does not perform any lexical analysis and it does not automatically generate feedback suggestions based on the historical feedback when the instructor highlights a similar text [5].

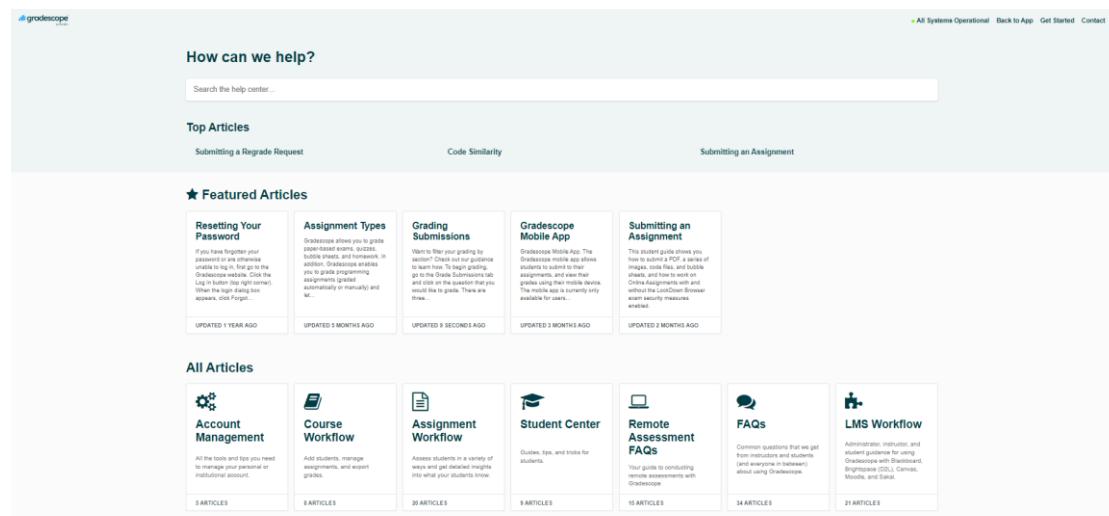
*Figure 2.1 Screenshot of FeedbackFruits*



### 2.1.2. Gradescope

Gradescope is an online tool that allows instructors to grade various types of assignments, including paper-based and programming assignments. It offers features such as creating and using rubrics for grading, managing submissions, and providing detailed analytics. For paper assignments, it supports grading for paragraphs, proofs, diagrams, fill-in-the-blank, true or false, and more. It also allows for the grading of programming assignments, either automatically or manually. Gradescope is primarily used in high school and higher-ed courses in subjects such as Math, Chemistry, Computer Science, Physics, Economics, and Business, which does not cater to university level. In addition to the limitations, Gradescope does not provide lexical analysis, and when an instructor highlights text and provides feedback, it will not generate historical feedback from previous submissions [6].

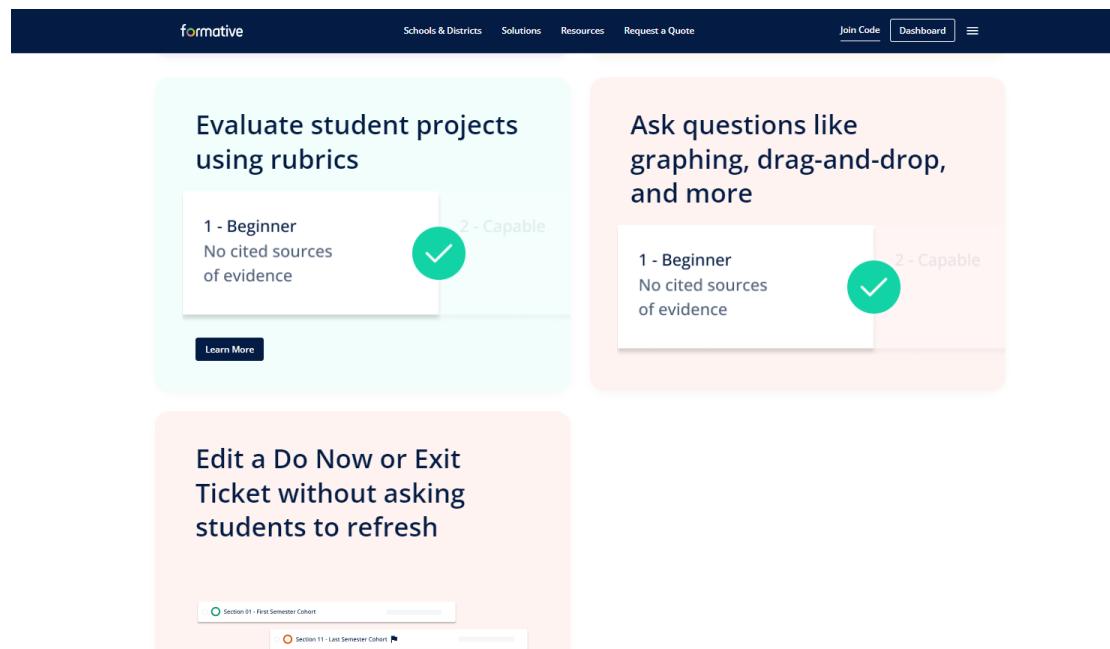
Figure 2.2 Screenshot of gradescope



### 2.1.3. Formative.ai

Formative.ai is an emerging artificial intelligence technology that is integrated into the Formative platform, which is a website designed to help educators create digital assignments, monitor learners' work and engagement in real-time, and provide feedback. The strengths of Formative AI include its potential to automate content creation and provide real-time feedback. However, it does not allow the instructor to provide specific feedback in the file directly. Instructor needs to download the file to provide feedback and re-upload the file for students to view. On top of the limitations, the system does not have the function to generate feedback suggestions from the historical feedback based on the text highlighted by the instructor [7].

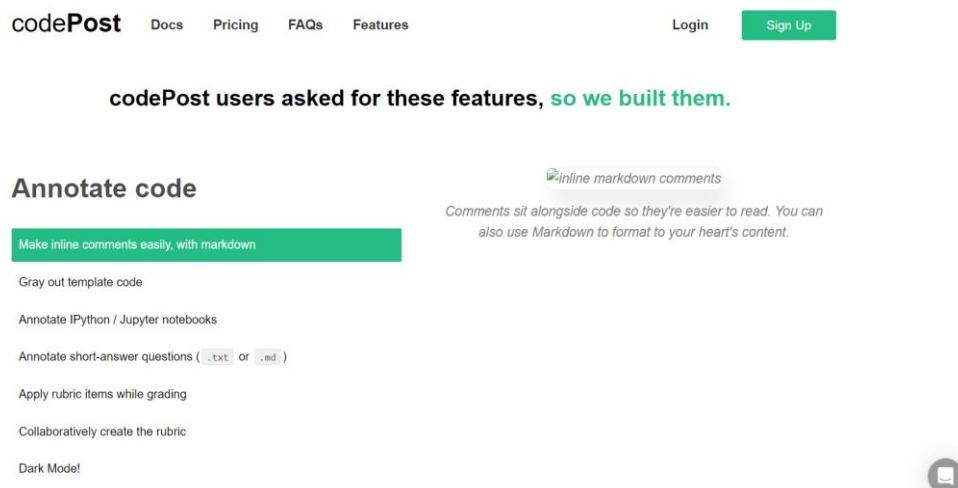
Figure 2.3 Screenshot of Formative.AI



#### 2.1.4. CodePost

CodePost is an online platform designed to help computer science instructors grade and provide feedback on student code. It allows instructors to create assignments, set up grading rubrics, and provide feedback on student code. CodePost also includes features such as plagiarism detection and code auto grading. One of the strengths of CodePost is its ability to handle large classes and complex assignments. However, it is more focused on programming assignments and does not support lexical analysis. Other than that, it does not have the automatic generation of historical feedback feature as well [8].

Figure 2.4 Screenshot of codePost



## 2.2. Competitor Analysis

Table 2.1 Competitor Analysis

Competitors	SemakLah	Feedbackfruit	Gradescope	Formative.AI	codePost
Features					
Create Classroom	Yes	Yes	Yes	Yes	Yes
Manage Assignment	Yes	Yes	Yes	Yes	Yes
Set Assignment Attributes	Yes	Yes	Yes	Yes	Yes
Support PDF Submission	Yes	Yes	No	Yes	No
Customizable criteria of rubrics	Yes	Yes	Yes	Yes	Yes
Generate Feedback Suggestion Based on Historical Feedback	Yes	No	No	No	No
Configurable grading	Yes	Yes	Yes	Yes	Yes
Provide Lexical Analysis	Yes	No	No	Yes	No
Monitor Grading Progress	Yes	Yes	Yes	Yes	Yes

## 2.3. Existing Algorithms

This section will discuss the existing algorithms that have been applied to lexical analysis for language and grammatical quality assessment, and text-matching solutions.

### 2.3.1. Lexical Analysis for spelling and grammar check

Golding and Schabes [9] introduced a hybrid approach called 'Tribayes' that combines the trigram and Bayesian methods, which proved to be effective in correcting real-word spelling errors. The trigram model, a second-order Markov process for language modeling, encodes the context using part-of-speech trigrams to identify the correct word from a confusion set of real words generated by single edit operations. The Bayesian approach is a feature-based method that uses context words and collocations to determine the probability that a word is correct. The 'Tribayes' approach used in the spelling mistake correction (SMC) model for real-word error correction involves the use of contextual information and statistical language modeling. The trigram model captures the contextual information using part-of-speech trigrams, while the Bayesian approach incorporates feature-based methods to improve the accuracy of real-word error correction in the SMC model.

### 2.3.2. Text-Matching Solution

Fuzzy string matching is a technique for finding strings that match a pattern approximately. Li demonstrated the application of the FuzzyWuzzy library in Python, which used the Levenshtein Distance to calculate the differences between strings. The article provides examples of how fuzzy string matching can be used, such as spell checking, spelling-error correction, and checking for duplicate records. It also explains the importance of fuzzy string matching in scenarios where exact string matching is not sufficient, such as when dealing with misspelled words or partial input. The FuzzyWuzzy library in Python is used to perform fuzzy string matching by calculating the differences between strings, allowing for approximate matching of strings even in the presence of misspellings or partial input. The article provided examples of using the FuzzyWuzzy library to compare strings and demonstrates its application in scenarios such as spell checking and duplicate record detection [10].

### 2.3.3. Existing Feedback Generation Method

The study compares three feedback comment generation methods: retrieval-based, simple generation, and retrieve-and-edit. The retrieval-based method encodes the input with its context and predicts the feedback comment with attention and copy mechanisms. It uses independent networks called retriever and editor. The simple generation method directly generates feedback comments without using retrieval-based techniques. It involves encoding the input with its context and predicting the feedback comment using a neural network. On the other hand, the retrieve-and-edit method is a hybrid of retrieval-based and simple generation. It first retrieves candidate feedback comments from a retrieval-based system and then edits the comments using a neural network to generate the final feedback comment. The study found that the performance order is not as expected, with different methods outperforming others in specific types of feedback comment generation. For instance, simple generation outperforms the other two methods in preposition feedback comment generation, while retrieval-based exhibits the best performance in general feedback comments [11].

## CHAPTER 3: SYSTEM REQUIREMENTS AND DESIGN

### 3.1. Project Development Status

SemakLah is a new project that will be built from scratch and has never been developed before.

### 3.2. Project Scope

The proposed project is a web application designed for university instructors and students, aimed at streamlining the management and evaluation of technical written assignments. This platform will provide a centralized hub for instructors to assign tasks, offer feedback, and grade submissions efficiently. Students will benefit from a user-friendly interface for submitting assignments, accessing assignment rubrics, and receiving instructor feedback.

The system is scoped to ensure simplicity and clarity:

- Each instructor manages a single classroom.
- The platform accepts only one submission per assignment.
- All submissions must be in PDF format.
- The system supports only English.
- It is tailored exclusively for question-based written assignments.

Overall, the project aims to create a robust digital environment that fosters collaboration, communication, and academic excellence within the university setting.

### **3.3. System Capabilities**

1. Classroom Management:
  - a. Creation of classrooms.
  - b. Manage student enrolment.
2. Assignment Management:
  - a. Instructor manages assignments.
  - b. Students can download assignments.
  - c. Support for file uploads in PDF format.
  - d. Efficient organization and tracking of assignments for each class.
3. Feedback Generation:
  - a. Natural Language Processing (NLP) techniques for extracting text and performing lexical analysis that shows the overall language and grammatical quality of writing assignments.
  - b. Instructors can create feedback for students.
  - c. Historical feedback data is used to generate suggestions for feedback based on similar text.
  - d. Flexibility for instructors to accept, edit, or overwrite suggested feedback.
4. Grading:
  - a. Instructors can assign grades to student assignments.
  - b. Efficient grading process facilitated by the system.
  - c. Instructors can return graded assignments.
  - d. Students can access their feedback and grades through the system.

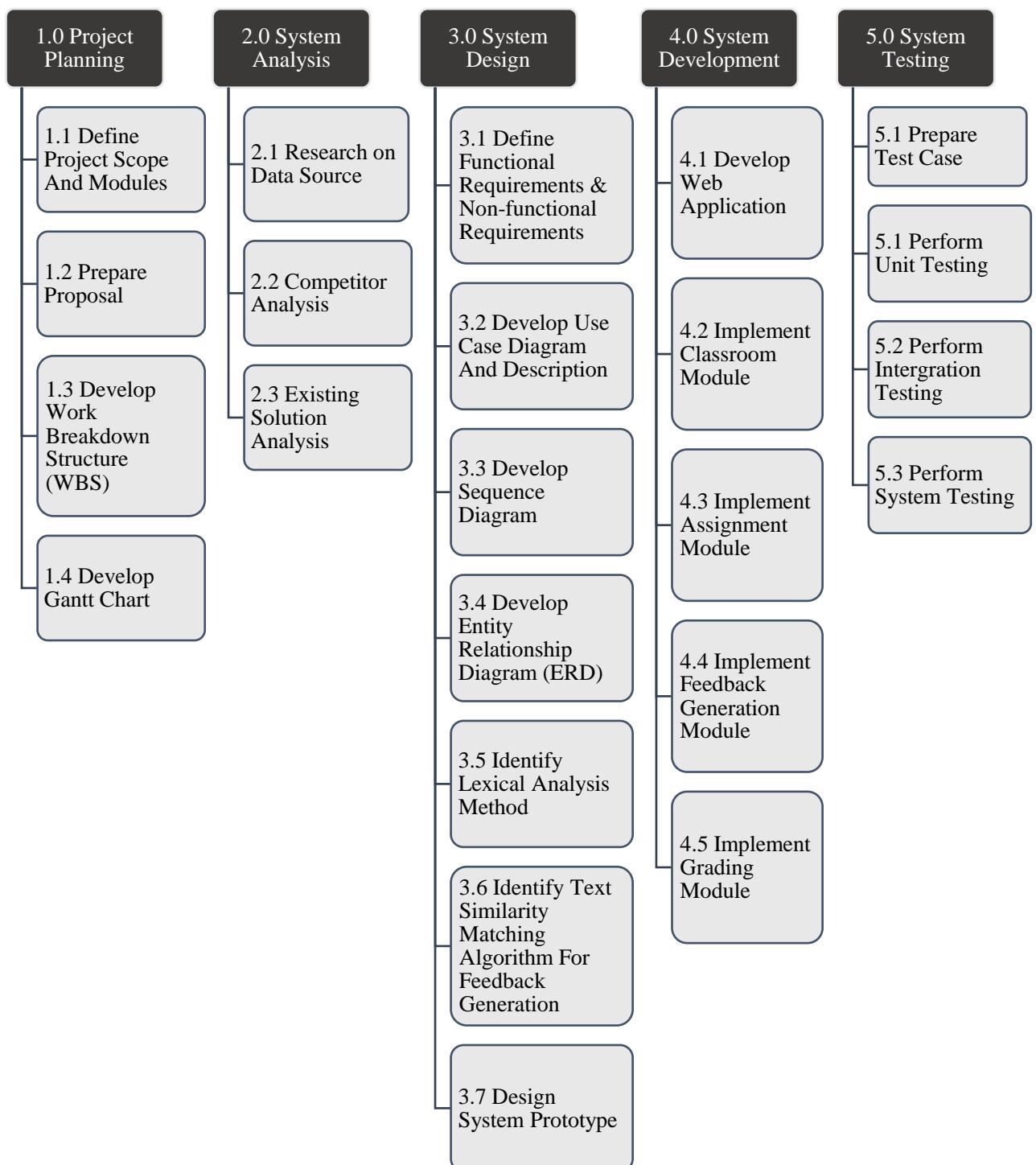
### **3.4. System Limitations**

1. Instructor limitation in a classroom and assignment:
  - a. The system only allows one instructor in one classroom.
  - b. The system only allows one instructor to create an assignment, it does not include collaboration between instructors.
2. Student-student interaction is limited:
  - a. The student will not be able to interact with one another in any form in using this system.
3. Language is limited:
  - a. Lexical analysis is limited to only English language as NLP tools can only support English.
4. The structure of grading rubrics is fixed:
  - a. The system will have one type of grading rubrics available for all instructors.

## 3.5. Project Management

### 3.5.1. Work Breakdown Structure (WBS)

*Table 3.1 Work Breakdown Structure (WBS)*



### 3.5.2. Gantt Chart

Figure 3.1 Gantt Chart

Milestone description	Start	Oct 2023			Nov 2023			Dec 2023			Jan 2024			Feb 2024			Mar 2024			Apr 2024			May 2024			June 2024											
		W1 16	W2 23	W3 30	W4 6	W5 13	W6 20	W7 27	W8 4	W9 11	W10 18	W11 25	W12 1	W13 8	W14 15	W15 22	W16 29	W17 5	W18 12	W19 19	W20 26	W21 4	W22 11	W23 18	W24 25	W25 1	W26 8	W27 15	W28 22	W29 29	W30 6	W31 13	W32 20	W33 27	W34 3	W35 10	W36 17
<b>1.0 Project Planning</b>																																					
Define Project Scope and Modules	30/10/2023																																				
Prepare Proposal	16/10/2023																																				
Develop WBS	23/10/2023																																				
Develop Gantt Chart	23/10/2023																																				
<b>2.0 System Analysis</b>																																					
Research on Data Source	1/11/2023																																				
Competitor Analysis	6/11/2023																																				
Existing Solution Analysis	13/11/2023																																				
<b>3.0 System Design</b>																																					
Define Functional Requirements & Non-Functional Requirements	20/11/2023																																				
Develop Use Case Diagram and Descriptions	27/11/2023																																				
Develop Sequence Diagram	4/12/2023																																				
Develop ERD	11/12/2023																																				
Identify Lexical Analysis Method	18/12/2023																																				
Identify Text-Similarity Matching Algorithm for Feedback Generation	25/12/2023																																				
Design System Prototype Phase 1 (Classroom & Assignment Management Module)	1/1/2024																																				
Design System Prototype Phase 2 (Feedback Generation & Grading Module)	15/1/2024																																				
<b>4.0 System Development &amp; 5.0 System Testing</b>																																					
Develop Web Application	29/1/2024																																				
Implement Classroom Module	5/2/2024																																				
System Unit Testing	19/2/2024																																				
Implement Assignment Module	26/2/2024																																				
System Unit Testing	11/3/2024																																				
Implement Feedback Generation Module	18/3/2024																																				
System Unit Testing	1/4/2024																																				
Implementation of Grading Module	8/4/2024																																				
System Unit Testing	22/4/2024																																				
System Integration Testing	29/4/2024																																				
System Testing	13/5/2024																																				

### 3.5.3. SWOT Analysis

*Table 3.2 SWOT Analysis*

Strength	Weaknesses
<ul style="list-style-type: none"> <li>- Comprehensive tools for instructors to manage classrooms, assignments, and student submission efficiently.</li> <li>- Feedback generation is based on historical data, facilitating consistency and efficiency in providing constructive feedback.</li> <li>- Provide insights into spelling and grammatical error</li> </ul>	<ul style="list-style-type: none"> <li>- The system heavily relies on consistent internet connectivity, which may pose challenges for users in areas with limited internet access.</li> <li>- Automated feedback suggestions may produce false positives or negatives, requiring careful review by instructors.</li> <li>- The system relies on the instructor's input for creating feedback and grading, and the quality of suggestions is contingent on the historical data available.</li> </ul>
Opportunity	Threats
<ul style="list-style-type: none"> <li>- Adoption of online learning environments since covid-19.</li> <li>- The maturity of certain AI technologies, such as natural language processing, presents an opportunity to enhance the system's text extraction and analysis capabilities.</li> <li>- The demand for customizable and adaptive online learning environment</li> </ul>	<ul style="list-style-type: none"> <li>- Instructor and students may resist adopting the new system, especially if they are accustomed to existing methods and tools.</li> <li>- Data security and privacy concerns related to text extraction and student information.</li> <li>- Competition from other educational platforms offering similar features.</li> </ul>

### 3.6. Development Methodology

The proposed semi-automatic grading web application adopts an agile development methodology that aligns with the project needs for continuous adaptation to evolve requirements through iterative development cycles. Agile methodology can be applied to all individual modules.

In the classroom management module, the agile methodology will be applied through short sprints. The focus of Sprint 1 is on user authentication and account registration, while sprint 2 is dedicated to classroom creation and student enrolment management.

In assignment management module, sprint 3 might focus on assignment management which includes creation, viewing, updating, deleting, while sprint 4 will handle the submission management. This approach ensures that each functionality is developed incrementally and can be refined based on user feedback at the end of each sprint.

In the feedback generation module, sprint 5 and sprint 6 will focus on text extraction and lexical analysis respectively. The iterative development allows for early testing and refinement, ensuring that the system meets the instructor's feedback needs effectively.

Finally, each functionality within the grading module can be addressed in separate sprints. Sprint 7 might involve the development of grading rubric management, while Sprint 8 could focus on the implementation of score entry and viewing functionalities. This approach allows for regular validation and adjustments based on instructor feedback.

By breaking down the development process into manageable sprints, each lasting for a week, the project can regularly reassess its goals, accommodate changes, and receive ongoing feedback.

## 3.7. User Requirements

### 3.7.1. Requirement Gathering

The requirement gathering process for the proposed system was informed by survey of existing feedback and grading systems, together with collaborative brainstorming sessions with my supervisor. The survey involved a thorough observation of similar platforms, identifying their features, functionalities, and areas for improvement. Additionally, engaging in brainstorming sessions with my supervisor provided valuable insights into the specific needs and challenges faced by instructors in the context of grading and providing feedback.

### **3.7.2. Functional Requirements**

#### **Classroom Management Module**

1. Instructor and student shall log in to their respective accounts.
2. Instructor and student shall register for an account if they are new users.
3. Instructor shall manage online classrooms with different courses.
4. Instructor shall manage student enrolment to the online classroom.

#### **Assignment Management Module**

1. Instructor shall manage assignments.
2. Instructor shall set assignment attributes such as title, description, instructions, assignment start date, assignment end date, and upload the assignment question in PDF format.
3. Instructor shall manage the list of student submissions.
4. Student shall download the assignments.
5. Student shall manage their assignment submission by uploading the assignment file in PDF, deleting, or reviewing the submitted assignment.

#### **Feedback Generation Module**

1. System shall extract digital text from the assignment files.
2. System shall perform lexical analysis which includes analyzing the number of spelling errors and number of grammatical errors.
3. System shall perform similarity matching between the highlighted text and feedback history (i.e., text units in prior assignments with feedback).
4. Instructor will receive top 3 feedback suggestions based on feedback history when similar text is highlighted at character level.
5. Instructor shall be able to review the summary of lexical analysis.
6. Instructor shall highlight text to provide feedback.
7. Instructor shall edit on the feedback suggestions.

### Grading Module

1. Instructor shall manage grading rubrics with different weightage.
2. Instructor shall enter scores according to the grading rubrics for each assignment submitted by the student.
3. Instructor shall view the generated final scores and grade for each student assignment.
4. Instructor shall review the dashboard of grading progress.
5. Instructor shall return the feedback and grades for students to review.
6. Student shall review the feedback and grades given by the instructors when the instructor returns the feedback and grading.

### 3.7.3. Non-Functional Requirements

#### 1. Performance:

- The system should provide a response time of less than 3 seconds for user interactions (e.g., logging in, creating assignments, submitting assignments).
- The system should be able to handle concurrent logins and interactions from at least 1000 users without significant performance degradation.

#### 2. Scalability:

- The system should be designed to accommodate a minimum of 5000 users and 100 classrooms initially, with the ability to scale easily to support growth.
- The system must be scalable enough to support 500,000 visits at the same time while maintaining optimal performance.

#### 3. Portability:

- The system should work well on Windows 10 and Windows 11 without any change in its behavior and performance.

#### 4. Availability:

- The system should be available to Malaysia users 99.9% of the time every month.
- Planned maintenance should be scheduled during off-peak hours to minimize impact on users.

#### 5. Security:

- User authentication and authorization should be secured.
- The system should have role-based access control to ensure that only authorized users can perform specific actions.

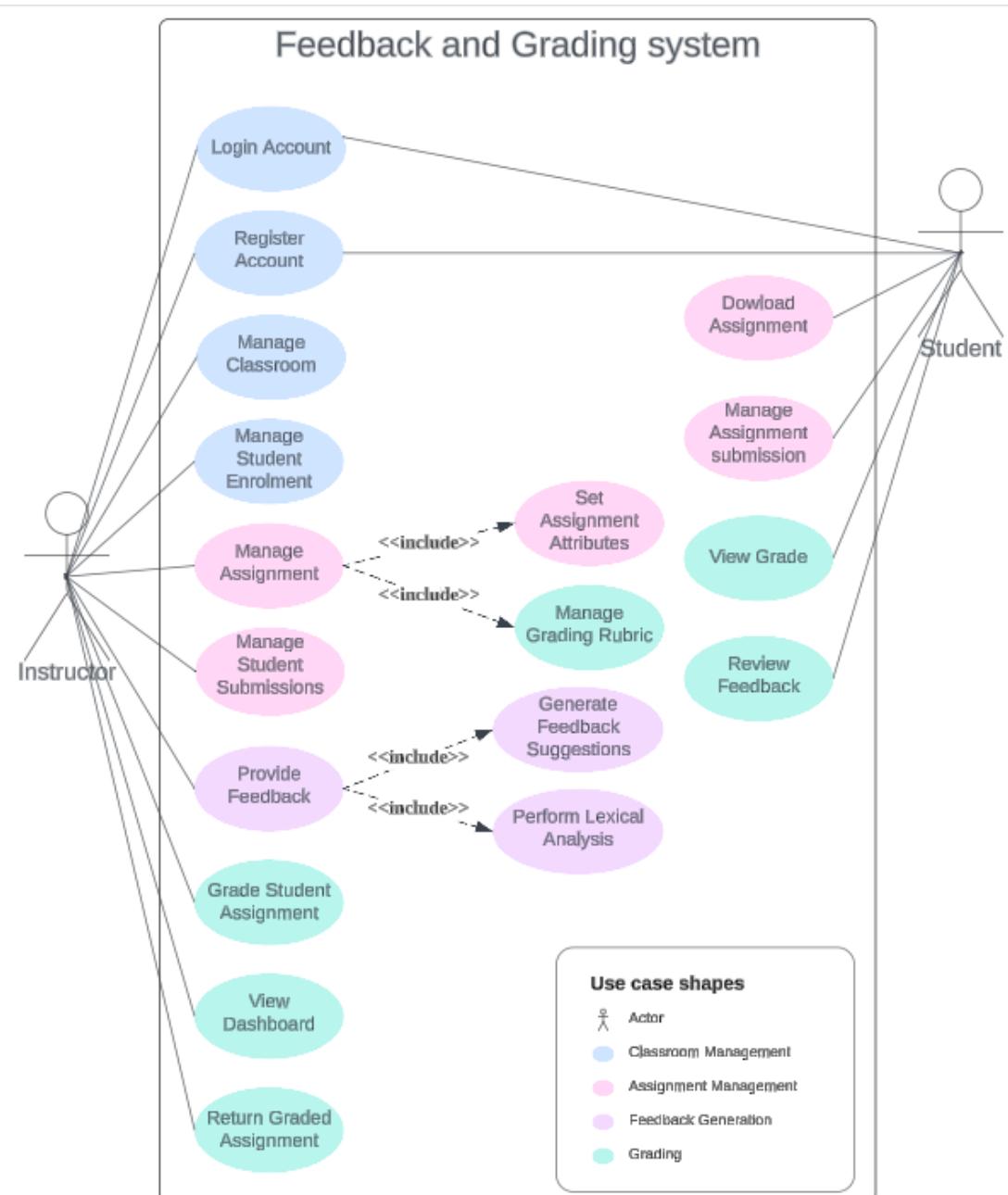
#### 6. Localization:

1. The date format must be as follows: dd/mm/yyyy

## 3.8. System Analysis

### 3.8.1. Use Case Diagram

Figure 3.2 Use Case Diagram



### 3.8.2. Use Case Descriptions

#### 3.8..1 UC001: Login Account

*Table 3.3 Use Case Description for UC001 Login Account*

<b>Use Case Name:</b>	Login Account					
<b>Scenario:</b>	A registered user attempts to access the system.					
<b>Triggering Event:</b>	User opens SemakLah website and initiates the login process.					
<b>Brief Description:</b>	Users (Instructor and student) log in to their respective accounts.					
<b>Actors:</b>	Instructor and Student					
<b>Related use cases:</b>	N/A					
<b>Preconditions:</b>	Users have valid credentials.					
<b>Postconditions:</b>	User successfully logged into the system.					
<b>Flow of activities:</b>	<table border="1"> <thead> <tr> <th>Actor</th> <th>System</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>User navigates to the login.</li> <li>User enters their email and password.</li> <li>User login successfully</li> </ul> </td><td> <ul style="list-style-type: none"> <li>System verifies credentials.</li> <li>System grants access if authentication is successful.</li> </ul> </td></tr> </tbody> </table>	Actor	System	<ul style="list-style-type: none"> <li>User navigates to the login.</li> <li>User enters their email and password.</li> <li>User login successfully</li> </ul>	<ul style="list-style-type: none"> <li>System verifies credentials.</li> <li>System grants access if authentication is successful.</li> </ul>	
Actor	System					
<ul style="list-style-type: none"> <li>User navigates to the login.</li> <li>User enters their email and password.</li> <li>User login successfully</li> </ul>	<ul style="list-style-type: none"> <li>System verifies credentials.</li> <li>System grants access if authentication is successful.</li> </ul>					
<b>Exception Condition:</b>	Invalid credentials: - User receives an error message and is prompted to re-enter credentials.					

### 3.8..2 UC002: Register Account

*Table 3.4 Use Case Description for UC002 Register Account*

<b>Use Case Name:</b>	Register Account	
<b>Scenario:</b>	A new user creates an account.	
<b>Triggering Event:</b>	User navigates to SemakLah sign up page.	
<b>Brief Description:</b>	Users (Instructor and Student) create accounts to gain access to the system.	
<b>Actors:</b>	Instructor and Student	
<b>Related use cases:</b>	N/A	
<b>Preconditions:</b>	User does not have an account.	
<b>Postconditions:</b>	User account is successfully registered.	
<b>Flow of activities:</b>	<b>Actor</b> <ul style="list-style-type: none"> <li>• User enters required registration information. (e.g. id, first name, last name, email, password, gender, and role)</li> </ul>	<b>System</b> <ul style="list-style-type: none"> <li>• System validates information.</li> <li>• System creates a new account if validation is successful.</li> </ul>
<b>Exception Condition:</b>	Duplicate email account <ul style="list-style-type: none"> <li>- System displays an error message, prompting the user to choose a different email.</li> </ul>	

### 3.8..3 UC003: Manage Classroom

*Table 3.5 Use Case Description for UC003 Manage Classroom*

<b>Use Case Name:</b>	Manage Classroom									
<b>Scenario:</b>	Instructor oversees online classroom.									
<b>Triggering Event:</b>	Instructor selects any “Manage Classroom” option which includes (create, update, and delete)									
<b>Brief Description:</b>	Instructor manages online classrooms based on different courses.									
<b>Actors:</b>	Instructor									
<b>Related use cases:</b>	N/A									
<b>Preconditions:</b>	Instructor is logged into the system.									
<b>Postconditions:</b>	Instructor successfully manages the selected classroom.									
<b>Flow of activities:</b>	<table border="1"> <thead> <tr> <th>Actor</th> <th>System</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>Instructor creates classroom.</li> <li>Instructor enters required information. (e.g. Course Code, Course Description, Classroom limit)</li> <li>Instructor creates classroom successfully</li> </ul> </td><td> <ul style="list-style-type: none"> <li>System validates information.</li> <li>System creates a new classroom if validation is successful.</li> </ul> </td></tr> <tr> <td> <ul style="list-style-type: none"> <li>Instructor updates classroom details.</li> <li>Instructor enters required information (e.g. Course Code, Course Description, Classroom limit)</li> <li>Instructor updates classroom successfully</li> </ul> </td><td> <ul style="list-style-type: none"> <li>System validates information.</li> <li>System updates classroom information if validation is successful.</li> </ul> </td></tr> <tr> <td> <ul style="list-style-type: none"> <li>Instructor removes classroom.</li> <li>Instructor removes classroom successfully.</li> </ul> </td><td> <ul style="list-style-type: none"> <li>System deletes classroom</li> </ul> </td></tr> </tbody> </table>	Actor	System	<ul style="list-style-type: none"> <li>Instructor creates classroom.</li> <li>Instructor enters required information. (e.g. Course Code, Course Description, Classroom limit)</li> <li>Instructor creates classroom successfully</li> </ul>	<ul style="list-style-type: none"> <li>System validates information.</li> <li>System creates a new classroom if validation is successful.</li> </ul>	<ul style="list-style-type: none"> <li>Instructor updates classroom details.</li> <li>Instructor enters required information (e.g. Course Code, Course Description, Classroom limit)</li> <li>Instructor updates classroom successfully</li> </ul>	<ul style="list-style-type: none"> <li>System validates information.</li> <li>System updates classroom information if validation is successful.</li> </ul>	<ul style="list-style-type: none"> <li>Instructor removes classroom.</li> <li>Instructor removes classroom successfully.</li> </ul>	<ul style="list-style-type: none"> <li>System deletes classroom</li> </ul>	
Actor	System									
<ul style="list-style-type: none"> <li>Instructor creates classroom.</li> <li>Instructor enters required information. (e.g. Course Code, Course Description, Classroom limit)</li> <li>Instructor creates classroom successfully</li> </ul>	<ul style="list-style-type: none"> <li>System validates information.</li> <li>System creates a new classroom if validation is successful.</li> </ul>									
<ul style="list-style-type: none"> <li>Instructor updates classroom details.</li> <li>Instructor enters required information (e.g. Course Code, Course Description, Classroom limit)</li> <li>Instructor updates classroom successfully</li> </ul>	<ul style="list-style-type: none"> <li>System validates information.</li> <li>System updates classroom information if validation is successful.</li> </ul>									
<ul style="list-style-type: none"> <li>Instructor removes classroom.</li> <li>Instructor removes classroom successfully.</li> </ul>	<ul style="list-style-type: none"> <li>System deletes classroom</li> </ul>									
<b>Exception Condition:</b>	N/A									

### 3.8..4 UC004: Manage Student Enrolment

*Table 3.6 Use Case Description for UC004 Manage Student Enrolment*

<b>Use Case Name:</b>	Manage Student Enrolment	
<b>Scenario:</b>	Instructors add or remove student enrolment in an online classroom.	
<b>Triggering Event:</b>	Instructor selects a “Manage Student Enrolment” option for a specific classroom, which includes enrolling and deleting a student.	
<b>Brief Description:</b>	Instructor manages student enrolment for a selected online classroom	
<b>Actors:</b>	Instructor and Student	
<b>Related use cases:</b>	N/A	
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. Instructor is logged into the system.</li> <li>2. Classroom exists.</li> <li>3. Student exists (if instructor is removing a student)</li> <li>4. Instructor selects a classroom</li> </ol>	
<b>Postconditions:</b>	Instructor successfully manages student enrolment for the selected classroom.	
<b>Flow of activities:</b>	<b>Actor</b>	<b>System</b>
	<ul style="list-style-type: none"> <li>• Instructor enters student id.</li> <li>• Student enrols into classroom</li> </ul>	<ul style="list-style-type: none"> <li>• System validates information.</li> <li>• System updates the list of students in the online classroom.</li> </ul>
<b>Exception Condition:</b>	<ul style="list-style-type: none"> <li>• Instructor removes a student</li> </ul>	<ul style="list-style-type: none"> <li>• System updates the list of students in the online classroom.</li> </ul>
	Exceed the class limit: <ul style="list-style-type: none"> <li>- System displays a message indicating that the class limit is exceeded.</li> </ul> Student does not have an account: <ul style="list-style-type: none"> <li>- System displays a message indicating that student not found.</li> </ul>	

### 3.8..5 UC005: Manage Assignment

*Table 3.7 Use Case Description for UC005 Manage Assignment*

<b>Use Case Name:</b>	Manage Assignment									
<b>Scenario:</b>	Instructor creates, edits, or deletes assignments.									
<b>Triggering Event:</b>	Instructor selects any “Manage Assignment” option for a specific classroom (which includes create, edit, and delete)									
<b>Brief Description:</b>	Instructor manages assignments for a selected classroom.									
<b>Actors:</b>	Instructor									
<b>Related use cases:</b>	UC006: Set Assignment Attributes UC007: Manage Grading Rubrics									
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. Instructor is logged into the system.</li> <li>2. A classroom exists.</li> <li>3. Instructor selects a classroom</li> </ol>									
<b>Postconditions:</b>	Instructor successfully manages assignments for the selected classroom.									
<b>Flow of activities:</b>	<table border="1"> <thead> <tr> <th><b>Actor</b></th> <th><b>System</b></th> </tr> </thead> <tbody> <tr> <td>• User creates an assignment</td> <td>• System prompt user to set assignment attributes</td> </tr> <tr> <td>• User updates an assignment.</td> <td>• System prompt user to update assignment attributes</td> </tr> <tr> <td>• User removes an assignment.</td> <td>• System removes the assignment</td> </tr> </tbody> </table>	<b>Actor</b>	<b>System</b>	• User creates an assignment	• System prompt user to set assignment attributes	• User updates an assignment.	• System prompt user to update assignment attributes	• User removes an assignment.	• System removes the assignment	
<b>Actor</b>	<b>System</b>									
• User creates an assignment	• System prompt user to set assignment attributes									
• User updates an assignment.	• System prompt user to update assignment attributes									
• User removes an assignment.	• System removes the assignment									
<b>Exception Condition:</b>	N/A									

### 3.8..6 UC006: Set Assignment Attributes

*Table 3.8 Use Case Description for UC006 Set Assignment Attributes*

<b>Use Case Name:</b>	Set Assignment Attributes	
<b>Scenario:</b>	Instructor sets attributes for a specific assignment.	
<b>Triggering Event:</b>	Instructor selects on each section of the assignment attributes.	
<b>Brief Description:</b>	When managing assignment, instructor can set assignment attributes such as title, description, instructions, assignment start date, assignment end date, and upload the assignment question in PDF.	
<b>Actors:</b>	Instructor	
<b>Related use cases:</b>	UC005: Manage Assignment	
<b>Preconditions:</b>	1. Instructor is logged into the system. 2. Instructor is creating assignment in a classroom.	
<b>Postconditions:</b>	Assignment attributes are successfully set	
<b>Flow of activities:</b>	<b>Actor</b> <ul style="list-style-type: none"> <li>• Instructor enters the assignment attribute (e.g. title, description, instructions, deadline, upload the assignment question in PDF, start date and end date)</li> <li>• Instructor creates assignment successfully.</li> </ul>	<b>System</b> <ul style="list-style-type: none"> <li>• System validates the assignment attributes.</li> <li>• System saves the assignment attributes.</li> </ul>
	<ul style="list-style-type: none"> <li>• Instructor updates the assignment attribute (e.g. title, description, instructions, deadline, upload the assignment question in PDF, start date and end date)</li> <li>• Instructor updates assignment successfully.</li> </ul>	<ul style="list-style-type: none"> <li>• System validates the assignment attributes.</li> <li>• System saves the assignment attributes.</li> </ul>
<b>Exception Condition:</b>	Invalid file format: - System displays an error message and prompts the instructor to upload a valid assignment file.	

### 3.8..7 UC007: Manage Grading Rubric

*Table 3.9 Use Case Description for UC007 Manage Grading Rubric*

<b>Use Case Name:</b>	Manage Grading Rubric									
<b>Scenario:</b>	Instructor creates, edits, or deletes grading rubrics with different criteria.									
<b>Triggering Event:</b>	Instructor selects the “Managing Grading Rubric” option for a specific assignment.									
<b>Brief Description:</b>	Instructor manages grading rubric for assignments.									
<b>Actors:</b>	Instructor									
<b>Related use cases:</b>	UC005: Manage Assignment									
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. Instructor is logged into the system.</li> <li>2. An assignment is created.</li> <li>3. Instructor selects on the grading rubric</li> </ol>									
<b>Postconditions:</b>	Instructor successfully manages the grading rubric for the selected assignment.									
<b>Flow of activities:</b>	<table border="1"> <thead> <tr> <th>Actor</th> <th>System</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>• Instructor adds a new row of grading rubric.</li> <li>• Instructor enters information.</li> <li>• Instructor saves the grading rubric successfully</li> </ul> </td><td> <ul style="list-style-type: none"> <li>• System creates a new row of grading rubric.</li> <li>• System verifies the question number.</li> <li>• System saves the grading rubric.</li> </ul> </td></tr> <tr> <td> <ul style="list-style-type: none"> <li>• Instructor updates information.</li> <li>• Instructor enters information.</li> <li>• Instructor saves the grading rubric successfully</li> </ul> </td><td> <ul style="list-style-type: none"> <li>• System updates the grading rubric.</li> <li>• System saves the grading rubric.</li> </ul> </td></tr> <tr> <td> <ul style="list-style-type: none"> <li>• Instructor deletes a row of grading rubric</li> </ul> </td><td> <ul style="list-style-type: none"> <li>• System deletes a row of grading rubric</li> </ul> </td></tr> </tbody> </table>	Actor	System	<ul style="list-style-type: none"> <li>• Instructor adds a new row of grading rubric.</li> <li>• Instructor enters information.</li> <li>• Instructor saves the grading rubric successfully</li> </ul>	<ul style="list-style-type: none"> <li>• System creates a new row of grading rubric.</li> <li>• System verifies the question number.</li> <li>• System saves the grading rubric.</li> </ul>	<ul style="list-style-type: none"> <li>• Instructor updates information.</li> <li>• Instructor enters information.</li> <li>• Instructor saves the grading rubric successfully</li> </ul>	<ul style="list-style-type: none"> <li>• System updates the grading rubric.</li> <li>• System saves the grading rubric.</li> </ul>	<ul style="list-style-type: none"> <li>• Instructor deletes a row of grading rubric</li> </ul>	<ul style="list-style-type: none"> <li>• System deletes a row of grading rubric</li> </ul>	
Actor	System									
<ul style="list-style-type: none"> <li>• Instructor adds a new row of grading rubric.</li> <li>• Instructor enters information.</li> <li>• Instructor saves the grading rubric successfully</li> </ul>	<ul style="list-style-type: none"> <li>• System creates a new row of grading rubric.</li> <li>• System verifies the question number.</li> <li>• System saves the grading rubric.</li> </ul>									
<ul style="list-style-type: none"> <li>• Instructor updates information.</li> <li>• Instructor enters information.</li> <li>• Instructor saves the grading rubric successfully</li> </ul>	<ul style="list-style-type: none"> <li>• System updates the grading rubric.</li> <li>• System saves the grading rubric.</li> </ul>									
<ul style="list-style-type: none"> <li>• Instructor deletes a row of grading rubric</li> </ul>	<ul style="list-style-type: none"> <li>• System deletes a row of grading rubric</li> </ul>									
<b>Exception Condition:</b>	Entering the same question number - System displays an error message and prompts the instructor to fill in a unique question number.									

### 3.8..8 UC008: Manage Student Submissions

*Table 3.10 Use Case Description for UC008 Manage Student Submissions*

<b>Use Case Name:</b>	Manage Student Submissions	
<b>Scenario:</b>	Instructor views & downloads student submission for a specific assignment.	
<b>Triggering Event:</b>	Instructor selects “View Submission” button for a specific assignment.	
<b>Brief Description:</b>	Instructors manage student submissions for a selected assignment.	
<b>Actors:</b>	Instructor	
<b>Related use cases</b>	N/A	
<b>Preconditions:</b>	1. Instructor is logged into the system. 2. An assignment exists.	
<b>Postconditions:</b>	Instructor successfully manages student submissions for the selected assignment.	
<b>Flow of activities:</b>	<b>Actor</b> <ul style="list-style-type: none"> <li>• Instructors view the list of students’ submissions.</li> </ul>	<b>System</b> <ul style="list-style-type: none"> <li>• System displays the list of students’ submission.</li> </ul>
	<ul style="list-style-type: none"> <li>• Instructors download student’s submission</li> </ul>	<ul style="list-style-type: none"> <li>• System downloads a copy of student’s submission in the local device.</li> </ul>
<b>Exception Condition:</b>	N/A	

### 3.8..9 UC009: Download Assignment

*Table 3.11 Use Case Description for UC009 Download Assignment*

<b>Use Case Name:</b>	Download Assignment	
<b>Scenario:</b>	Students download assignment questions.	
<b>Triggering Event:</b>	Student selects the "Download file" button.	
<b>Brief Description:</b>	Student downloads assignment.	
<b>Actors:</b>	Student	
<b>Related use cases</b>	N/A	
<b>Preconditions:</b>	1. Student is logged into the system. 2. Assignment in a classroom is available for download.	
<b>Postconditions:</b>	Assignment is successfully downloaded to their local device.	
<b>Flow of activities:</b>	<b>Actor</b> <ul style="list-style-type: none"> <li>• Student downloads the assignment file.</li> </ul>	<b>System</b> <ul style="list-style-type: none"> <li>• System retrieves the assignment file and download into local device</li> </ul>
<b>Exception Condition:</b>	N/A	

### 3.8..10 UC010: Manage Assignment Submission

*Table 3.12 Use Case Description for UC010 Manage Assignment Submission*

<b>Use Case Name:</b>	Manage Assignment Submission							
<b>Scenario:</b>	Students uploads, delete, and review their assignment submission.							
<b>Triggering Event:</b>	Student selects the "Add Submission" button.							
<b>Brief Description:</b>	Student manages their assignment submission.							
<b>Actors:</b>	Student							
<b>Related use cases</b>	N/A							
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. Student is logged into the system.</li> <li>2. Student is enrolled into the classroom.</li> <li>3. Assignment submission phase is active.</li> <li>4. Students add submission.</li> </ol>							
<b>Postconditions:</b>	Student successfully manages their assignment submission.							
<b>Flow of activities:</b>	<table border="1"> <thead> <tr> <th>Actor</th> <th>System</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>• Student upload assignment file.</li> <li>• Student submits assignment successfully.</li> </ul> </td><td> <ul style="list-style-type: none"> <li>• System verifies the file format.</li> <li>• System updates the assignment submission status.</li> <li>• System display PDF of the assignment submitted.</li> </ul> </td></tr> <tr> <td> <ul style="list-style-type: none"> <li>• Student deletes assignment file.</li> </ul> </td><td> <ul style="list-style-type: none"> <li>• System deletes the assignment.</li> <li>• System updates the assignment submission status.</li> </ul> </td></tr> </tbody> </table>	Actor	System	<ul style="list-style-type: none"> <li>• Student upload assignment file.</li> <li>• Student submits assignment successfully.</li> </ul>	<ul style="list-style-type: none"> <li>• System verifies the file format.</li> <li>• System updates the assignment submission status.</li> <li>• System display PDF of the assignment submitted.</li> </ul>	<ul style="list-style-type: none"> <li>• Student deletes assignment file.</li> </ul>	<ul style="list-style-type: none"> <li>• System deletes the assignment.</li> <li>• System updates the assignment submission status.</li> </ul>	
Actor	System							
<ul style="list-style-type: none"> <li>• Student upload assignment file.</li> <li>• Student submits assignment successfully.</li> </ul>	<ul style="list-style-type: none"> <li>• System verifies the file format.</li> <li>• System updates the assignment submission status.</li> <li>• System display PDF of the assignment submitted.</li> </ul>							
<ul style="list-style-type: none"> <li>• Student deletes assignment file.</li> </ul>	<ul style="list-style-type: none"> <li>• System deletes the assignment.</li> <li>• System updates the assignment submission status.</li> </ul>							
<b>Exception Condition:</b>	<p>Assignment submission phase in not active</p> <ul style="list-style-type: none"> <li>- Students will not be able to manage the submission.</li> </ul> <p>No assignment is uploaded.</p> <ul style="list-style-type: none"> <li>- Students will not be able to delete assignment.</li> </ul>							

### 3.8..11 UC011: View Grade

*Table 3.13 Use Case Description for UC011 View Grade*

<b>Use Case Name:</b>	View Grade					
<b>Scenario:</b>	Student views the grade for their assignment after the instructor returns the grading.					
<b>Triggering Event:</b>	Student selects the specific assignment.					
<b>Brief Description:</b>	Student views their grades.					
<b>Actors:</b>	Student					
<b>Related use cases</b>	N/A					
<b>Preconditions:</b>	1. Student is logged into the system. 2. Grades are available for viewing.					
<b>Postconditions:</b>	Student successfully views their grades.					
<b>Flow of activities:</b>	<table border="1"> <thead> <tr> <th>Actor</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>• Student selects the specific assignment</td> <td>• System displays the submission and corresponding grades.</td> </tr> </tbody> </table>	Actor	System	• Student selects the specific assignment	• System displays the submission and corresponding grades.	
Actor	System					
• Student selects the specific assignment	• System displays the submission and corresponding grades.					
<b>Exception Condition:</b>	N/A					

### 3.8..12 UC012: Review Feedback

*Table 3.14 Use Case Description for UC012 Review Feedback*

<b>Use Case Name:</b>	Review Feedback	
<b>Scenario:</b>	Student review the feedback for their assignment after the instructor returns the feedback.	
<b>Triggering Event:</b>	Student selects the specific assignment and click on “View Feedback”	
<b>Brief Description:</b>	Students reviews feedback given by instructors.	
<b>Actors:</b>	Student	
<b>Related use cases</b>	N/A	
<b>Preconditions:</b>	1. Student is logged into the system. 2. Feedback is available for reviewing. 3. Feedback is available for reviewing and the assignment is graded	
<b>Postconditions:</b>	Student can see individual assignment feedback.	
<b>Flow of activities:</b>	<b>Actor</b> <ul style="list-style-type: none"> <li>• Student views feedback</li> </ul>	<b>System</b> <ul style="list-style-type: none"> <li>• System displays the submission and respective feedback.</li> </ul>
<b>Exception Condition:</b>	Student had not submitted the assignment. - System will not display the feedback	

### 3.8..13 UC013: Provide Feedback

*Table 3.15 Use Case Description for UC013 Provide Feedback*

<b>Use Case Name:</b>	Provide Feedback							
<b>Scenario:</b>	Instructor provides feedback on student assignment.							
<b>Triggering Event:</b>	Instructor highlight a text and click on the “Add Note” icon.							
<b>Brief Description:</b>	The instructor highlights specific text and provides feedback on a student assignment.							
<b>Actors:</b>	Instructor							
<b>Related use cases</b>	UC014: Generate Feedback Suggestions UC015: Perform Lexical Analysis							
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. Instructor is logged into the system.</li> <li>2. Student submission exists.</li> <li>3. Instructor is grading the assignment.</li> </ol>							
<b>Postconditions:</b>	Feedback is successfully provided							
<b>Flow of activities:</b>	<table border="1"> <thead> <tr> <th>Actor</th> <th>System</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>• Instructor highlights text</li> <li>• Instructor adds note.</li> <li>• Instructor enters the feedback.</li> <li>• Instructor adds the feedback successfully.</li> </ul> </td> <td> <ul style="list-style-type: none"> <li>• System updates the feedback for the assignment.</li> <li>• System saves the feedback in database.</li> </ul> </td> </tr> <tr> <td> <ul style="list-style-type: none"> <li>• Instructor updates feedback.</li> <li>• Instructor enters new feedback.</li> <li>• Instructor updates the feedback successfully.</li> </ul> </td> <td></td> </tr> </tbody> </table>	Actor	System	<ul style="list-style-type: none"> <li>• Instructor highlights text</li> <li>• Instructor adds note.</li> <li>• Instructor enters the feedback.</li> <li>• Instructor adds the feedback successfully.</li> </ul>	<ul style="list-style-type: none"> <li>• System updates the feedback for the assignment.</li> <li>• System saves the feedback in database.</li> </ul>	<ul style="list-style-type: none"> <li>• Instructor updates feedback.</li> <li>• Instructor enters new feedback.</li> <li>• Instructor updates the feedback successfully.</li> </ul>		
Actor	System							
<ul style="list-style-type: none"> <li>• Instructor highlights text</li> <li>• Instructor adds note.</li> <li>• Instructor enters the feedback.</li> <li>• Instructor adds the feedback successfully.</li> </ul>	<ul style="list-style-type: none"> <li>• System updates the feedback for the assignment.</li> <li>• System saves the feedback in database.</li> </ul>							
<ul style="list-style-type: none"> <li>• Instructor updates feedback.</li> <li>• Instructor enters new feedback.</li> <li>• Instructor updates the feedback successfully.</li> </ul>								
<b>Exception Condition:</b>	N/A							

### 3.8..14 UC014 Generate Feedback Suggestions

*Table 3.16 Use Case Description for UC014 Generate Feedback Suggestions*

<b>Use Case Name:</b>	Generate Feedback Suggestions	
<b>Scenario:</b>	The system generates feedback suggestions for the instructor based on the historical feedback.	
<b>Triggering Event:</b>	Instructor highlights on a text.	
<b>Brief Description:</b>	Instructors receive feedback suggestions based on history when similar text is highlighted.	
<b>Actors:</b>	Instructor	
<b>Related use cases</b>	UC013: Provide Feedback	
<b>Preconditions:</b>	1. Instructor is logged into the system. 2. Instructor is providing feedback on a specific assignment.	
<b>Postconditions:</b>	Instructor can view and use generated feedback suggestions.	
<b>Flow of activities:</b>	<b>Actor</b> <ul style="list-style-type: none"> <li>• Instructor highlights the text.</li> <li>• Instructor adds note.</li> <li>• Instructors views the top 3 generated feedback.</li> <li>• Instructor edits the feedback.</li> </ul>	<b>System</b> <ul style="list-style-type: none"> <li>• System extracts the text.</li> <li>• System performs similarity matching with the historical text.</li> <li>• System generate top 3 feedback suggestion.</li> </ul>
<b>Exception Condition:</b>	No feedback history: - System will not generate any feedback suggestions	

### 3.8..15 UC015: Perform Lexical Analysis

*Table 3.17 Use Case Description for UC015 Perform Lexical Analysis*

<b>Use Case Name:</b>	Perform Lexical Analysis					
<b>Scenario:</b>	The system performs lexical analysis on assignment files.					
<b>Triggering Event:</b>	Instructors review the lexical analysis					
<b>Brief Description:</b>	System extracts text from assignments and performs lexical analysis (number of spelling and grammatical errors).					
<b>Actors:</b>	Instructor					
<b>Related use cases</b>	UC013: Provide Feedback					
<b>Preconditions:</b>	The system must have access to the assignment files.					
<b>Postconditions:</b>	Instructor can view the statistic of lexical analysis for each student					
<b>Flow of activities:</b>	<table border="1"> <thead> <tr> <th>Actor</th> <th>System</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>Instructor selects a student assignment.</li> <li>Instructor views the lexical analysis of the assignment.</li> </ul> </td> <td> <ul style="list-style-type: none"> <li>System loads the student assignment.</li> <li>System performs lexical analysis.</li> <li>System displays the percentage of lexical analysis.</li> </ul> </td> </tr> </tbody> </table>	Actor	System	<ul style="list-style-type: none"> <li>Instructor selects a student assignment.</li> <li>Instructor views the lexical analysis of the assignment.</li> </ul>	<ul style="list-style-type: none"> <li>System loads the student assignment.</li> <li>System performs lexical analysis.</li> <li>System displays the percentage of lexical analysis.</li> </ul>	
Actor	System					
<ul style="list-style-type: none"> <li>Instructor selects a student assignment.</li> <li>Instructor views the lexical analysis of the assignment.</li> </ul>	<ul style="list-style-type: none"> <li>System loads the student assignment.</li> <li>System performs lexical analysis.</li> <li>System displays the percentage of lexical analysis.</li> </ul>					
<b>Exception Condition:</b>	N/A					

### 3.8..16 UC016: Grade Student Assignment

*Table 3.18 Use Case Description for UC016 Grade Student Assignment*

<b>Use Case Name:</b>	Grade Student Assignment	
<b>Scenario:</b>	Instructor grades student assignment	
<b>Triggering Event:</b>	Instructor is grading student assignment.	
<b>Brief Description:</b>	Instructor enters scores according to grading rubrics.	
<b>Actors:</b>	Instructor	
<b>Related use cases</b>	N/A	
<b>Preconditions:</b>	1. Instructor is logged into the system. 2. Grading rubric is available for the assignment. 3. Student submission exists.	
<b>Postconditions:</b>	Assignment is successfully graded.	
<b>Flow of activities:</b>	<b>Actor</b> <ul style="list-style-type: none"> <li>• Instructor enters marks according to the grading rubrics for each assignment.</li> <li>• Instructor saves grading.</li> </ul>	<b>System</b> <ul style="list-style-type: none"> <li>• System calculates the total marks.</li> <li>• System updates the marks for the assignment.</li> </ul>
<b>Exception Condition:</b>	N/A	

## 3.8..17 UC017: View Dashboard

*Table 3.19 Use Case Description for UC017 View Dashboard*

<b>Use Case Name:</b>	View Dashboard	
<b>Scenario:</b>	Instructor wants to view the dashboard of grading process.	
<b>Triggering Event:</b>	Instructor selects the "View Submission" button	
<b>Brief Description:</b>	Instructors may review the dashboard of grading progress.	
<b>Actors:</b>	Instructor	
<b>Related use cases</b>	N/A	
<b>Preconditions:</b>	1. Instructor is logged into the system. 2. Assignment exists	
<b>Postconditions:</b>	Dashboard is successfully displayed.	
<b>Flow of activities:</b>	<b>Actor</b> <ul style="list-style-type: none"> <li>• Instructor views submission</li> <li>• Instructor views the submission status and grading process.</li> </ul>	<b>System</b> <ul style="list-style-type: none"> <li>• System displays the grading progress dashboard including the number of submissions, and number of assignments graded.</li> </ul>
<b>Exception Condition:</b>	N/A	

### 3.8..18 UC018: Return Graded Assignment

*Table 3.20 Use Case Description for UC018 Return Graded Assignment*

<b>Use Case Name:</b>	Return Graded Assignment							
<b>Scenario:</b>	Instructor wants to return the feedback and grades students to review.							
<b>Triggering Event:</b>	Instructor selects the "Return Submission" button.							
<b>Brief Description:</b>	Instructor returns the feedback and grades for students to review once instructor is done with all grading.							
<b>Actors:</b>	Instructor							
<b>Related use cases</b>	N/A							
<b>Preconditions:</b>	<ol style="list-style-type: none"> <li>1. Instructor is logged into the system.</li> <li>2. Assignments are graded.</li> </ol>							
<b>Postconditions:</b>	Graded assignments are successfully made available to students.							
<b>Flow of activities:</b>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding-bottom: 5px;">Actor</th> <th style="text-align: left; padding-bottom: 5px;">System</th> </tr> </thead> <tbody> <tr> <td style="padding-top: 5px;">• Instructor returns submission</td> <td style="padding-top: 5px;">• System enables all students in the classroom to view the feedback and grade of their assignment.</td> </tr> <tr> <td style="padding-top: 5px;">• Instructor reverts submission</td> <td style="padding-top: 5px;">• System disables all students in the classroom to view the feedback and grade of their assignment.</td> </tr> </tbody> </table>		Actor	System	• Instructor returns submission	• System enables all students in the classroom to view the feedback and grade of their assignment.	• Instructor reverts submission	• System disables all students in the classroom to view the feedback and grade of their assignment.
Actor	System							
• Instructor returns submission	• System enables all students in the classroom to view the feedback and grade of their assignment.							
• Instructor reverts submission	• System disables all students in the classroom to view the feedback and grade of their assignment.							
<b>Exception Condition:</b>	N/A							

### 3.8.3. Sequence Diagram

Figure 3.3 Sequence Diagram for Login Account

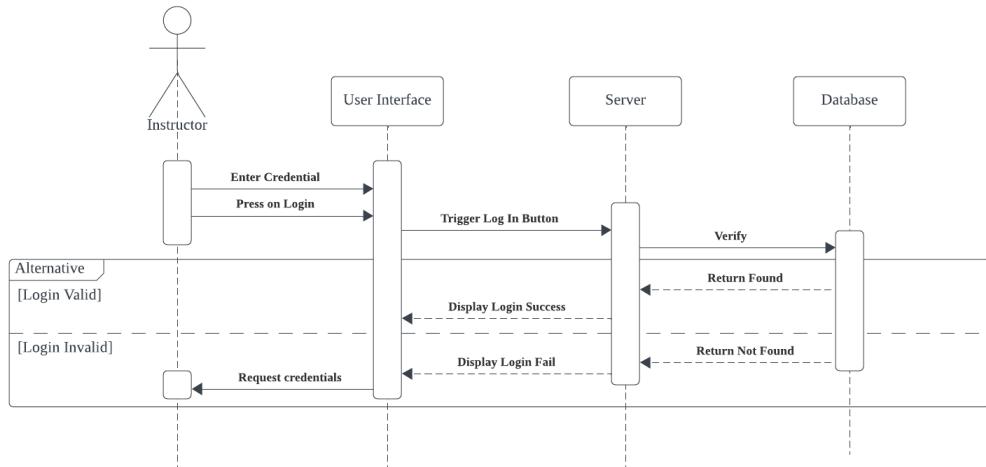


Figure 3.4 Sequence Diagram for Register Account

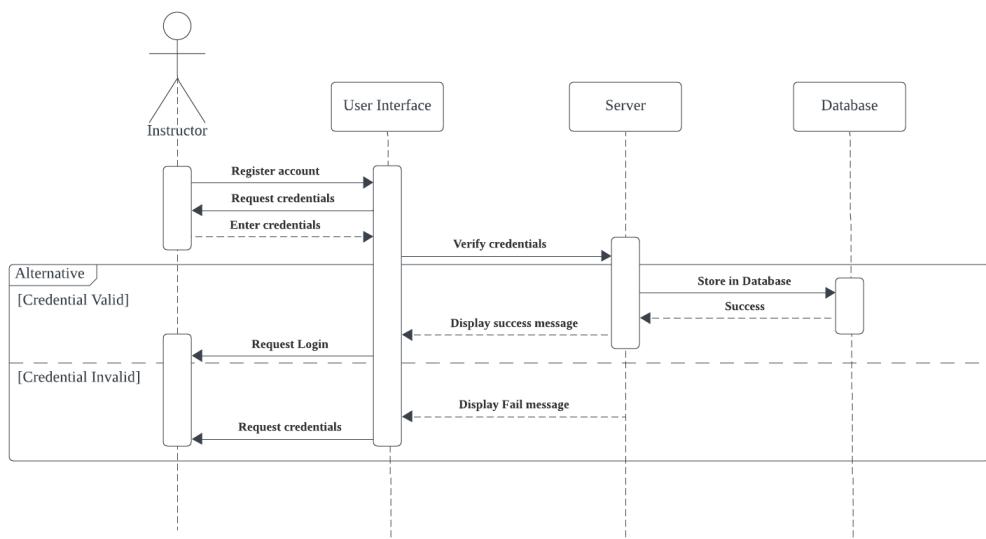


Figure 3.5 Sequence Diagram for Manage Classroom

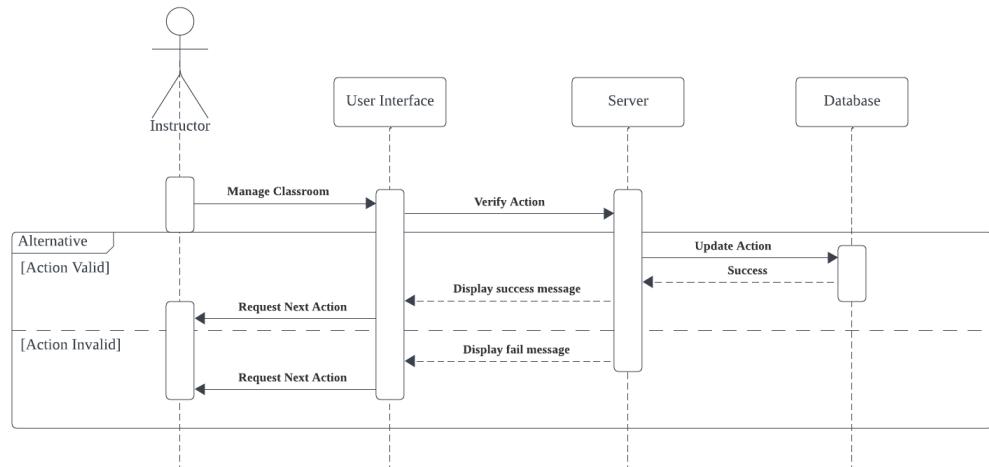


Figure 3.6 Sequence Diagram for Manage Student Enrolment

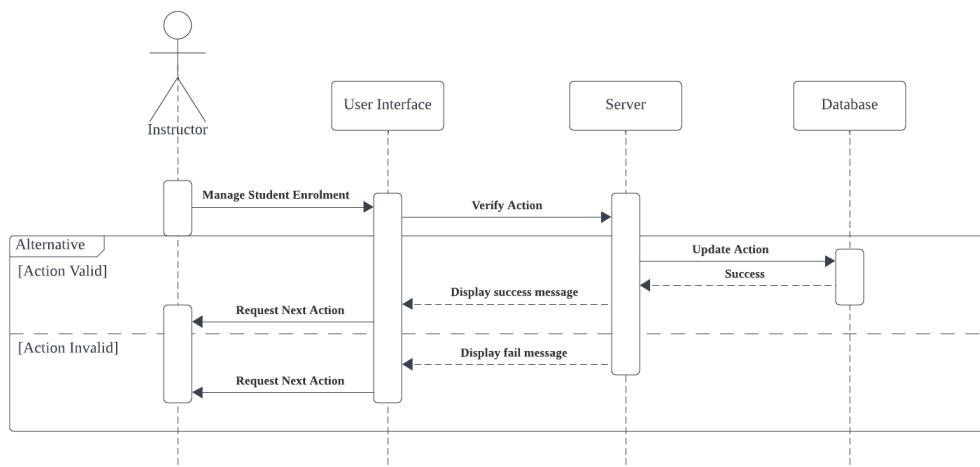


Figure 3.7 Sequence Diagram for Manage Assignment

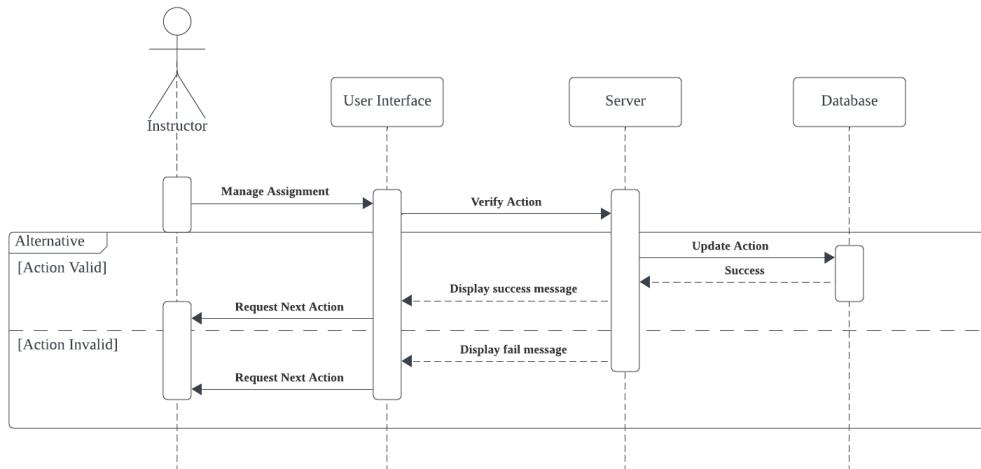


Figure 3.8 Sequence Diagram for Set Assignment Attributes

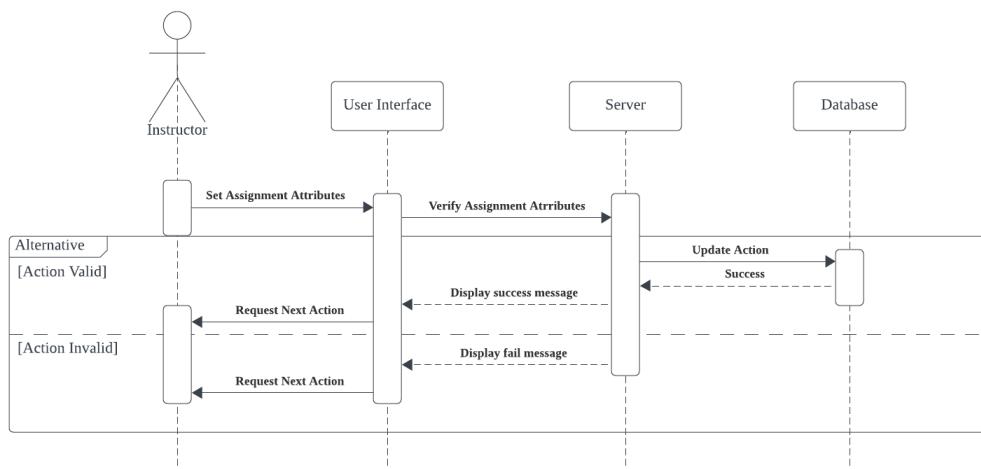


Figure 3.9 Sequence Diagram for Manage Grading Rubric

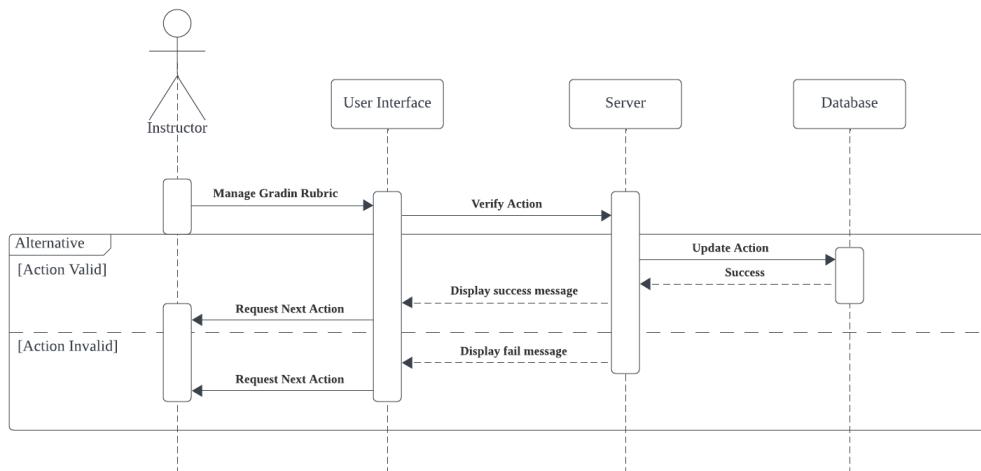


Figure 3.10 Sequence Diagram for Manage Student Submissions

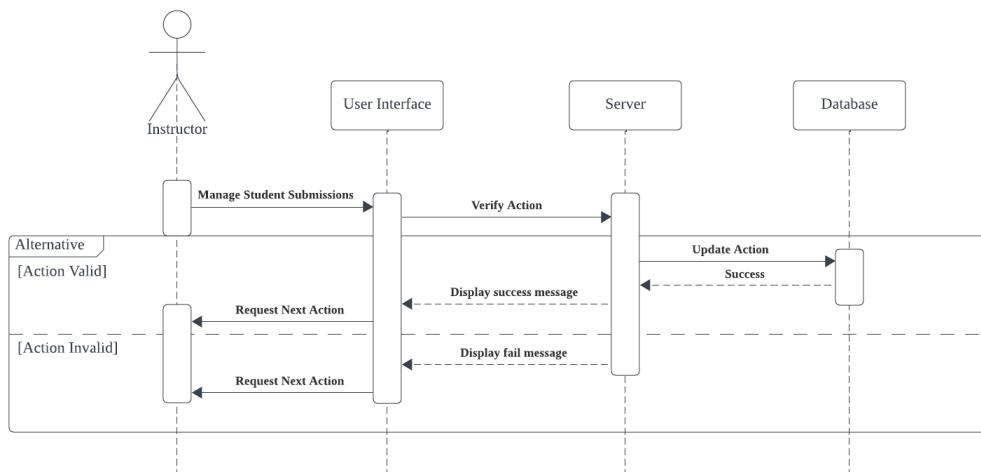


Figure 3.11 Sequence Diagram for Download Assignment

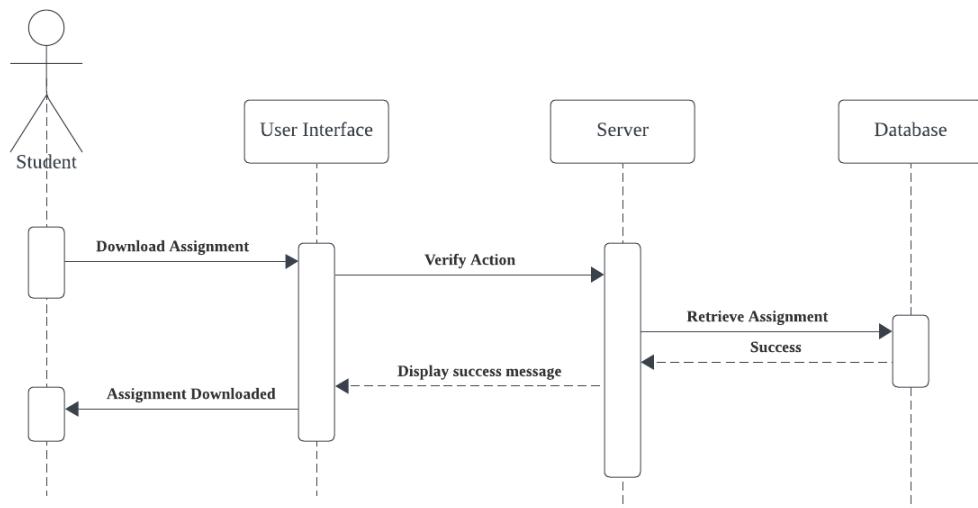


Figure 3.12 Sequence Diagram for Manage Assignment Submission

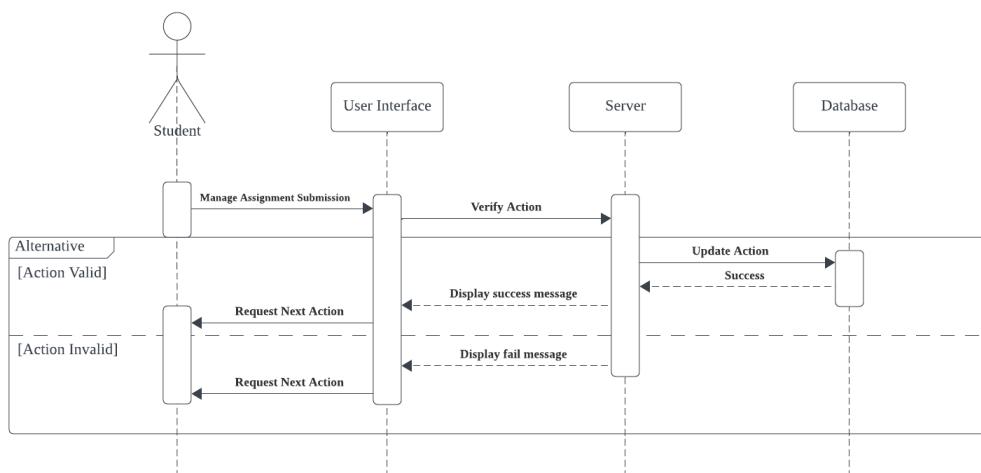


Figure 3.13 Sequence Diagram for View Grade

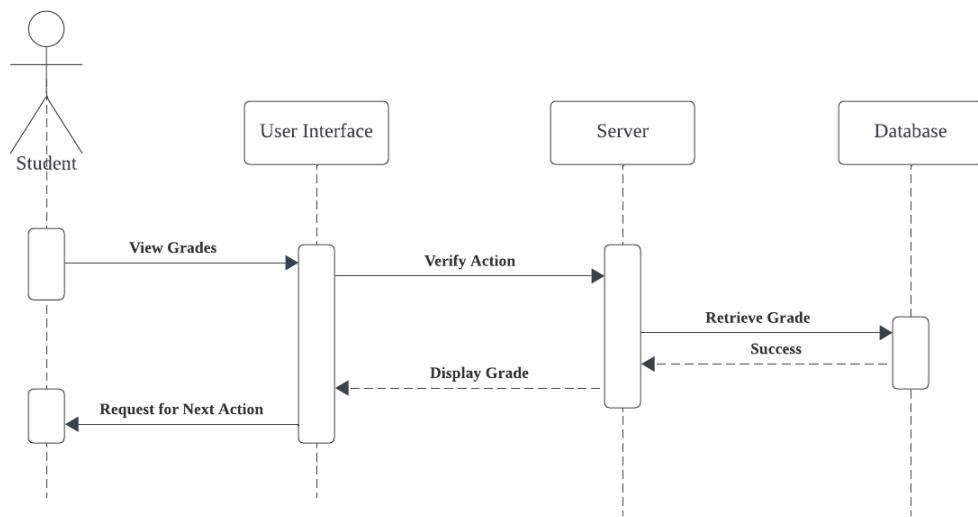


Figure 3.14 Sequence Diagram for Review Feedback

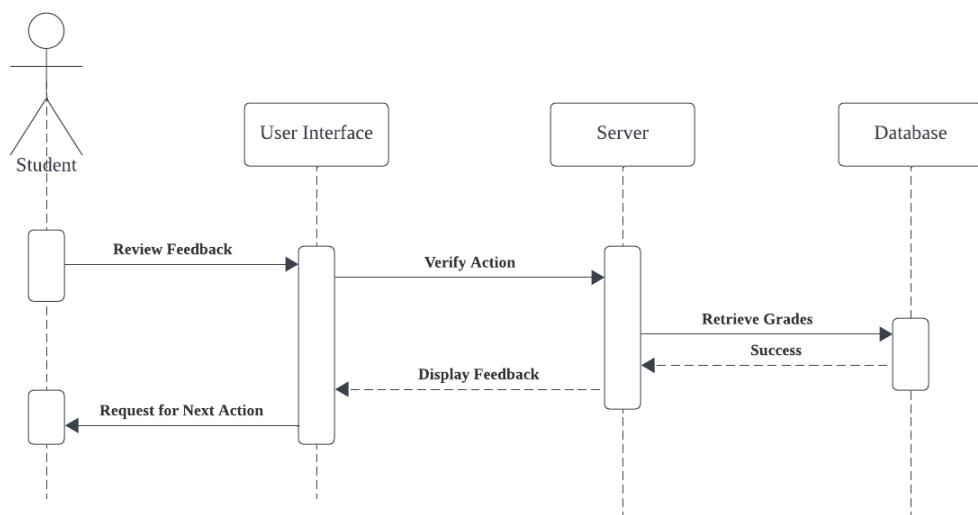


Figure 3.15 Sequence Diagram for Provide Feedback, Generate Feedback Suggestions

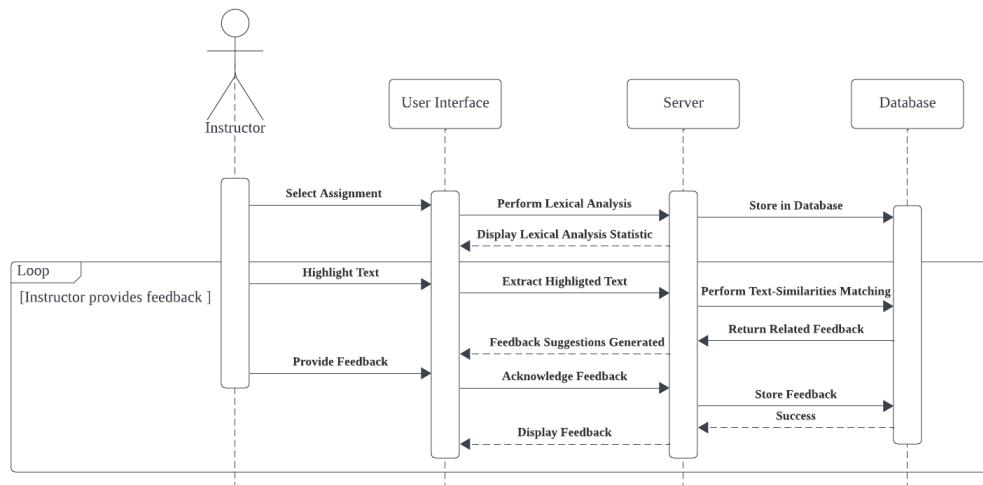


Figure 3.16 Sequence Diagram for Grade Student Assignment

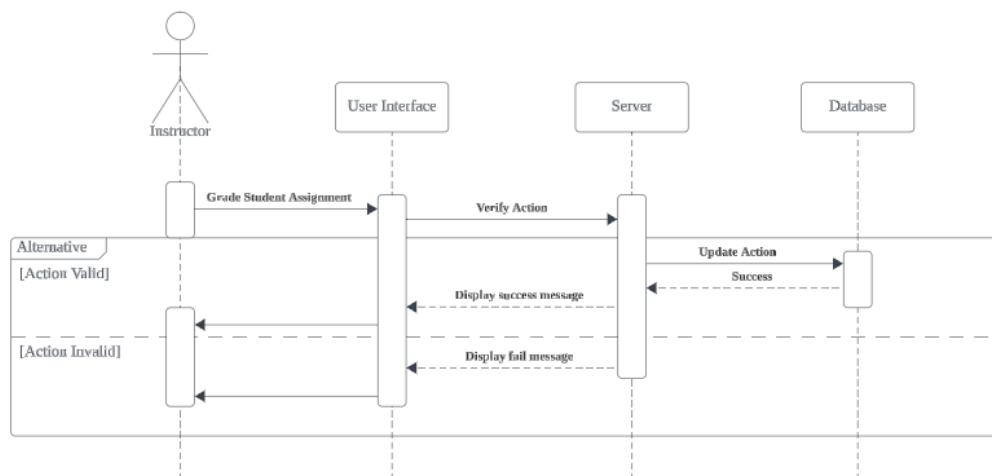


Figure 3.17 Sequence Diagram for View Dashboard

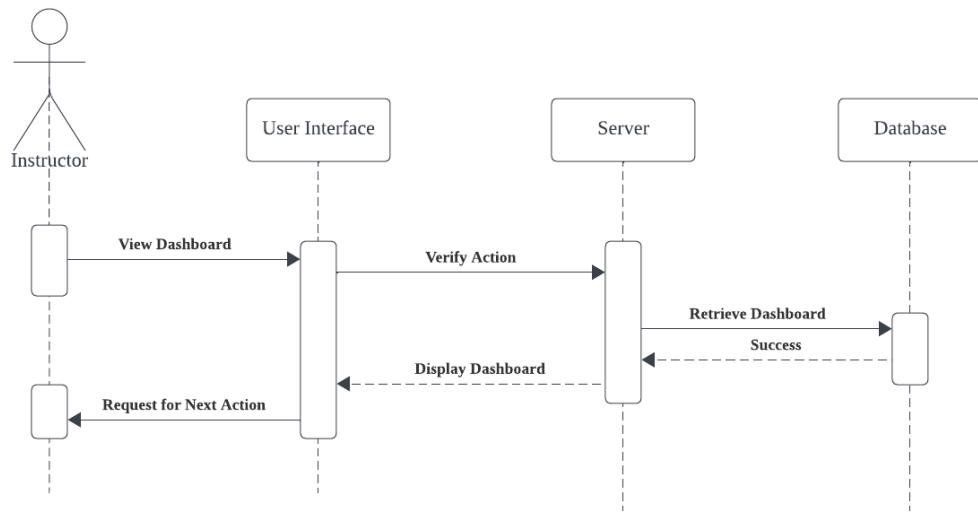
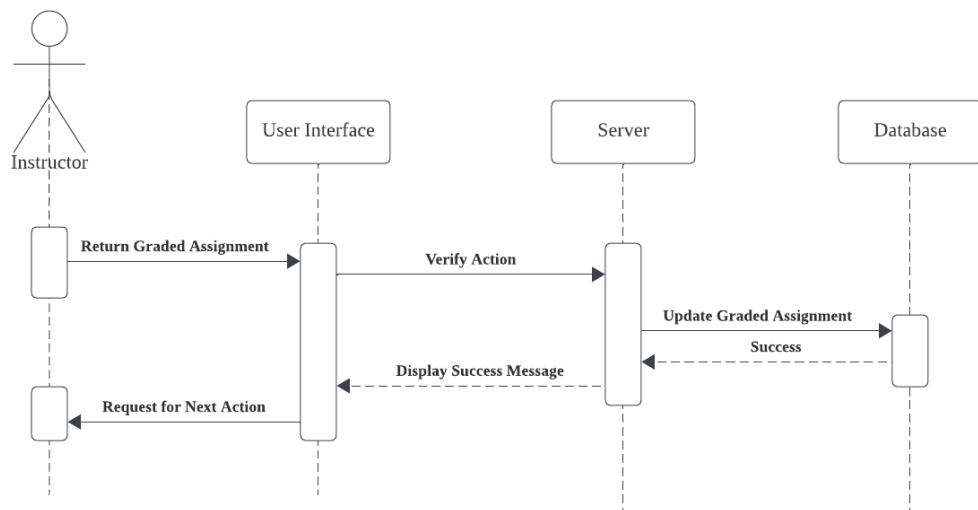
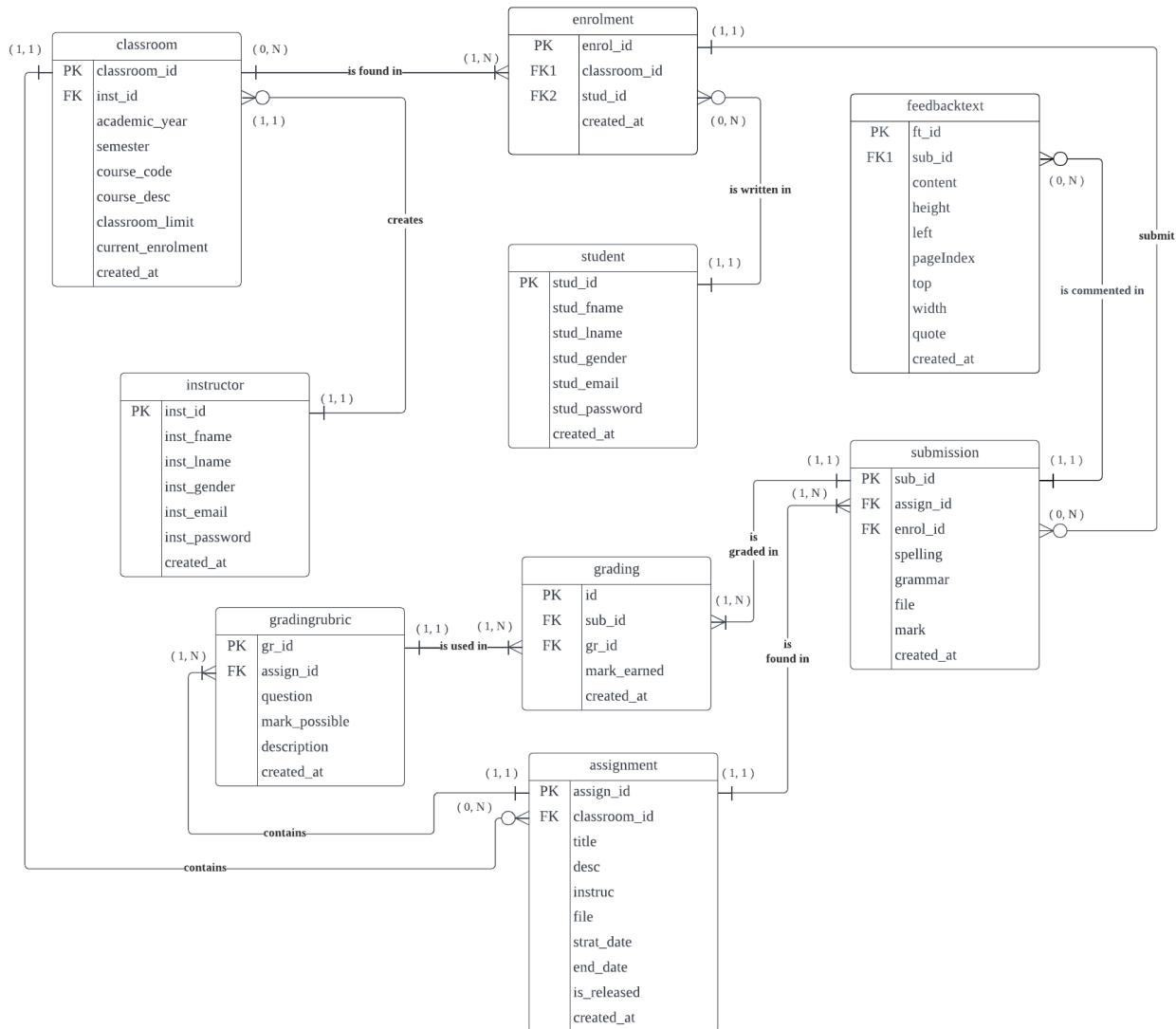


Figure 3.18 Sequence Diagram for Return Graded Assignment



### 3.8.4. Entity Relationship Diagram (ERD)

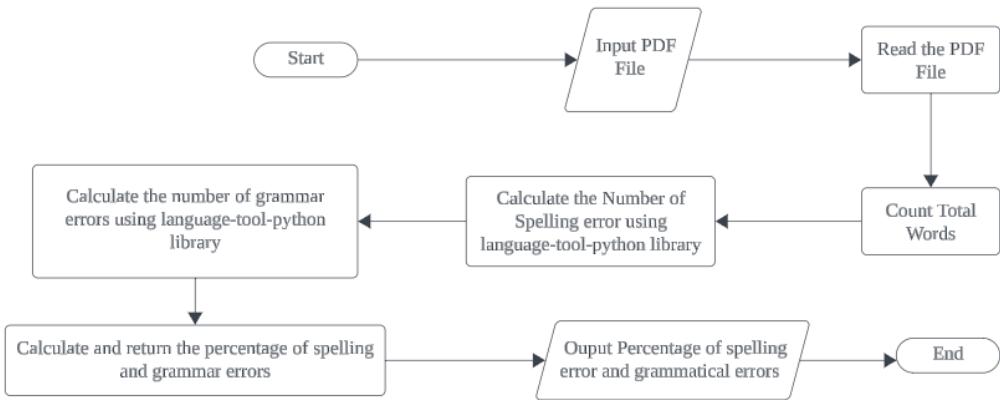
Figure 3.19 Entity Relationship Diagram (ERD)



## 3.9. Intelligent Components

### 3.9.1. Lexical Analysis (Spelling & Grammar Error)

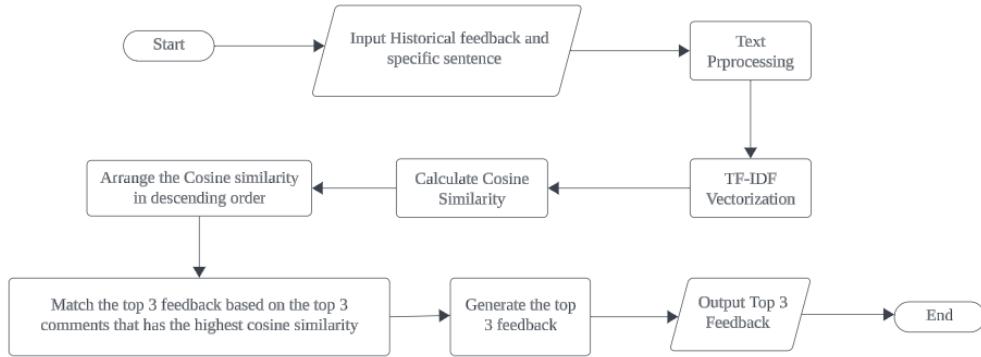
Figure 3.20 Flow of language-tool-python library.



The flow commences when there is a PDF input file, the PDF file will be read, and the text will be extracted to count the total words. The language-tool-python library is used to facilitate spelling and grammar checking using the LanguageTool API. The process begins with the importation of the library and the creation of a LanguageTool instance, specifying the desired language. The input text is then checked for spelling and grammar issues, producing a list of matches. Each match represents a detected issue with details such as rule ID, error message, and suggested corrections. The percentage of spelling and grammar errors will be calculated as output, concluding the spelling and grammar checking process. The language-tool-python library serves as a bridge between Python applications and the LanguageTool service, allowing for seamless integration of grammar and style checking capabilities [12].

### 3.9.2. Feedback Suggestion Generation

*Figure 3.21 Flow of Feedback Suggestion Generation*



The process commences with the retrieval of historical text stored in database. Once retrieved, both the historical and highlighted text will undergo text preprocessing like tokenization, removing stop words and punctuation, and converting the text into lowercase. Next, the TF-IDF Vectorization is applied using the TfIdfVectorizer, to convert them into numerical representations. The cosine similarity between the TF-IDF vectors of the historical and highlighted text is then calculated. The comments are sorted in the descending order of the cosine similarity. The respective feedback of the top 3 comments will be generated [13].

### 3.10.Data Source

The dataset for this project is derived from a collection of past written assignments provided by the instructor. This dataset includes two distinct assignment questions: one on classification and the other on clustering.

The first assignment revolves around the concept of classification, a fundamental topic in data science and machine learning. A total of 39 sets of student assignments have been collected for this assignment. Within these 39 sets, 12 commonly used comments have been identified. For detailed analysis and application within the system, a subset of 10 sets of student assignments has been selected. These 10 sets contain a total of at least 74 historical comments, providing a rich source of data for understanding feedback patterns and student performance.

The second assignment deals with clustering techniques, another essential topic in data science and machine learning. A total of 10 sets of student assignments have been gathered for this assignment. While specific comments have not been detailed for this assignment, the feedback collected will still be integral to shaping the feedback mechanisms within the platform.

By using this dataset, the web application can provide personalized and detailed feedback to students based on historical comments and instructor evaluations.

### **3.11. Technology Deployed**

#### **3.11.1. Hardware**

1. RAM : 8 GB
2. GPU : 2.0 GHz

#### **3.11.2. Software**

1. OS system : Windows
2. Browser : Google Chrome
3. Development IDE : Visual Studio Code
4. Version Control : GitHub
5. Database : Supabase
6. Diagramming Application : Lucidchart
7. Design & Prototyping tools : Figma

#### **3.11.3. Programming Languages/Tools**

1. Programming Language : HTML, CSS, JavaScript, Python
2. Frameworks : React.js, as the front-end framework.  
Node.js, as the backend framework.
3. Library : scikit-learn, language-tool-python

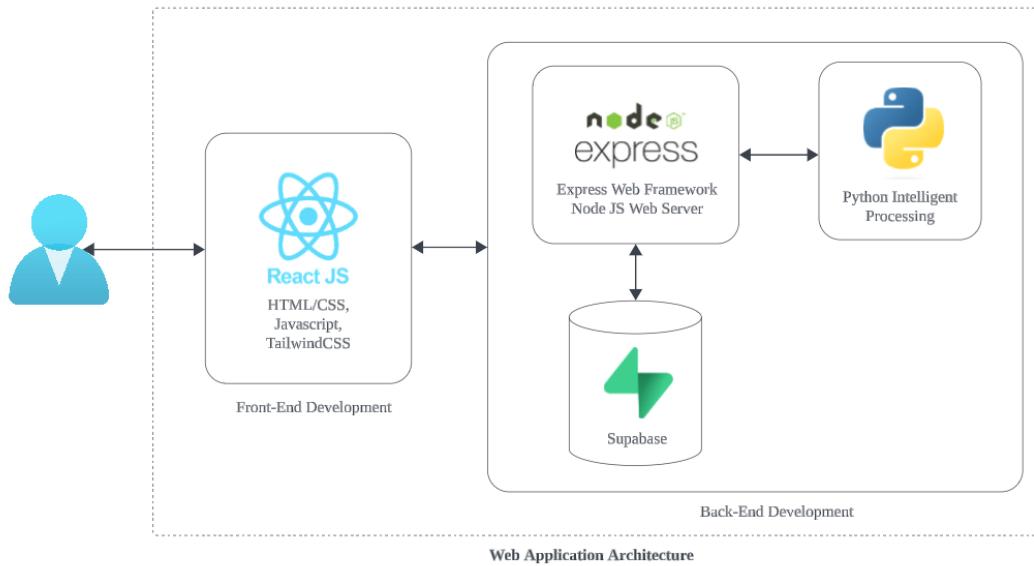
#### **3.11.4. Chosen algorithm**

1. language-tool-python library for spell-checking and grammar-checking using rule-based.
2. Text-similarity matching using TF-IDF Vectorization and Cosine Similarity.

## CHAPTER 4: SYSTEM DESIGN & IMPLEMENTATION

### 4.1. System Architecture

*Figure 4.1 System Architecture*



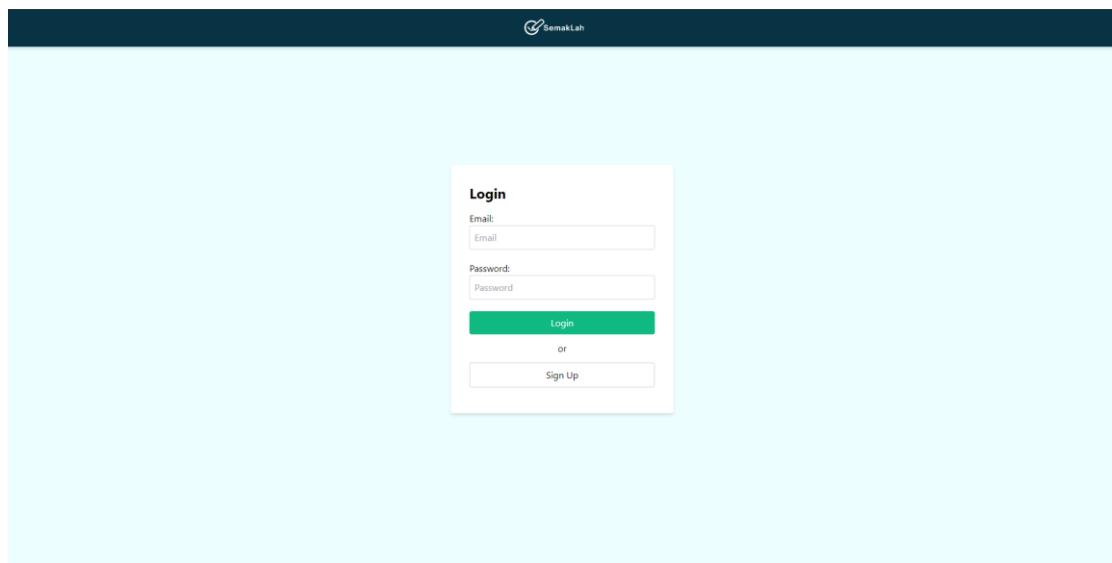
SemakLah system architecture diagram illustrates a web application designed for managing and evaluating technical written assignments at the university level. The front-end is developed using React JS, employing HTML/CSS, JavaScript, and TailwindCSS to create a dynamic and user-friendly interface for instructors and students. The back end is powered by Node.js with the Express framework, which handles server-side operations and API requests. Data storage and management are facilitated by Supabase, an open-source backend-as-a-service platform. Additionally, the back end incorporates Python for intelligent processing tasks, specifically lexical analysis, and text-similarity matching for feedback generation. Users interact with the application through the front-end, which communicates with the back-end server to manage data and process assignments, providing a seamless and efficient experience.

## 4.2. User Interface Design

### 4.2.1. General

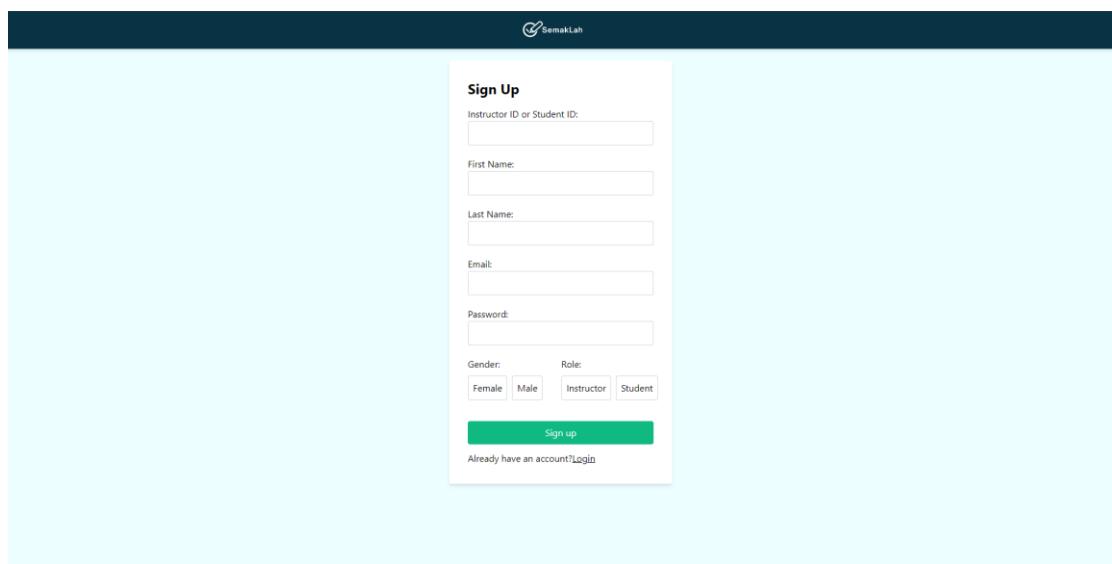
#### 4.2..1 Login

Figure 4.2 User Login Page



#### 4.2..2 Sign Up

Figure 4.3 User Sign Up Page

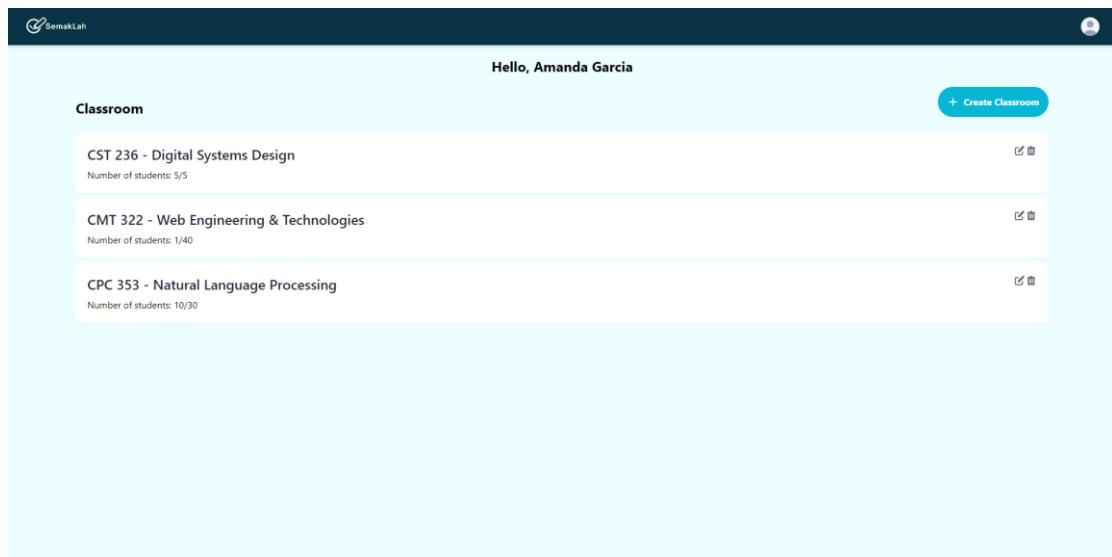


Users share the same login and sign-up page. It is differentiated from the role that they select. Every user is compulsory to login to their respective account to access their homepage and perform different actions.

## 4.2.2. Instructors

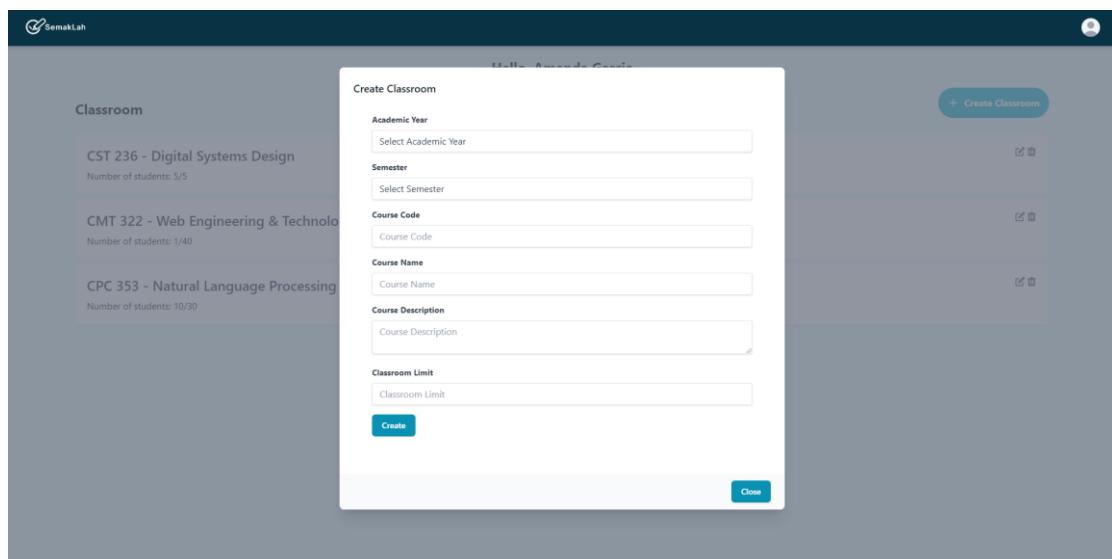
### 4.2..1 Homepage

Figure 4.4 Instructor Homepage



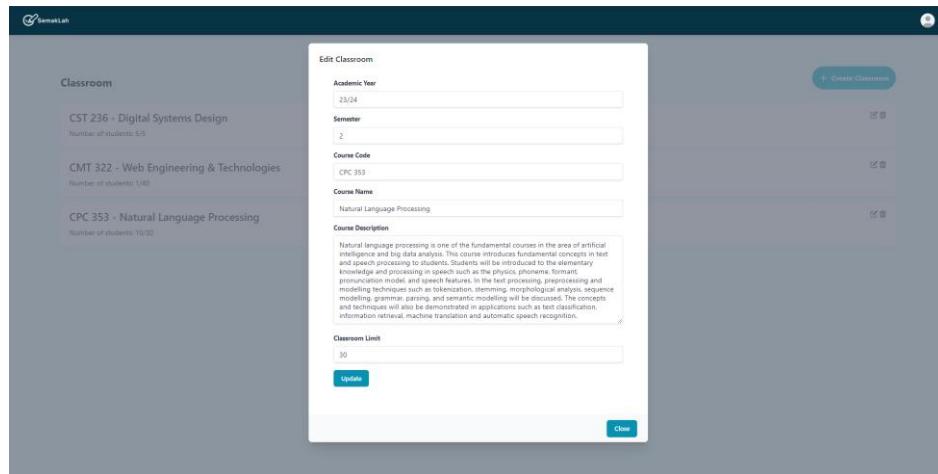
### 4.2..2 Manage Classroom – Create Classroom

Figure 4.5 Instructor create classroom.



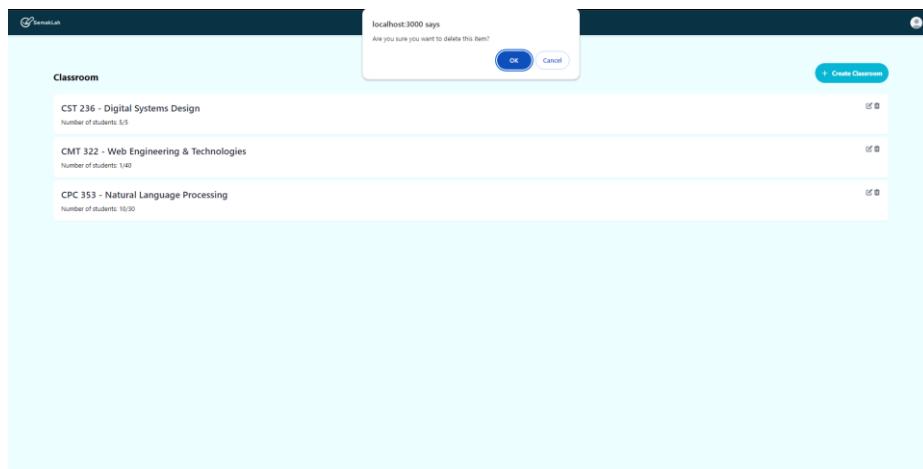
#### 4.2..3 Manage Classroom – Update Classroom

*Figure 4.6 Instructor updates classroom.*



#### 4.2..4 Manage Classroom – Delete Classroom

*Figure 4.7 Instructor deletes classroom.*



Instructor homepage contains the list of classrooms and the ability to manage classroom which includes create, edit, and delete a classroom. While creating a classroom, after the instructor clicks on **+ Create Classroom** button, the instructor is required to fill in all the necessary information which includes the academic year, semester, course code, course name, course description and the classroom limit. While updating a classroom, the instructor will be able to edit and update the details of the classroom after clicking on the icon. While deleting a classroom, the instructor can simply click on the icon and they will be prompted to confirm deletion to avoid any unintended deletion from occurring.

#### 4.2..5 Classroom Details Page

Figure 4.8 Instructor classroom details page.

The screenshot shows the 'CPC 353 - Natural Language Processing' classroom details page. At the top, there is a 'Back' button and a user profile icon. Below the title, the 'Class Token' is displayed as b0af00b4-90c7-4993-ac30-9ac9bf2fb69. A descriptive text block explains the course content, mentioning artificial intelligence, big data analysis, and various speech processing techniques like tokenization, stemming, and semantic modeling. The page is divided into sections: 'Assignment List' and 'Student List'. The 'Assignment List' section contains two entries: 'Assignment 1 Classification' (start date: 3/1/2024, end date: 5/30/2024) and 'Assignment 2: Clustering' (start date: 5/1/2024, end date: 6/1/2024). Each assignment entry includes a 'View Submission' button and icons for edit and delete. The 'Student List' section displays a table with columns: First Name, Last Name, Email, Student ID, and Remove. The table lists ten students with their respective details. A 'Create Assignment' button is located at the top right of the assignment list area.

First Name	Last Name	Email	Student ID	Remove
Amirah	Abdullah	amirah.abdullah@student.usm.my	150001	[trash]
Hafiz	Ibrahim	hafiz.ibrahim@student.usm.my	150002	[trash]
Nurul	Aziz	nurulaziz@student.usm.my	150003	[trash]
Muhammad	Hassan	mohammad.hassan@student.usm.my	150004	[trash]
Siti	Mohamad	siti.mohamad@student.usm.my	150005	[trash]
Azim	Wong	azim.wong@student.usm.my	150006	[trash]
Fatin	Tan	fatin.tan@student.usm.my	150007	[trash]
Aisyah	Lim	aisyah.lim@student.usm.my	150008	[trash]
Syahir	Lee	syahir.lee@student.usm.my	150009	[trash]
Amir	Ng	amir.ng@student.usm.my	150010	[trash]

#### 4.2..6 Manage Assignment – Create Assignment

Figure 4.9 Instructor creates assignment.

The screenshot shows the 'Create Assignment' dialog box overlaid on the classroom details page. The dialog box has fields for 'Title' (Assignment 3), 'Description', 'Instructions', 'File' (Choose File, No file chosen), 'Start Date' (mm/dd/yyyy), and 'End Date' (mm/dd/yyyy). There are 'Create Assignment' and 'Close' buttons at the bottom. In the background, the 'Assignment List' and 'Student List' sections are visible. The 'Assignment List' shows 'Assignment 1 Classification' and 'Assignment 2: Clustering'. The 'Student List' table is identical to the one in Figure 4.8, listing ten students with their details. The 'Create Assignment' button is also present in the background assignment list area.

#### 4.2..7 Manage Assignment – Update Assignment

Figure 4.10 Instructor updates assignment.

The screenshot shows the assignment update interface for the CPC 353 - Natural Language Processing course. The main page displays the class token and a brief description of the course. On the left, there's an 'Assignment List' section showing two assignments: 'Assignment 1 Classification' and 'Assignment 2: Clustering'. The 'Assignment 1 Classification' card includes start and end dates. On the right, there's a 'Student List' table with columns for First Name, Last Name, Email, Student ID, and Remove button. A modal window titled 'Update Assignment' is open over the student list, containing fields for Title (Assignment 1 Classification), Description (Natural language processing is one of the fundamental courses in the area of artificial intelligence. It deals with the processing of speech such as the phonics, phoneme, formant, pronunciation model, and speech modelling will be discussed. The concepts and techniques will also be demonstrated in applications.), Instructions (Submit in PDF Format. Max 10 pages content.), and File (Choose File). There are also fields for Start Date (03/31/2024 12:00 AM) and End Date (05/30/2024 12:00 AM). At the bottom of the modal are 'Update Assignment' and 'Close' buttons.

#### 4.2..8 Manage Assignment – Delete Assignment

Figure 4.11 Instructor deletes assignment.

The screenshot shows the assignment delete confirmation dialog. It displays the message 'localhost:3000 says Are you sure you want to delete this item?' with 'OK' and 'Cancel' buttons. Below the dialog, the assignment list and student list tables are visible, identical to Figure 4.10. The assignment list shows 'Designing a Simple Digital Clock Using VHDL' with start and end dates. The student list table has the same structure as Figure 4.10.

#### 4.2..9 Grading Rubric

Figure 4.12 Instructor manages grading rubric

Question Number	Mark Possible	Descriptions
1	10	Complete this question with a correct answer

**Total Mark: 10**

#### 4.2..10 Grading Rubric – Add a new row.

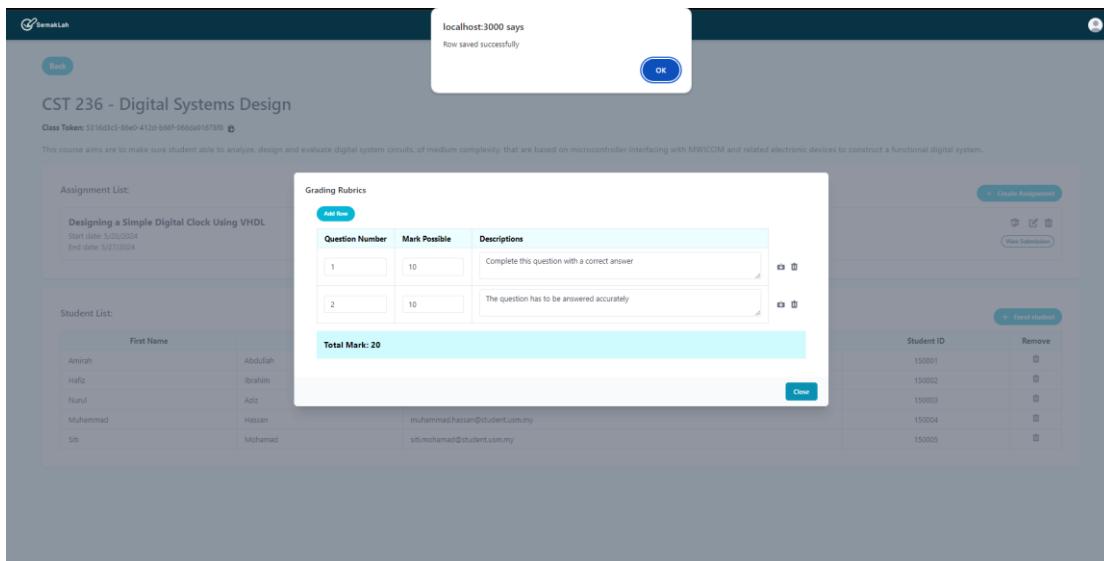
Figure 4.13 Instructor adds grading rubric

Question Number	Mark Possible	Descriptions

**Total Mark: 10**

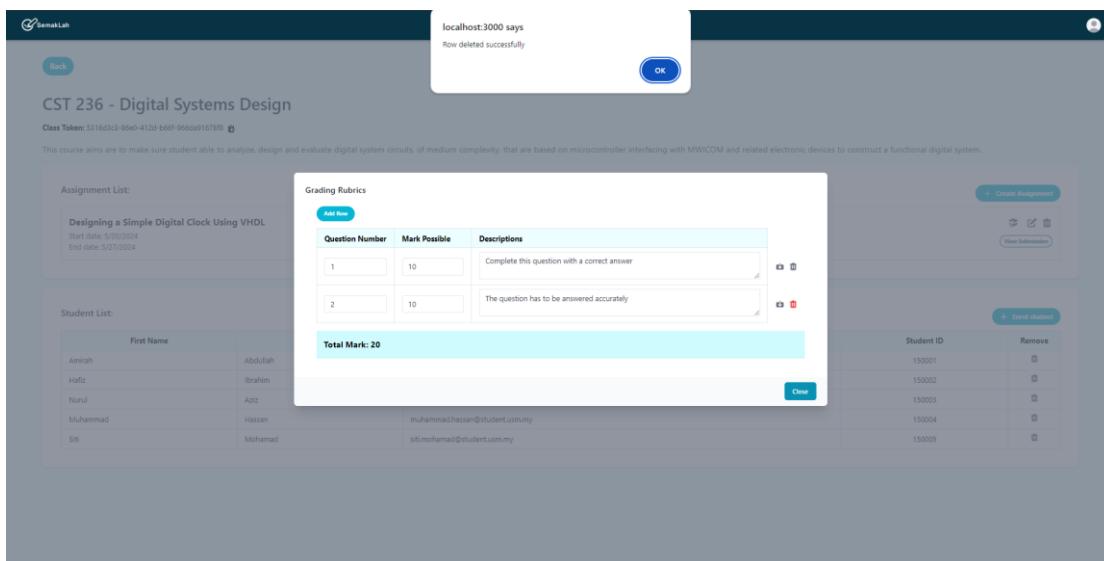
#### 4.2..11 Grading Rubric – Saving a new row.

Figure 4.14 Instructor saves grading rubric



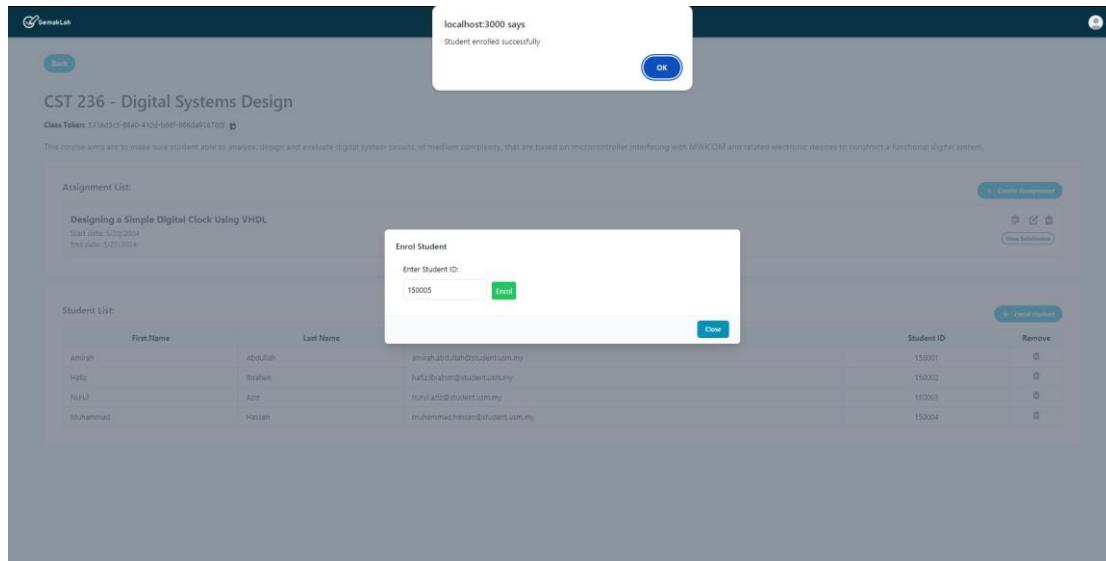
#### 4.2..12 Grading Rubric – Deleting a row.

Figure 4.15 Instructor deletes grading rubric.



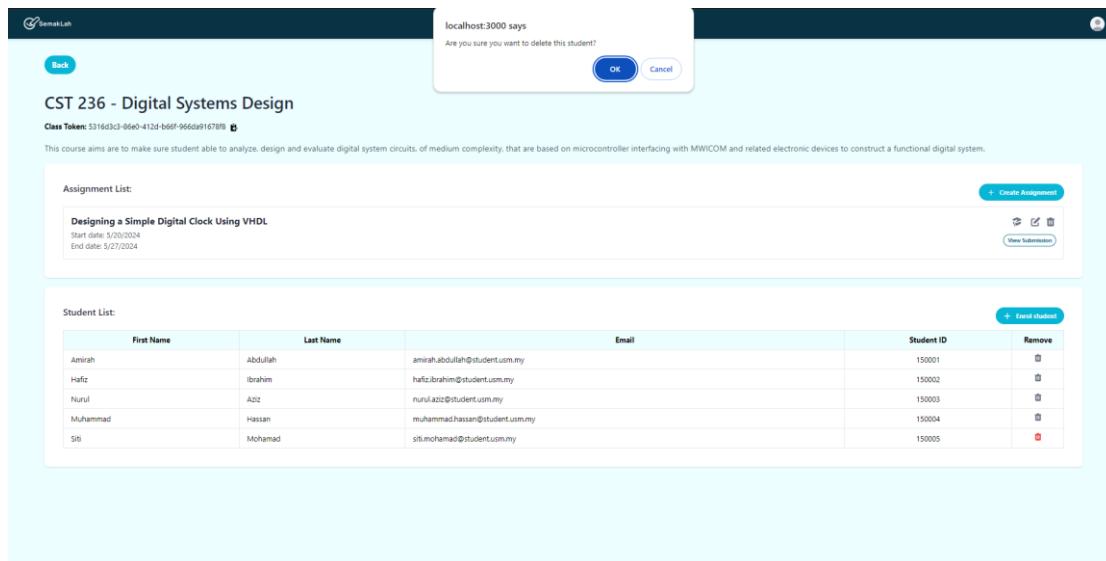
#### 4.2..13 Manage student enrolment – Enrol student.

*Figure 4.16 Instructor enrolls student.*



#### 4.2..14 Manage student enrolment – remove a student.

*Figure 4.17 Instructor removes student.*



When the instructor clicks on the specific classroom, they will be able to see the classroom details that include the class token, assignment list and student list. In this page, they can perform different action such as managing assignments which includes creating, editing, and deleting an assignment, managing the grading rubrics, and managing student enrolment.

While creating an assignment, after the instructor click on  button, the instructor is required to set the attributes of the assignments which includes title, description, instruction, uploading the file in PDF and selecting the start and end date of the assignment. After creating an assignment, the instructor can also update the assignment anytime by clicking on the  icon and update the attributes. If there is an existing assignment that an instructor would like to delete, they can click on the  icon and the system will prompt them to confirm their deletion.

Other than managing assignment, instructor should also manage the grading rubric by clicking on the  icon, and simply click on the  button to add a new row of grading rubric. After they have key in the information, they should click on  icon to save the row of grading rubric. If there is any existing row of grading rubric, instructor can perform deletion by clicking on the  icon.

Besides that, the instructor could also manage student enrolment. Instructor can enrol students into the classroom by clicking the , insert student's id and click on the  button to enrol. The alternative would be an instructor copying the class token and sending it to their students to join the classroom. The instructor could also remove a student by clicking on the  icon and confirm the deletion.

Instructor can proceed with viewing the details of the assignment by clicking on a specific assignment.

## 4.2..15 Assignment Details Page

*Figure 4.18 Instructor assignment details page*

The screenshot shows the assignment details page for 'Designing a Simple Digital Clock Using VHDL'. At the top, there is a back button and a user profile icon. The title of the assignment is 'Designing a Simple Digital Clock Using VHDL'. Below the title, there are two date fields: 'Start Date : 20/05/2024, 12:00:00 am' and 'End Date : 27/05/2024, 11:59:00 pm'. A detailed instruction text follows, describing the task of designing a digital clock using VHDL with specific requirements like counters and display drivers. A 'View Submissions' button is located at the bottom of this section.

Below the instruction, there is a large preview window showing the assignment details. The preview includes the assignment title, due date (2023/2024), released date (18 Apr 2024), report due date (10 May 2024), group (Mar 2 in a group), and a list of deliverables (I. Softcopy of Report, II. Source Code of Program). It also contains a brief description of the task, input requirements, and output format. A note about handing in a typed technical report is included.

At the bottom of the preview window, there is a table titled 'Grading Rubrics' with columns for 'Question Number', 'Mark Possible', and 'Descriptions'. One row is shown with a question number of 1, a mark possible of 10, and a description: 'Complete this question with a correct answer'. A total mark of 10 is displayed at the bottom of the rubrics table.

This page displays a summary of the assignments for instructors to see and they can edit the grading rubrics here as well. To view the submission, the instructor should click on the [View Submissions](#) button.

#### 4.2..16 View Submission Page

Figure 4.19 Instructor views submission page.

Submission Status	
Submission	10/10
Grading Progress	9/10

Submission List							
First Name	Last Name	Email Address	Student ID	Grading	Grade	Download	
Amirah	Abdullah	amirah.abdullah@student.usm.my	150001	<span>Graded</span>	52 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Hafiz	Ibrahim	hafiz.ibrahim@student.usm.my	150002	<span>Graded</span>	76 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Nurul	Aziz	nurul.aziz@student.usm.my	150003	<span>Graded</span>	68 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Muhammad	Hassan	muhammad.hassan@student.usm.my	150004	<span>Graded</span>	64 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Siti	Mohamed	siti.mohamed@student.usm.my	150005	<span>Graded</span>	52 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Azim	Wong	azim.wong@student.usm.my	150006	<span>Graded</span>	48 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Fatin	Tan	fatin.tan@student.usm.my	150007	<span>Graded</span>	50 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Aisyah	Lim	aisyah.lim@student.usm.my	150008	<span>Graded</span>	34 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Syahir	Lee	syahir.lee@student.usm.my	150009	<span>Graded</span>	86 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Amir	Ng	amir.ng@student.usm.my	150010	<span>Grading</span>	-	<a href="#">Download</a>	<a href="#">Revert</a>

#### 4.2..17 View Submission Page – After releasing submission.

Figure 4.20 Instructor view submission page after releasing submission

Submission Status	
Submission	10/10
Grading Progress	10/10

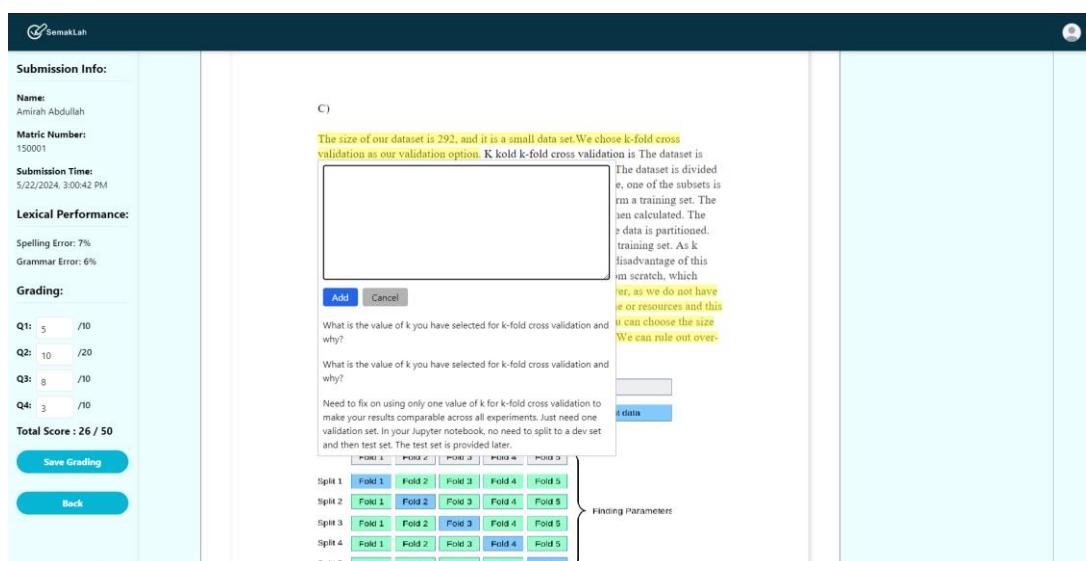
  

Submission List							
First Name	Last Name	Email Address	Student ID	Grading	Grade	Download	
Amirah	Abdullah	amirah.abdullah@student.usm.my	150001	<span>Graded</span>	52 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Hafiz	Ibrahim	hafiz.ibrahim@student.usm.my	150002	<span>Graded</span>	76 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Nurul	Aziz	nurul.aziz@student.usm.my	150003	<span>Graded</span>	68 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Muhammad	Hassan	muhammad.hassan@student.usm.my	150004	<span>Graded</span>	64 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Siti	Mohamed	siti.mohamed@student.usm.my	150005	<span>Graded</span>	52 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Azim	Wong	azim.wong@student.usm.my	150006	<span>Graded</span>	48 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Fatin	Tan	fatin.tan@student.usm.my	150007	<span>Graded</span>	50 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Aisyah	Lim	aisyah.lim@student.usm.my	150008	<span>Graded</span>	34 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Syahir	Lee	syahir.lee@student.usm.my	150009	<span>Graded</span>	86 / 100	<a href="#">Download</a>	<a href="#">Revert</a>
Amir	Ng	amir.ng@student.usm.my	150010	<span>Graded</span>	60 / 100	<a href="#">Download</a>	<a href="#">Revert</a>

After clicking in the button, instructor will be redirected to this page where they are able to see a dashboard of a submission status. The submission status includes the total number of submissions and the grading process. Other than that, the instructor can also monitor the list of submission from this page. To grade a student, instructor can click on the button; To download an assignment, instructor can click on the icon, and the assignment will be downloaded into their local system. Once the instructor has graded the assignment with marks, the button will turn to and the total mark will be displayed. Once all the assignments are ready to be released, the instructor can click on button to release the assignment to students. They can also choose to revert the submission by clicking on the button if there is a need to do so.

#### 4.2..18 Grading & Feedback Page (While giving feedback)

Figure 4.21 Instructor grading & feedback page (while giving feedback)



## 4.2..19 Grading & Feedback Page

Figure 4.22 Instructor grading & feedback page.

**Submission Info:**

- Name: Amirah Abdullah
- Matric Number: 150001
- Submission Time: 5/22/2024, 3:00:42 PM

**Lexical Performance:**

- Spelling Error: 7%
- Grammar Error: 6%

**Grading:**

Q1:	5 /10
Q2:	10 /20
Q3:	8 /10
Q4:	3 /10

Total Score : 26 / 50

**Save Grading**

**Back**

**Assignment 1 Report**  
Semester II 2021/2022

Name	Matrix Number

**A)**

Figure1

The distribution of the class label is balance from since the minority group "yes" is more than 40% of the data set. From the above Figure1, we see that the distribution of the class label is not a skewed class proportion.

**B)**

F1 is calculated as follows:

$$F_1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

where:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

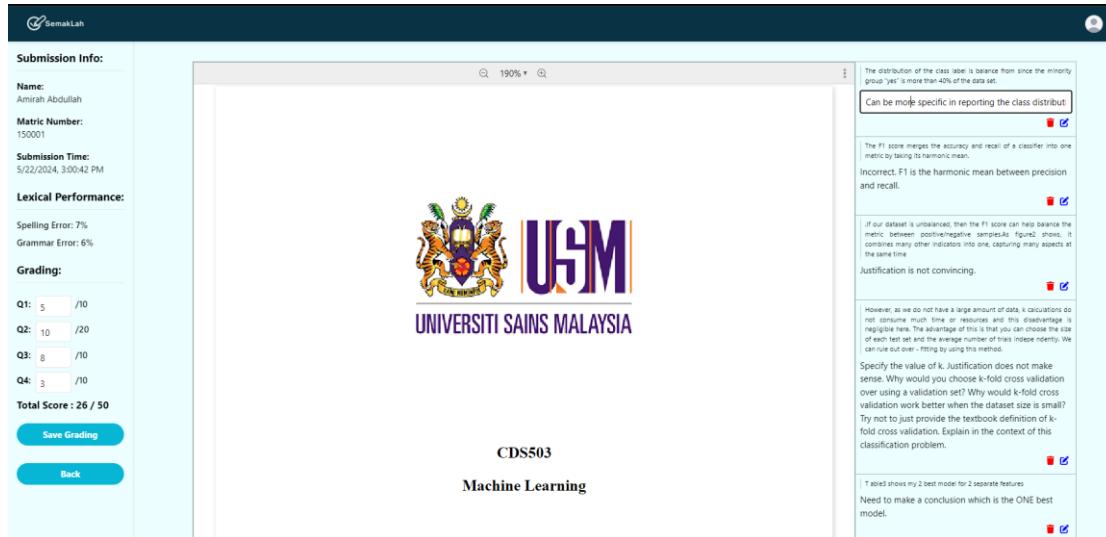
**Feedback Comments:**

- The distribution of the class label is balance from since the minority group "yes" is more than 40% of the data set. Can be more specific in reporting the class distribution between the two classes. (Red)
- The F1 score merges the accuracy and recall of a classifier into one metric by taking its harmonic mean. Incorrect: F1 is the harmonic mean between precision and recall. (Red)
- If our dataset is unbalanced, then the F1 score can help balance the metric between positive/negative samples. As figure1 shows, it combines many other indicators into one, capturing many aspects at the same time. Justification is not convincing. (Red)
- However, as we do not have a large amount of data, k calculations do not consume much time or resources and this disadvantage is negligible here. The advantage of this is that you can choose the size of each test set and the average number of trials independently. We can rule out over-fitting by using this method. (Green)
- Table3 shows my 2 best model for 2 separate features. Need to make a conclusion which is the ONE best model. (Red)
- Since the data is from different student, the emotion score of every student can be vary in a large range, and normalization data is able to reduce the training model errors. Inaccurate. Normalization does not guarantee to reduce the trained model error. (Red)
- First we can ask experts can lead to significant improvements. Not relevant. (Red)
- Then we can try to use other new classification algorithms and improve parameters. This should be part of the scope of the assignment. (Red)
- Oversample a few classes and/or under sample most classes to reduce class imbalances. Does not make sense as you have reported that the class distribution is balanced. (Red)

**Save Feedbacks**

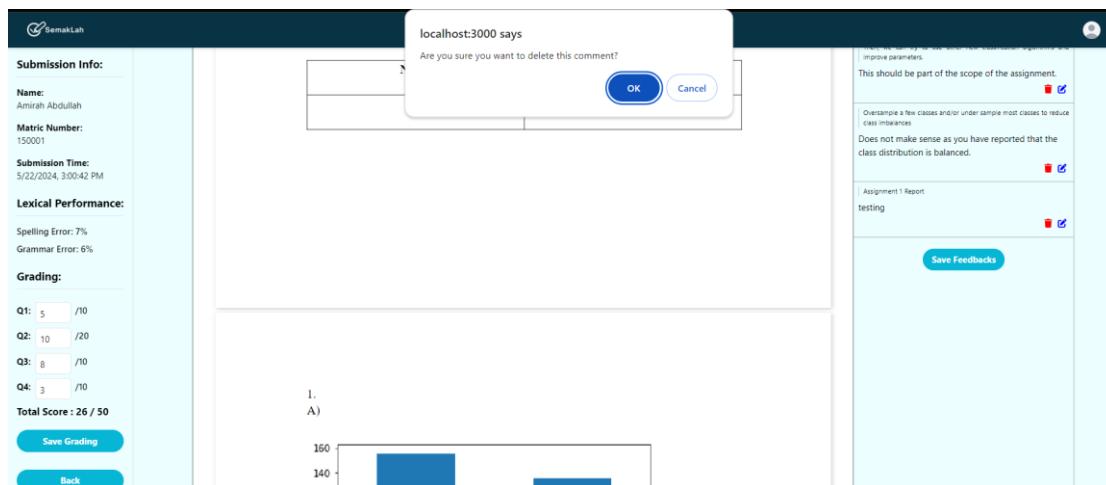
#### 4.2..20 Grading & Feedback Page – Editing comments.

Figure 4.23 Instructor edits comments.



#### 4.2..21 Grading & Feedback Page – Deleting comments.

Figure 4.24 Instructor deletes comments.

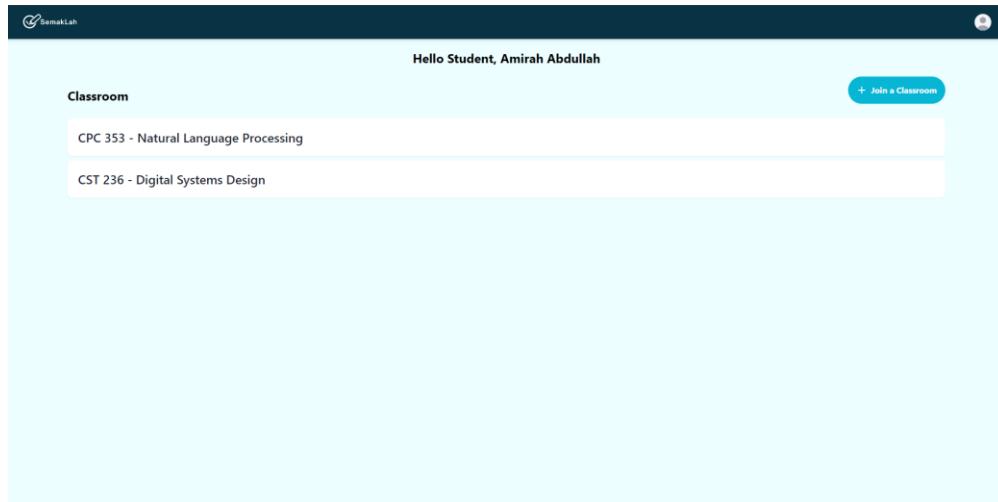


This is the grading and feedback page once the instructor clicks on an individual assignment to grade. The left panel here displays the summary of submission info, lexical performance, and grading. The PDF of the submission is shown in the center of the page, while the right panel displays all the comments. The instructor should click on the button **Save Feedbacks** to save the feedback. Otherwise, instructor can click on the **☒** icon to update the comments, click **☒** icon to delete the comments and confirm the deletion.

### 4.2.3. Students

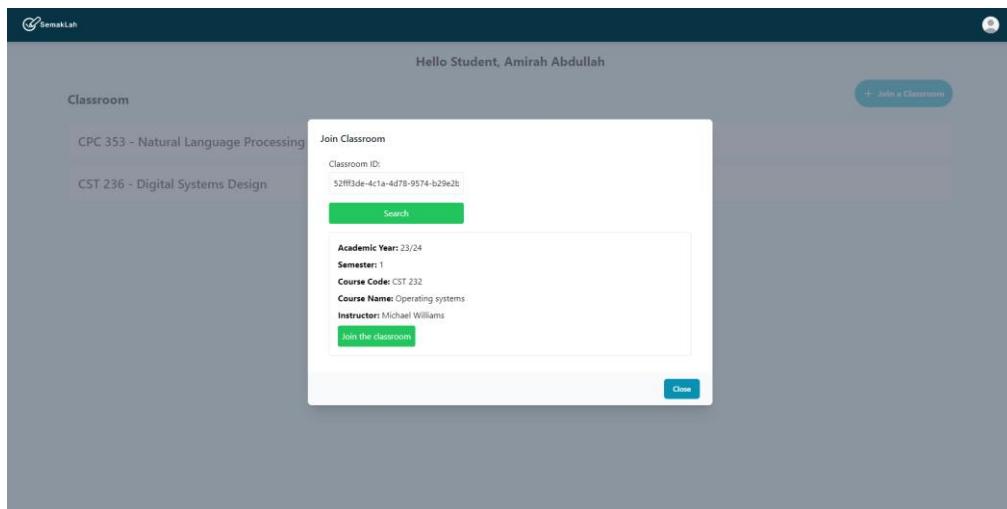
#### 4.2..1 Homepage

*Figure 4.25 Student homepage.*



#### 4.2..2 Join a Classroom

*Figure 4.26 Student join a classroom.*



The homepage of a student is slightly different from the instructor homepage, student will be able to see the classroom that they are enrolled but they are not able to manage the classroom, but they can join an available classroom by clicking on the [+ Join a Classroom](#) button, they can insert the class token given by their instructor and [Search](#) whether the classroom is still available. If the classroom is still available, they can click on the [Join the classroom](#) button to join the classroom.

#### 4.2..3 Classroom Details Page

*Figure 4.27 Student classroom details page.*

The screenshot shows a web-based application interface for a classroom. At the top, there is a header with the logo 'SemakLah' and a 'Back' button. Below the header, the title 'CPC 353 - Natural Language Processing' is displayed. A descriptive text block follows, explaining the course content. Underneath, there is a section titled 'Assignments' containing two entries: 'Assignment 2: Clustering' (start date: 5/1/2024, end date: 6/1/2024) and 'Assignment 1 Classification' (start date: 3/31/2024, end date: 5/30/2024).

When the student clicks into a specific classroom they can view the classroom details including the list of assignments.

#### 4.2..4 Assignment Details Page (with feedback button)

*Figure 4.28 Student assignment details page (with feedback button)*

The screenshot shows a detailed view of an assignment. At the top, it displays the assignment title 'Assignment 1 Classification' along with its start and end dates. Below this, there is an 'Instruction' section with a note about submission format ('Submit in PDF Format. Max 10 pages content.') and two buttons: 'Add Submission' and 'Give Feedback'. The main content area contains a document titled 'CDS503: Machine Learning Assignment 1: Classification'. The document discusses depression as a mood disorder and provides a problem statement about classifying depression based on emotion patterns. It includes a list of questions for the student to study and answer. The bottom of the document shows a navigation bar with icons for back, forward, search, and other document functions.

#### 4.2..5 Assignment Details Page

Figure 4.29 Student assignment details page by default.

The screenshot shows the assignment details page for 'Assignment 2: Clustering'. At the top, there are buttons for 'Back' and a user profile icon. Below that, the title 'Assignment 2: Clustering' is displayed. Underneath the title, there are two timestamp fields: 'Start Date : 01/05/2024, 12:00:00 am' and 'End Date : 01/06/2024, 11:59:00 pm'. A section titled 'Instruction' contains the following text: 'Assignment 2 is an individual assignment. Submit the following items in the assignment dropdown elearn latest by 11.59 p.m. on Friday, 1 June 2024. Please make sure you include your full name and matric number on your report.' Below this is a blue 'Add Submission' button. The main content area is a PDF viewer showing the assignment document. The PDF title is 'CDS503: Machine Learning Assignment 2: Clustering'. It includes a table of attributes: longitude (Geographic coordinate that specifies the east–west position of a point on the Earth's surface), latitude (Geographic coordinate that specifies the north–south position of a point on the Earth's surface), and country\_name (Country name). The PDF also contains instructions for K-Means Clustering (Part 1: 25 marks) and Hierarchical Clustering (Part 2: 25 marks). At the bottom of the PDF, there is a grading rubric table with three rows. The first row (Question Number 1, Mark Possible 25) describes appropriate clustering with justified experimentation. The second row (Question Number 2, Mark Possible 25) describes appropriate clustering with thoughtful comparison. The third row (Question Number 3, Mark Possible 50) based on accuracy of the best performing clustering on actual continent for each country.

Question Number	Mark Possible	Description
1	25	Number of clusters selected is appropriate and clearly justified. Demonstrate thorough experimentation. Provide correct and thoughtful description of clusters.
2	25	Number of clusters selected is appropriate and clearly justified. Demonstrate thorough experimentation. Provide correct and thoughtful comparison.
3	50	Based on accuracy of the best performing clustering on actual continent for each country

When a student clicks into a specific assignment, they can see the details of the assignments including the start date, end date, instruction, PDF question file, and grading rubrics. Students should click on the **Add Submission** button to upload their assignments. If the student has submitted an assignment and the instructor has released the assignments, they will be able to click the **View Feedback** button to view their grades distribution and feedback.

#### 4.2..6 Submission Page – Before submission, before due date

Figure 4.30 Student submission page (Before due date, without submission)

The screenshot shows a web-based submission interface for an assignment titled "Assignment 2: Clustering". At the top, there is a "Submission Status" section containing the following table:

Submission Status	Not Submitted
Time Remaining	4 days, 17 hours, 50 minutes
Last Modified	-
Grading Status	Not Graded

Below this is a "File Submission" section with a file input field showing "No file chosen", a "Delete" button, and a "Submit" button.

#### 4.2..7 Submission Page – After due date

Figure 4.31 Student submission page (After due date, without submission)

The screenshot shows a web-based submission interface for an assignment titled "Designing a Simple Digital Clock Using VHDL". At the top, there is a "Submission Status" section containing the following table:

Submission Status	Not Submitted
Time Remaining	Overdue by 0 days, 6 hours, 14 minutes
Last Modified	-
Grading Status	Not Graded

Below this is a "File Submission" section with a file input field showing "No file chosen", a "Delete" button, and a "Submit" button.

#### 4.2..8 Submission Page – After Submission

Figure 4.32 Student submission page (after submission)

The screenshot shows the 'Assignment 2: Clustering' submission page on the SemeLah platform. At the top, there's a 'Submission Status' section with the following details:

Submission Status	Submitted
Time Remaining	4 days, 17 hours, 49 minutes
Last Modified	28/05/2024, 02:44:20 am
Grading Status	Not Graded

Below this is a 'File Submission' section showing a PDF file named 'Report JLSY\_01.pdf' uploaded. There are 'Delete' and 'Submit' buttons below the file preview.

The main content area displays the assignment document itself, which includes the USM logo, APEX logo, course code CDS503, assignment title 'ASSIGNMENT 2', and the academic session 'ACADEMIC SESSION 2021/2022 SEMESTER 2'.

When student clicks on the **Add Submission** button, they will be redirect to the submission page. If the submission phase is still active, students will upload their assignment in a PDF format and click on the **Submit** button; if the submission phase is no longer active, the buttons will be disabled. After submitting the assignment, if the submission phase is active, students can click on **Delete** button to delete the assignment and reupload a new submission.

#### 4.2..9 Feedback review page

Figure 4.33 Student feedback review page.

The screenshot shows a student feedback review page. On the left, there's a 'Submission Info' panel with fields for Name (Amin Ng), Matric Number (198010), and Submission Time (5/3/2024, 11:11:19 PM). Below that are sections for Lexical Performance (Spelling Error: 0%, Grammar Error: 7%) and Grading (Q1: 8 / 10, Q2: 8 / 20, Q3: 7 / 10, Q4: 7 / 10, Total: 30 / 50 (60.00%)).

The central area contains several questions and their answers:

- a. Report the class distribution. Is this a balanced or unbalanced data set?**

A class imbalance problem with skewed data set is considered unbalanced dataset. For example, a dataset shows whether a patient is diagnosed with cancer with more instances of 'Yes' than 'No'. It considers imbalance data set.

In the above example, the patient diagnosed with cancer is called majority class while those make up smaller proportion is called minority class. The degree of imbalance is highlighted below.

Degree of Imbalance	Proportion of Minority Class
High	>40% of the data set
Moderate	90% of the data set
Low	<10% of the data set

From the above table, the sharp boundary 'Depression', with 74% represents the depression and 'Yes' represents depression. It has two extremes 0 and 1 respectively using k-fold encoder. To check the data distribution, the method 'Value count' is used on the Depression feature, giving the result of 185 'No' and 100 'Yes'. Since the minority class is more than 40%, the data set is considered unbalanced.
- b. Please select and justify a suitable metric to evaluate the performance of your classification model.**

The suitable metric to evaluate the performance of classification model would be F1 score.

There are four types of performance metric: accuracy, precision, recall and F1 score. Accuracy is measure of how many instances are correctly predicted over total instances. Precision is measure how good of the model at what it predicted. Recall is measure how good of the model at picking the correct ones. F1 is harmonic mean of precision and recall.

In the assignment task we have passed the disease to patient. In the case of tuberculosis the patient is diagnosed with Depression which is false so we can use unnecessary treatment. In the case of true negative the patient is diagnosed as no Depression when he is, he will not receive any treatment. We can easily say if the disease happen, therefore it is a better choice since it takes the harmonic mean between recall and precision.
- c. Given the size of the data set, which validation option (e.g., percentage split, k-fold cross validation) do you think is suitable to be used in your machine learning experiments. Specify the validation option you are selecting for your machine learning experiments. Briefly explain the reason for using the validation option.**

The size of data set is 201 entries. Given the data set is tiny, it will be more appropriate to use k-fold cross validation.

Percentage split splits the data into training, validation, and testing set. Given our data entries only 201, the splitting of data will result in only small range of data being allocated to each stage, decreasing the performance of the model.

K-fold cross validation suggests that data is divided into  $k$  folds, one subset is training, remaining serve as testing and continue till every fold has been testing fold. The advantage of this method is less data is required but needs more computation power. Considering we have only 201 entries, it will be ideal in this case.
- d. Which feature representation produces a better model? Explain how you determine the best performing model based on the performance metric you have selected. Can you explain why one feature representation is better than the other?**

The classification algorithms that are going to run in this experiment are K-nearest Neighbors, Naive Bayes and Support Vector Machine. These three algorithms are run on the Frey-Face-binary and Norm-Face-Binary respectively.

The first step is data cleaning and data preparation. After load data into the data frame, we check is there any missing value in the data using is.na() function. The result is shown as below.

```
##> #> [1] "parsons.csv, frame, datasetname"
##> #> [1] "NaiveBayes: 201 entries, 3 to 196
##> #> [1] "NormFace: 201 entries"
```

The right panel contains a 'Class Distribution' table and several comments:

- Is this assignment task is to predict the depression prior to the use of k-fold encoder (the patient is diagnosed with depression or not), because using k-fold encoder, the data is split into k groups, then each group is assigned to the different model. We don't know any of the specific hyper-parameters is used for this k-fold encoder.**
- You have provided convincing justification of your primary performance metric selection. Just go with F1 score.**
- Increasing the performance of the model.**
- Not the most appropriate justification. Link back to model being biased due to limited data.**
- Classification solution suggests that data is needed much.**
- What is the value of k you have selected for k-fold cross validation and why?**
- The k-fold cross validation is used to compare the accuracy, precision and F1 score of the model. Need to fix on using only one value of k for k-fold cross validation to make your results comparable across all experiments. The goal is to compare different values of k in k-fold cross validation. Performance improvement should not be a product of the validation option. Rather, it should come from the properties of the model.**
- The best performing algorithm has been highlighted in yellow which is Naive Bayes in Frequency Feature Representation Model and SVM Polywak in Normalized Face Representation model.**
- Need to make a conclusion which is the ONE best model.**
- By comparing KNN score, we found that the Normalized Feature Representation Model score is 1.0. Whereas, SVM score is 0.98. Hence, KNN score is better than SVM score. Hence, KNN Model will have more F1 score of 0.98, hence it's better.**
- Performance metric for the Freq and Norm baseline models should be the same. Set the random\_state parameter in DummyClassifier to a fixed value.**
- It is not a good idea for reporting.**
- There is no evidence comparing training error and validation or test error to prove this statement.**
- In this case, coding more features will help the need to improve.**
- More data usually means addition in terms of the number of instances.**
- Features of faces can be added such as eyes, nose, mouth, hair, skin, age, gender, education, and social status, etc. The face detection library (HOG).**
- Gender is already included as a feature. The class label is partially based on HOG so cannot be used as features.**
- The best suggestion is to add patient M feature. Patient M has a very high probability of being healthy. Feature to an individual can help to predict the data better. For example, the values can be created by having the values in column 'Healthy' into a sequence of power 2. The numerical values gives flexibility to model, however it can.**
- For this to work, you must first provide evidence that the current hypothesis function learnt is too simple and the model suffers from high bias (underfitting).**

When student click into the **View Feedback** button, they will be able to see their lexical performance and grade distribution at the left panel, and all the comments on the right panel.

## 4.3. Implementation of Intelligent Component

### 4.3.1. Lexical Analysis (Spelling & Grammar Error)

The implementation of lexical analysis begins by importing essential modules:

*Figure 4.34 Importing essential modules*

```

1 import sys
2 import io
3 import language_tool_python
4 import pdfplumber
5 import json

```

1. “sys” for interfacing with the Python runtime,
2. “io” for handling input/output operations,
3. “language\_tool\_python” for conducting grammar and spelling checks,
4. “pdfplumber” for extracting text from PDF files, and
5. “json” for outputting results in JSON format.

*Figure 4.35 Core function.*

```

7 def analyze_pdf_document(pdf_data):
8     tool = language_tool_python.LanguageTool('en-US')
9
10    # Read the document content
11    with pdfplumber.open(io.BytesIO(pdf_data)) as pdf:
12        document_content = '\n'.join(page.extract_text() for page in pdf.pages)
13
14    # Calculate total number of words in the document
15    total_words = len(document_content.split())
16
17    # Perform spelling and grammar check
18    matches = tool.check(document_content)
19
20    # Get the number of spelling errors
21    spelling_errors = [match for match in matches if match.ruleId == 'MORFOLOGIK_RULE_EN_US']
22
23    # Get the number of grammar errors
24    grammar_errors = [match for match in matches if match.ruleId != 'MORFOLOGIK_RULE_EN_US']
25
26    # Calculate and return the percentage of spelling and grammar errors
27    spelling_error_percentage = (len(spelling_errors) / total_words) * 100
28    grammar_error_percentage = (len(grammar_errors) / total_words) * 100
29
30    return spelling_error_percentage, grammar_error_percentage
31
32 # Use the function
33 pdf_data = sys.stdin.buffer.read() # Read the PDF data from stdin
34 spelling_error_percentage, grammar_error_percentage = analyze_pdf_document(pdf_data)
35

```

The core function, “analyze\_pdf\_document()” takes the PDF data as input, extracts the text content from each page of the PDF, then calculates the total number of words in the document. It then utilizes the “LanguageTool” module to check for spelling and grammar errors within the document. The matches found by “LanguageTool” are categorized into spelling errors (identified by the rule ID ‘MORFOLOGIK\_RULE\_EN\_US’) and grammar errors (all other rule IDs).

The MORFOLOGIK\_RULE\_EN\_US is a rule identifier used within the LanguageTool library to denote specific spelling errors detected in text, particularly in the context of US English. The "MORFOLOGIK" part of the identifier refers to the underlying algorithm used for morphological analysis, which essentially breaks down words into their basic linguistic components to identify potential misspellings. When the LanguageTool object is used to analyze text, occurrences of misspelled words that trigger this rule are flagged as potential spelling errors.

Grammar errors are detected by LanguageTool through the application of various language rules specific to the grammar structure of the language being checked. These rules cover a wide array of grammatical aspects, including subject-verb agreement, sentence structure, punctuation usage, and more. When LanguageTool analyzes the text, it compares the observed grammatical structure against these predefined rules. Instances where the text violates these rules are flagged as potential grammar errors. For example, LanguageTool might detect errors such as incorrect verb forms, missing punctuation marks, or improper sentence constructions.

After identifying the error, the function computes the percentage of each type of error relative to the total word count and returns these percentages.

Equation for spelling and grammar error percentages:

$$\text{Spelling error percentages} = \frac{\text{number of misspelled words}}{\text{number of total words}} \times 100\%$$

$$\text{Grammar error percentages} = \frac{\text{number of words with grammar error}}{\text{number of total words}} \times 100\%$$

Figure 4.36 json output

```

36     # Print the results as a JSON string
37     print(json.dumps({
38         'spelling': spelling_error_percentage,
39         'grammar': grammar_error_percentage,
40     }))

```

Finally, the script reads PDF data from standard input, analyzes it using the function, and prints the spelling and grammar error percentages as a JSON string.

### 4.3.2. Feedback Suggestion Generation

*Figure 4.37 Code snippet*

```

4   import sys
5   import json
6   import numpy as np
7   from sklearn.feature_extraction.text import TfidfVectorizer
8   from sklearn.metrics.pairwise import cosine_similarity
9
10  # Load the documents from stdin
11  input = json.load(sys.stdin)
12
13  documents = input['documents']
14  specific_sentence = input['specific_sentence']
15  feedback = input['feedback']
16
17  # Create TF-IDF vectorizer
18  vectorizer = TfidfVectorizer(stop_words='english')
19
20  # Compute TF-IDF matrix for the documents and the specific sentence
21  tfidf_matrix_documents = vectorizer.fit_transform(documents)

```

Feedback suggestion generation begins with retrieving historical sentences from the database. An instance of the “tfidfVectorizer” from the “sklearn” library is created. This vectorizer tokenizes the sentences into individual words, removes English stop words (like ‘is’, ‘the’, ‘and’, etc.), converts all characters to lowercase, and calculates the TF-IDF (Term Frequency-Inverse Document Frequency) values for each word in each document or sentence. The historical sentences are then converted into a matrix of TF-IDF features using the “fit\_transform” method of the “tfidfVectorizer”.

*Figure 4.38 Cosine similarity code snippet*

```

22  tfidf_matrix_specific_sentence = vectorizer.transform([specific_sentence])
23
24  # Calculate cosine similarities
25  cosine_similarities = cosine_similarity(tfidf_matrix_specific_sentence, tfidf_matrix_documents)

```

When a user highlights a text (specific sentence), the “transform” method uses the learned vocabulary dictionary from the vectorizer to convert the specific sentence into its TF-IDF representation. The cosine similarity between the TF-IDF vector of the specific sentence and the TF-IDF matrix of the historical documents is then calculated.

*Figure 4.39 Sorting code snippet*

```

27  # Get the indices of the documents sorted by similarity in descending order
28  sorted_indices = np.argsort(cosine_similarities[0])[::-1]
29
30  # Get the feedbacks sorted by similarity
31  sorted_feedbacks = [feedback[i] for i in sorted_indices[:3]]
32
33  # Write the output to stdout
34  output = {
35  |   'sorted_feedback': sorted_feedbacks
36  }

```

The documents are sorted in descending order based on their cosine similarity to the specific sentence, ensuring that the most similar documents are at the beginning of the list. The top 3 most similar sentences are retrieved from the documents, and the respective feedback will be suggested.

The equation of TF-IDF is:

$$TF - IDF(t, d, D) = TF(t, d) * IDF(t, D)$$

where:

- t is the term (word),
- d is the document,
- D is the corpus (set of documents),
- TF(t, d) is the term frequency of t in d, and
- IDF(t, D) is the inverse document frequency of t in D.

The equation for cosine similarity is:

$$\text{cosine\_similarity}(A, B) = \text{dot\_product}(A, B) / (\text{norm}(A) * \text{norm}(B))$$

Where:

- A and B are vectors,
- dot\_product(A, B) is the dot product of A and B, and
- norm(A) and norm(B) are the lengths (norms) of A and B.

## 4.4. Implementation Strategy

SemakLah's top-down implementation strategy involves a methodical breakdown of the system into modular components, each serving a distinct function within the platform.

We identify the major modules outlined in the specification: Classroom Management, Assignment Management, Feedback Generation, and Grading Module. Moving down a level, within the Classroom Management module, sub-modules can be identified. For instance, user authentication and registration can be broken down into smaller features such as username/password validation. Similarly, classroom management involves sub-functions like creating new classrooms.

Next, the Assignment Management module takes center stage. This includes functionalities for creating, editing, and managing assignments, as well as handling student submissions. Sub-modules within this module may include assignment creation/editing, submission management, and file handling.

Feedback Generation is another critical aspect of the system. This module involves extracting text from assignment files, analyzing for errors, suggesting feedback based on historical data, and providing a user-friendly interface for instructors to review and edit feedback. Sub-modules within this module would include text extraction, lexical analysis, similarity matching, and feedback suggestion generation.

Grading Module encompasses functionalities related to the assessment and evaluation of student submissions. This includes managing grading rubrics, entering scores, calculating final grades, and providing feedback to students. Sub-modules within this module might include rubric management, score entry, grade calculation, and feedback delivery.

By breaking down the system into these modular components, each with its own set of sub-modules and features, SemakLah can systematically develop and integrate the various functionalities required to create a comprehensive online learning platform. This top-down approach allows for a structured and manageable implementation process, ensuring that each component is developed and tested incrementally before being integrated into the larger system.

## CHAPTER 5: SYSTEM TESTING & EVALUATION

### 5.1. Testing Strategy

#### 5.1.1. Unit Testing

Each module in the project will undergo rigorous unit testing to validate the functionality of its individual components. Test cases will be meticulously designed to cover various scenarios outlined in the functional requirements. For the Classroom Management Module, unit tests will verify the correctness of login and sign-up processes for both instructors and students, as well as the functionality to create and manage online classrooms and student enrollments. Similarly, the Assignment Management Module will be tested to ensure proper assignment creation, student submission management, and file handling functionalities. The Feedback Generation Module will undergo unit tests to validate text extraction, lexical analysis, similarity matching, and feedback suggestion algorithms. Mock objects and stubs may be employed to simulate external dependencies, and unit tests will be automated to ensure efficiency and consistency in testing procedures.

#### 5.1.2. Integration Testing

After successful completion of unit testing, integration testing will be carried out to evaluate the seamless integration and communication between different modules of the system. Test cases will focus on verifying the interoperability of modules and the accuracy of data exchange between them. Integration tests will encompass scenarios such as registering for an account, creating, and managing online classrooms, assigning assignments to students, and generating feedback based on historical feedback. Emphasis will be placed on testing both positive and negative scenarios to identify any potential issues arising from module interactions. Integration testing will ensure that the system functions cohesively as a unified entity and adheres to the specified functional requirements.

### 5.1.3. System Testing

Once unit and integration testing phases are completed, comprehensive system testing will be conducted to assess the overall functionality and performance of the entire system. Test cases will emulate real-world usage scenarios to validate the system's behavior from end-user perspectives. System tests will cover login and registration processes, classroom and assignment management functionalities, student assignment submissions, feedback generation, and system scalability, reliability, and security aspects. Test environments will be carefully configured to replicate the production environment, allowing for thorough evaluation under simulated real-world conditions. System testing will ensure that the system meets the specified requirements, performs effectively, and delivers a seamless user experience across different usage scenarios. Additionally, performance testing and security testing will be conducted to identify and address any performance bottlenecks or vulnerabilities within the system.

## 5.2. Test Cases

### 5.2.1. TC001: Login Account

*Figure 5.1 Test case for Login Account*

<b>Test Case ID:</b>	TC001
<b>Test Case Description:</b>	Users (Instructor and student) log in to their respective accounts.
<b>Pre-condition:</b>	Users have valid credentials
<b>Test Procedure:</b>	<b>Expected Result:</b>
1. Enter email and password. 2. Click “Login” button	User is directed to their homepage

### 5.2.2. TC002: Register Account

*Figure 5.2 Test case for Register Account*

<b>Test Case ID:</b>	TC002
<b>Test Case Description:</b>	Users (Instructor and Student) create accounts to gain access to the system.
<b>Pre-condition:</b>	User does not have an account.
<b>Test Procedure:</b>	<b>Expected Result:</b>
1. Click on “Sign Up” 2. Enter ID, First Name, Last Name, Email, Password, Gender, and Role. 3. Click “Sign Up” button	User will be redirected to the login page

### 5.2.3. TC003: Manage Classroom

*Figure 5.3 Test case for Manage Classroom*

<b>Test Case ID:</b>	TC003
<b>Test Case Description:</b>	Instructor manages online classrooms based on different courses.
<b>Pre-condition:</b>	Instructor is logged into the system.
<b>Test Procedure:</b>	<b>Expected Result:</b>
1. Click on “Create Classroom” 2. Enter academic year, semester, course code, course name, course description and classroom limit. 3. Click “Create” button	User will see the classroom created in their homepage.
1. Click on “edit” icon. 2. Enter academic year, semester, course code, course name, course description and classroom limit. 3. Click “Update” button	User will see the classroom updated in their homepage.
1. Click on “Delete” icon. 2. Click on “OK” on the confirmation message.	User will see the classroom is deleted from the homepage

### 5.2.4. TC004: Manage Student Enrolment

*Figure 5.4 Test case for Manage Student Enrolment*

<b>Test Case ID:</b>	TC004
<b>Test Case Description:</b>	Instructor manages student enrolment for a selected online classroom
<b>Pre-condition:</b>	1. Instructor is logged into the system. 2. Classroom exists. 3. Student exists (if instructor is removing a student) 4. Instructors select a classroom
<b>Test Procedure:</b>	<b>Expected Result:</b>
1. Click “Enrol student” button. 2. Enter Student ID. 3. Click “Enrol” Button	Student is enrolled into the classroom and shown in the list

### 5.2.5. TC005: Set Assignment Attributes

*Figure 5.5 Test case for Set Assignment Attributes*

<b>Test Case ID:</b>	TC005
<b>Test Case Description:</b>	When managing assignment, instructor can set assignment attributes such as title, description, instructions, assignment start date, assignment end date, and upload the assignment question in PDF.
<b>Pre-condition:</b>	1. Instructor is logged into the system. 2. Instructor is creating assignment in a classroom.
<b>Test Procedure:</b>	<b>Expected Result:</b>
1. Provide the assignment attributes. 2. Click on “Create” Button	Assignment is created
1. Update the assignment attributes. 2. Click on “Update” Button	Assignment is updated

### 5.2.6. TC006: Manage Grading Rubric

*Figure 5.6 Test case for Manage Grading Rubric*

<b>Test Case ID:</b>	TC006
<b>Test Case Description:</b>	Instructor manages grading rubric for assignments.
<b>Pre-condition:</b>	1. Instructor is logged into the system. 2. An assignment is created. 3. Instructor clicks on the “rubric” icon.
<b>Test Procedure:</b>	<b>Expected Result:</b>
1. Click on “Add row” button. 2. Provide information. 3. Click on “Save” icon	A new grading rubric created
1. Edit information. 2. Click on “Save” icon	Grading rubric is updated
3. Click on “delete” button. 4. Provide information. 5. Click “OK” on the confirmation message	Grading rubric is deleted

### 5.2.7. TC007: Manage Assignment Submission

*Figure 5.7 Test case for Manage Assignment Submission*

<b>Test Case ID:</b>	TC008
<b>Test Case Description:</b>	Student manages their assignment submission.
<b>Pre-condition:</b>	1. Student is logged into the system. 2. Student is enrolled into the classroom. 3. Assignment submission phase is active. 4. Student click on “Add Submission” button
<b>Test Procedure:</b>	<b>Expected Result:</b>
1. Upload an assignment file. 2. Click on “Submit” button	Submission status is updated, and PDF is displayed

### 5.2.8. TC008: Grade Student Assignment

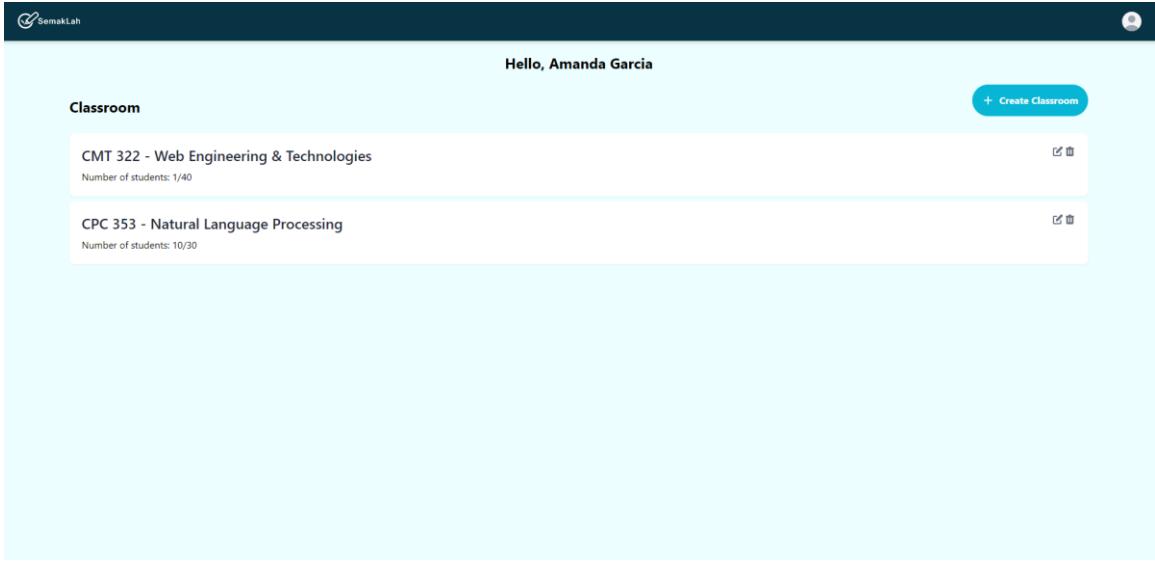
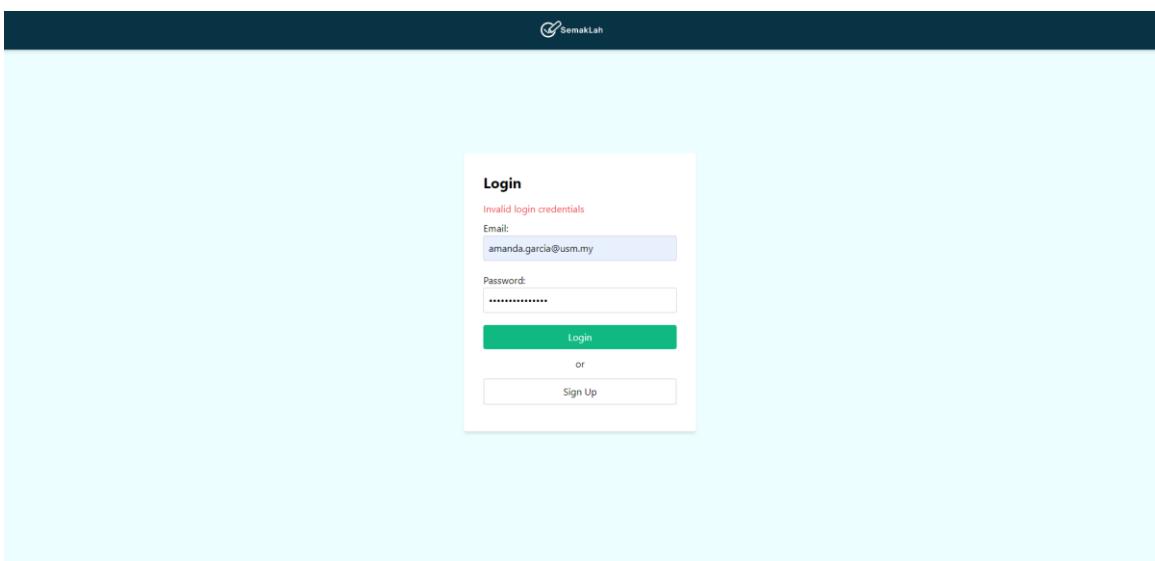
*Figure 5.8 Test case for Grade Student Assignment*

<b>Test Case ID:</b>	TC010
<b>Test Case Description:</b>	Instructor enters scores according to grading rubrics.
<b>Pre-condition:</b>	1. Instructor is logged into the system. 2. Grading rubric is available for the assignment. 3. Student submission exists. 4. Click on “Grading”
<b>Test Procedure:</b>	<b>Expected Result:</b>
1. Enters marks according to the grading rubrics for each assignment. 2. Click on “Save Grading” button	The mark for the assignment is updated.

## 5.3. Test Results

### 5.3.1. TC001: Login Account

Figure 5.9 Test result for Login Account

Test Data	Test Data Description	Expected result	Actual Results	Status
<b>Email:</b> amanda.garcia@usm.my <b>Password:</b> Letmein5678!	Correct email, password	Login to homepage	Login to homepage	Pass
				
<b>Email:</b> amanda.garcia@usm.my <b>Password:</b> Letmein	Correct email, wrong password	Display error message “Invalid login credentials”	Display error message “Invalid login credentials”	Pass
				

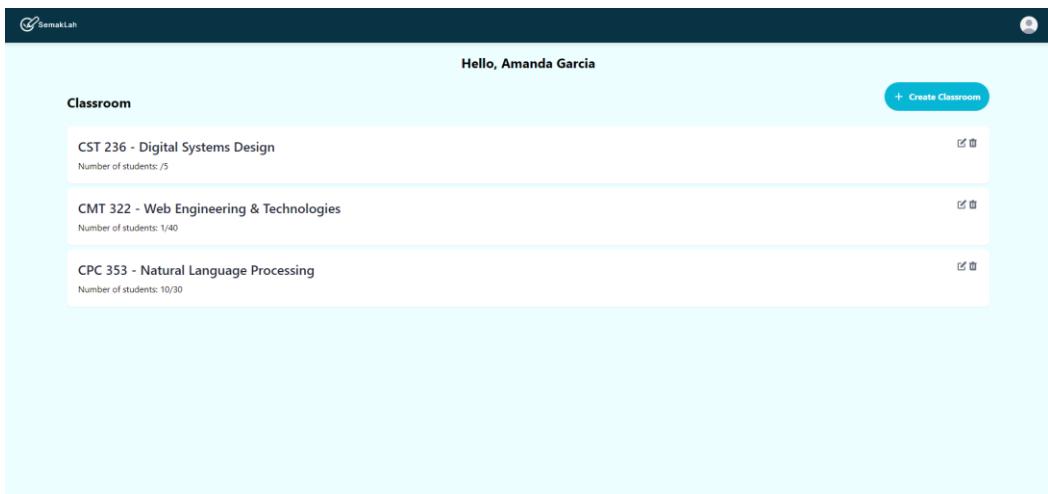
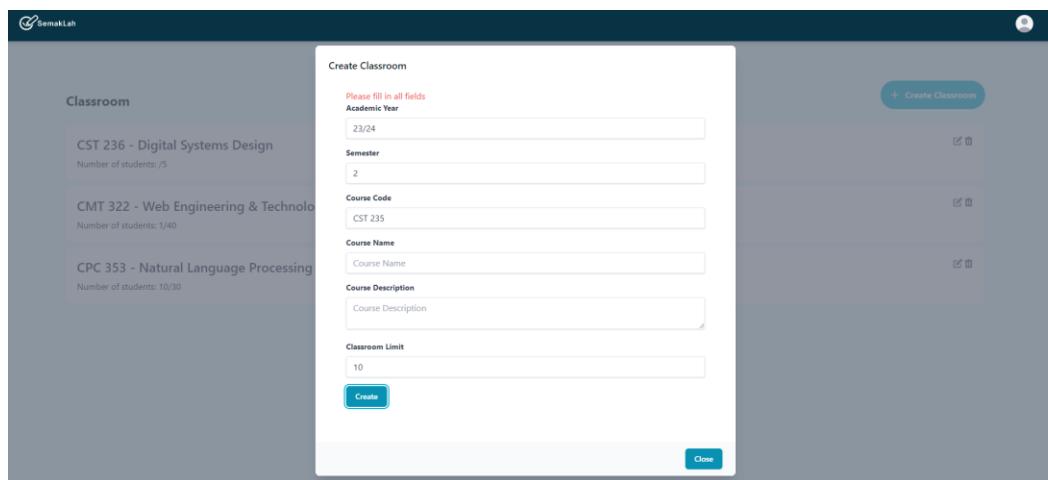
### 5.3.2. TC002: Register Account

Figure 5.10 Test result for Register Account

Test Data	Test Data Description	Expected result	Actual Results	Status
<b>Instructor ID or Student ID:</b> USM10000 <b>First name:</b> Samantha <b>Last name:</b> Johnson <b>Email:</b> samantha.johnson@usm.my <b>Password:</b> Passw0rd! <b>Gender:</b> female <b>Role:</b> Instructor	Complete data	Redirect to login page	Redirect to login page	Pass
<b>Instructor ID or Student ID:</b> USM11000 <b>First name:</b> Iris <b>Last name:</b> Yan <b>Email:</b> irisyan@student.usm.my <b>Password:</b> pwd12 <b>Gender:</b> Female <b>Role:</b> Student	Complete data  Password less than 6 characters	Display error message “Password should be at least 6 characters”	Display error message “Password should be at least 6 characters”	Pass

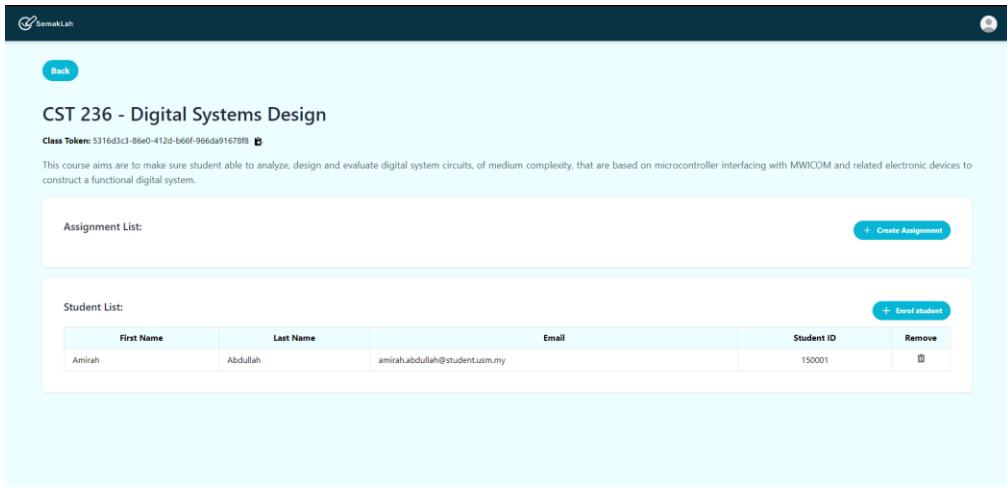
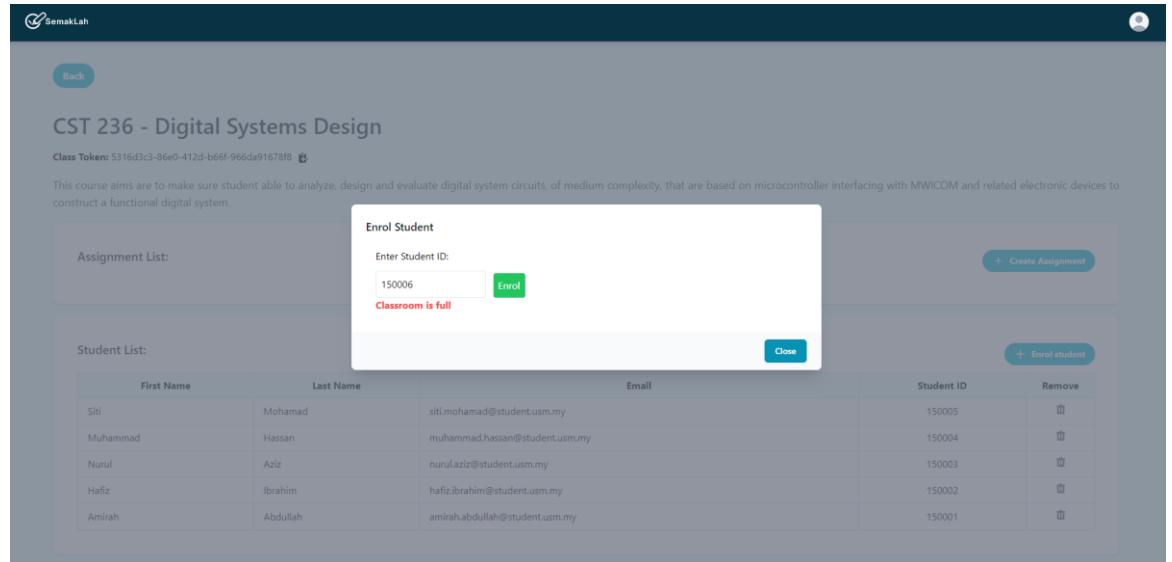
### 5.3.3. TC003: Manage Classroom

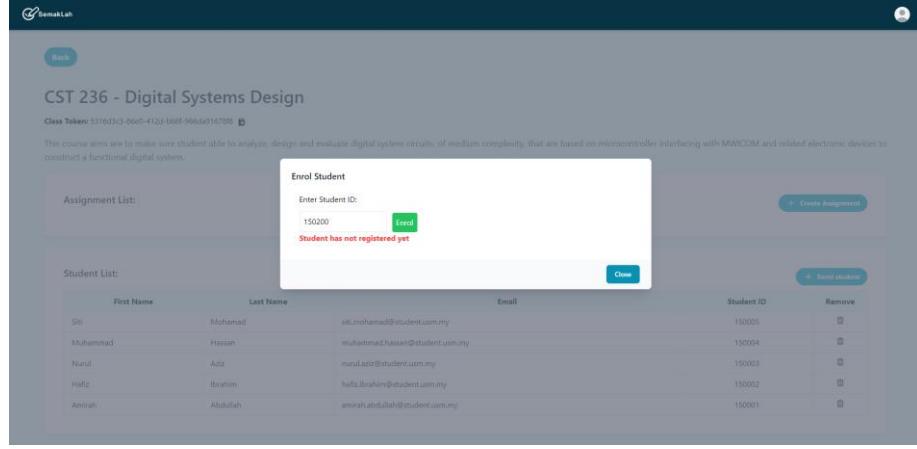
Figure 5.11 Test result for Manage Classroom

Test Data	Test Data Description	Expected result	Actual Results	Status
<b>Academic Year:</b> 23/24 <b>Semester:</b> 2 <b>Course Code:</b> CST 236 <b>Course Name:</b> Digital Systems Design <b>Course Description:</b> This course aims.... <b>Classroom limit:</b> 5	Complete data	Classroom is created at their homepage	Classroom is created at their homepage	Pass
<b>Academic Year:</b> 23/24 <b>Semester:</b> 2 <b>Course Code:</b> CST 235 <b>Course Name:</b> <b>Course Description:</b> <b>Classroom limit:</b> 5	Incomplete data	Display error message “Please fill in all fields”	Display error message “Please fill in all fields”	Pass
				

### 5.3.4. TC004: Manage Student Enrolment

Figure 5.12 Test result for Manage Student Enrolment

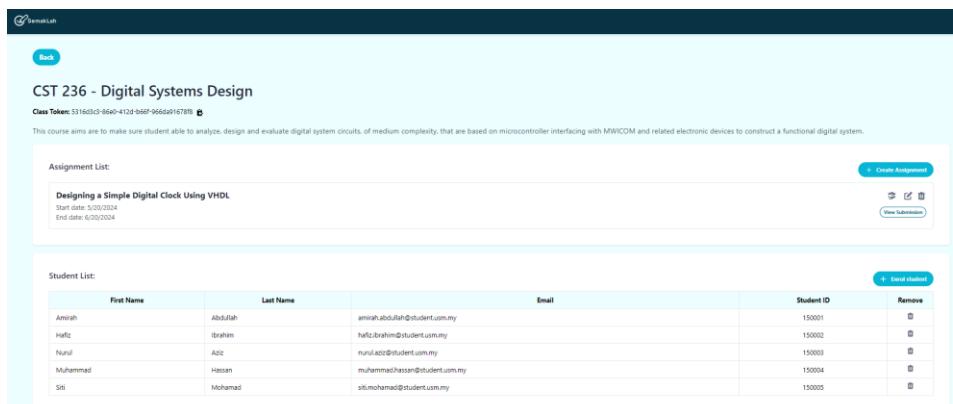
Test Data	Test Data Description	Expected result	Actual Results	Status
Student ID: 150001	Enrol a valid student	Student is enrolled into the classroom and shown in the list	Student is enrolled into the classroom and shown in the list	Pass
				
Student ID: 150006	Exceed class limit	Display error message “Classroom is full”	Display error message “Classroom is full”	Pass
				

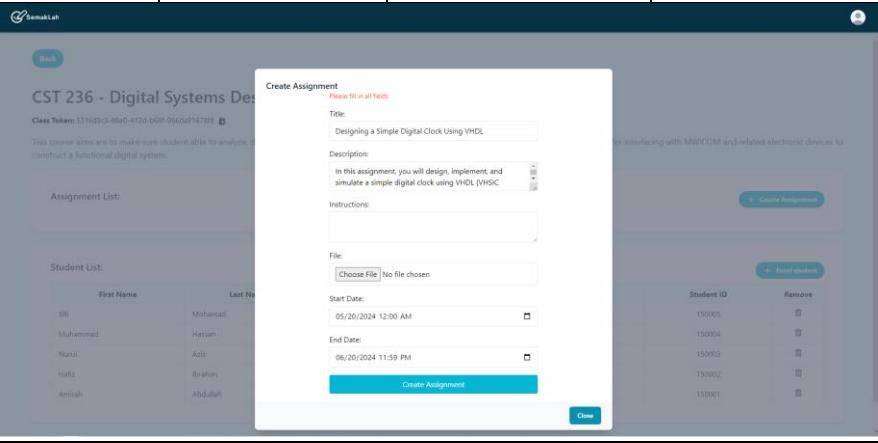
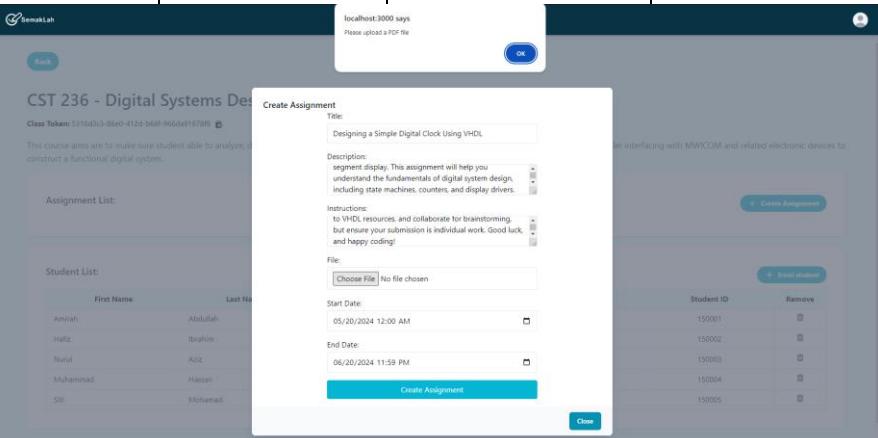
Student ID: 150200	Student has not registered yet	Display error message “Student has not registered yet”	Display error message “Student has not registered yet”	Pass
				

### 5.3.5. TC005: Set Assignment Attributes

Figure 5.13 Test result for Set Assignment Attributes

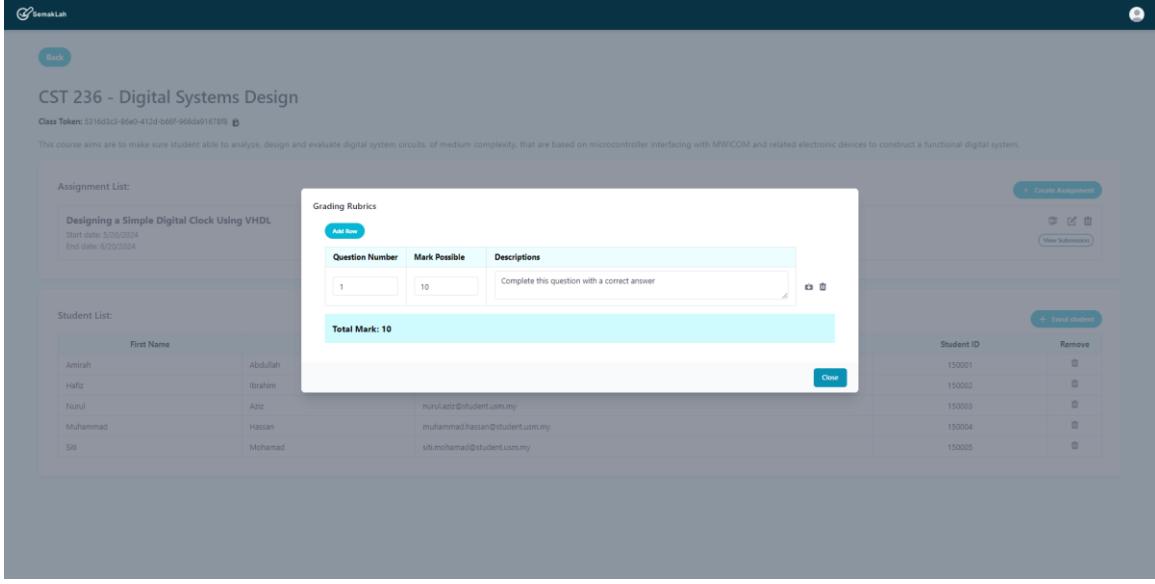
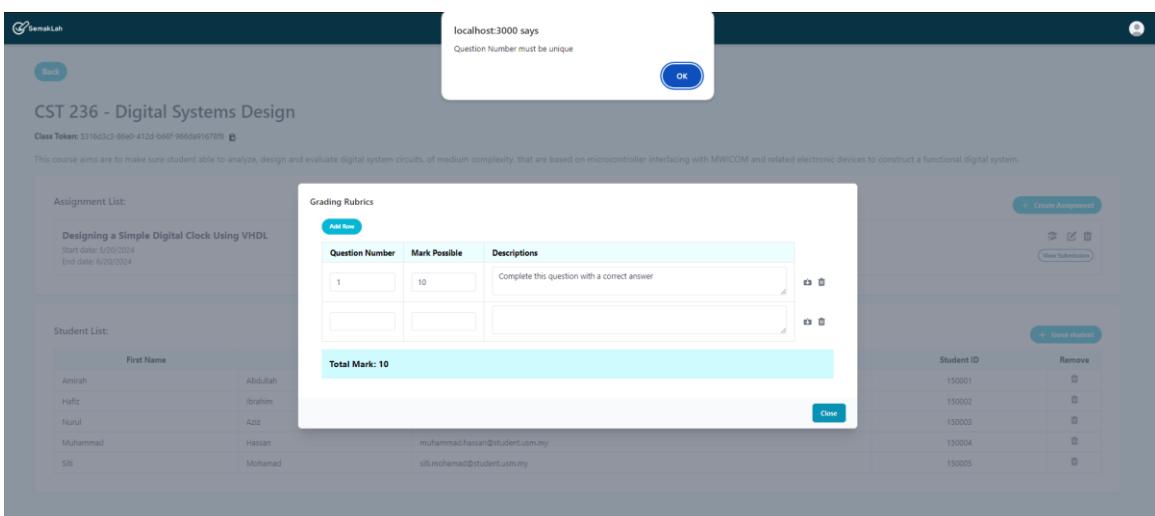
Test Data	Test Data Description	Expected result	Actual Results	Status
<b>Title:</b> Designing a Simple Digital Clock Using VHDL <b>Description:</b> In this assignment, ... <b>Instruction:</b> Design a digital clock using ... <b>File:</b> Assignment.pdf <b>Start Date:</b> 05/20/2024 12:00 AM <b>End Date:</b> 06/20/2024 11:59 PM	Complete data	Assignment is created	Assignment is created	Pass


--

<p><b>Title:</b> Designing a Simple Digital Clock Using VHDL</p> <p><b>Description:</b> In this assignment, ...</p> <p><b>Instruction:</b></p> <p><b>File:</b></p> <p><b>Start Date:</b> 05/20/2024 12:00 AM</p> <p><b>End Date:</b> 06/20/2024 11:59 PM</p>	Incomplete data	Display error message “Please fill in all fields”	Display error message “Please fill in all fields”	Pass
				
<p><b>Title:</b> Designing a Simple Digital Clock Using VHDL</p> <p><b>Description:</b> In this assignment, ...</p> <p><b>Instruction:</b> Design a digital clock using ...</p> <p><b>File:</b> Assignment.docx</p> <p><b>Start Date:</b> 05/20/2024 12:00 AM</p> <p><b>End Date:</b> 06/20/2024 11:59 PM</p>	Incompatible file	Display error message “Please upload a PDF File”	Display error message “Please upload a PDF File”	Pass
				

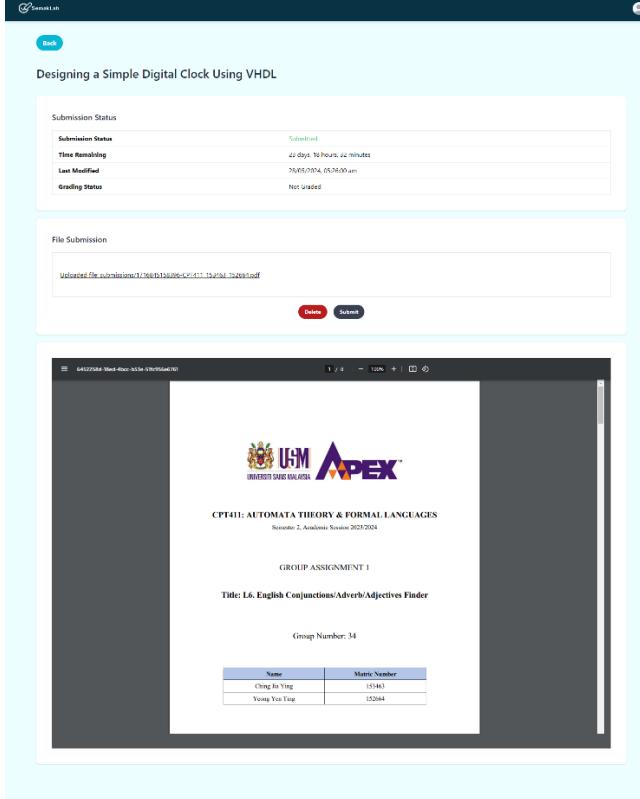
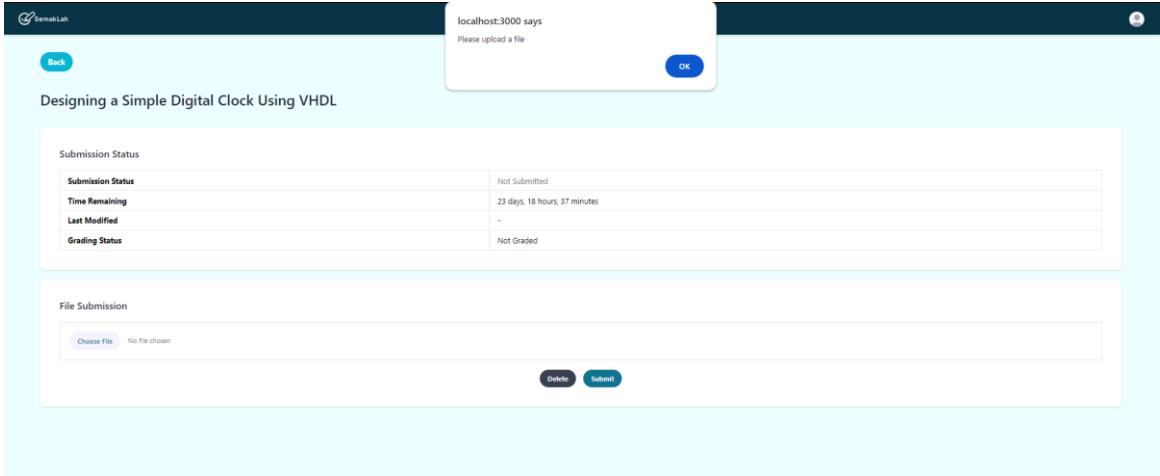
### 5.3.6. TC006: Manage Grading Rubric

Figure 5.14 Test result for Manage Grading Rubric

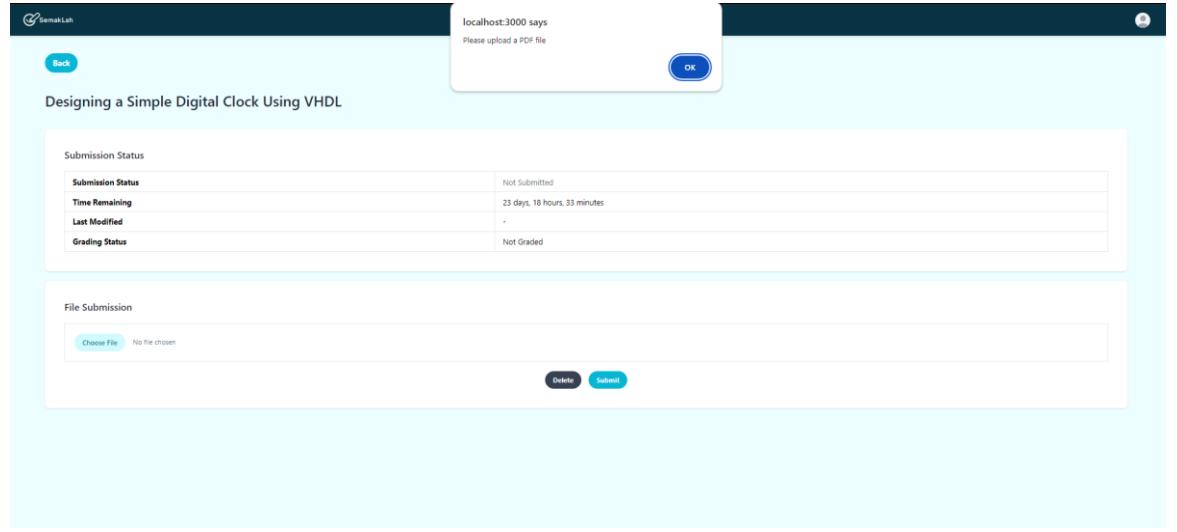
Test Data	Test Data Description	Expected result	Actual Results	Status
<b>Question Number:</b> 1 <b>Mark Possible:</b> 10 <b>Description:</b> Complete this question with a correct answer	Complete data	A new grading rubric created	A new grading rubric created	Pass
				
<b>Question Number:</b> 1 <b>Mark Possible:</b> 10 <b>Description:</b> Complete this question with a correct answer	Same question number	Display error message “Question Number must be unique”	Display error message “Question Number must be unique”	Pass
				

### 5.3.7. TC007: Manage Assignment Submission

Figure 5.15 Test result for Manage Assignment Submission

Test Data	Test Data Description	Expected result	Actual Results	Status
<b>File Submission:</b> Assignment.pdf	Complete data	Submission status is updated, and PDF is displayed	Submission status is updated, and PDF is displayed	Pass
				
<b>File Submission:</b> Incomplete data	Incomplete data	Display error message “Please upload a file”	Display error message “Please upload a file”	Pass
				

File Submission: Assignment.docx	Incompatible file	Display error message “Please upload a PDF file”	Display error message “Please upload a PDF file”	Pass
-------------------------------------	-------------------	--	--	------

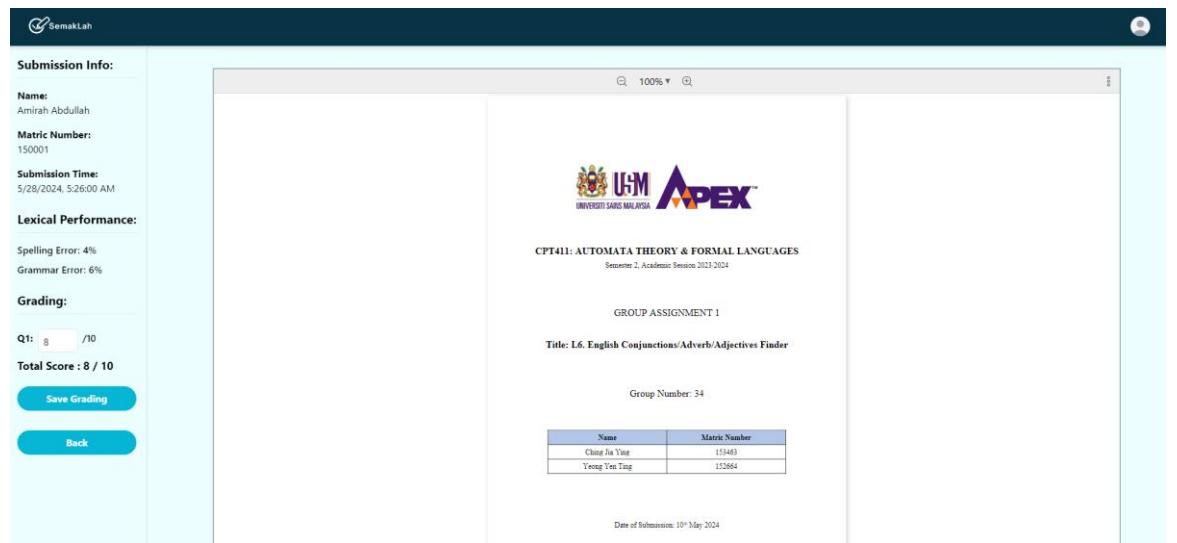


The screenshot shows a web page titled "Designing a Simple Digital Clock Using VHDL". At the top, there is a modal dialog box with the text "localhost:3000 says Please upload a PDF file" and a blue "OK" button. Below the modal, the main page displays "Submission Status" with fields: "Submission Status" (Not Submitted), "Time Remaining" (23 days, 18 hours, 33 minutes), "Last Modified" (empty), and "Grading Status" (Not Graded). Under "File Submission", there is a "Choose File" input field with the placeholder "No file chosen", a "Delete" button, and a "Submit" button.

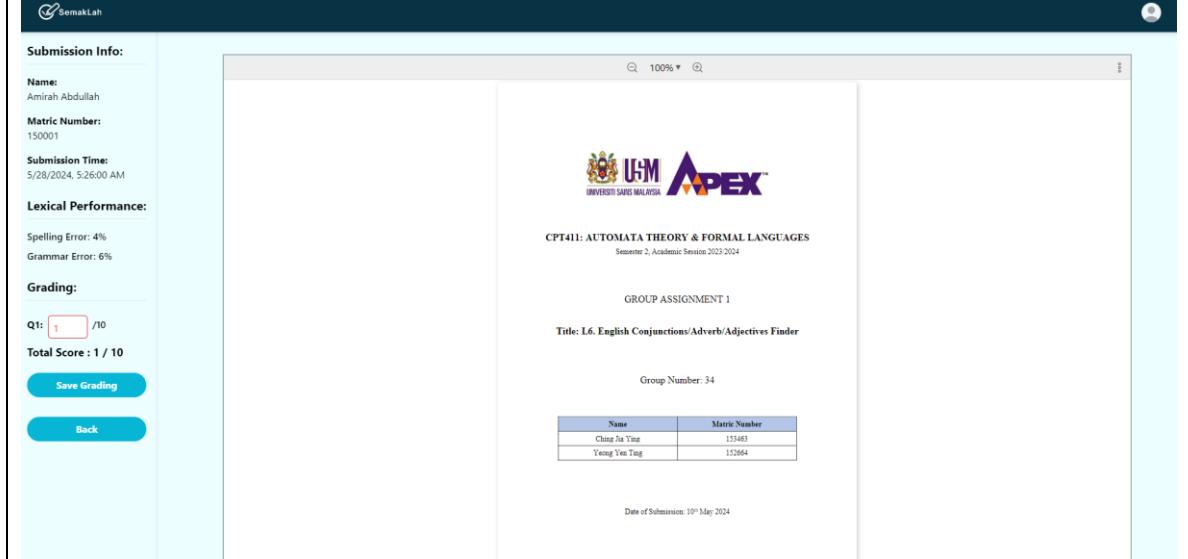
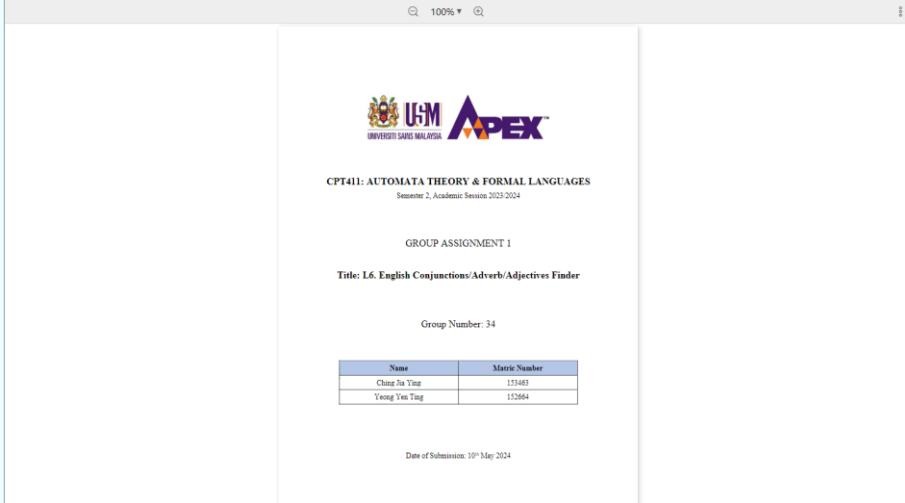
### 5.3.8. TC008: Grade Student Assignment

Figure 5.16 Test result for Grade Student Assignment

Test Data	Test Data Description	Expected result	Actual Results	Status
Q1: 8	Complete data	The mark for the assignment is updated.	The mark for the assignment is updated.	Pass



The screenshot shows a web page titled "Submission Info:". On the left, there is a sidebar with "Submission Info:" containing fields: "Name: Amriah Abdullah", "Matric Number: 150001", "Submission Time: 5/28/2024, 5:26:00 AM", "Lexical Performance: Spelling Error: 4%, Grammar Error: 6%", "Grading: Q1: 8 /10", "Total Score : 8 / 10", and buttons for "Save Grading" and "Back". The main content area displays the university logo (UUM APEX) and course information: "CPT411: AUTOMATA THEORY & FORMAL LANGUAGES Semester 2, Academic Session 2023/2024". It also shows "GROUP ASSIGNMENT 1", "Title: L6. English Conjunctions/Adverb/Adjectives Finder", "Group Number: 34", and a table of student names and matric numbers: Name: Chung fu Ying, Matric Number: 153463; Name: Yeong Yen Ting, Matric Number: 152664. At the bottom, it says "Date of Submission: 10th May 2024".

Q1:15	Exceed possible mark	The mark will turn red, and does not allow user to key in marks that exceed the possible mark	The mark will turn red, and does not allow user to key in marks that exceed the possible mark	Pass		
 <p>SemakLah</p> <p><b>Submission Info:</b></p> <p>Name: Amirah Abdullah Matric Number: 150001 Submission Time: 5/28/2024, 5:26:00 AM</p> <p><b>Lexical Performance:</b></p> <p>Spelling Error: 4% Grammar Error: 6%</p> <p><b>Grading:</b></p> <p>Q1: <span style="border: 1px solid red; padding: 2px;">1</span> /10 Total Score : 1 / 10</p> <p><b>Save Grading</b></p> <p><b>Back</b></p> <p data-cs="5" data-kind="parent">  <p>100%  </p> <p><b>UUM APEX™</b> UNIVERSITI SAINS MALAYSIA</p> <p><b>CPT411: AUTOMATA THEORY &amp; FORMAL LANGUAGES</b> Semester 2, Academic Session 2023-2024</p> <p>GROUP ASSIGNMENT 1</p> <p>Title: L6. English Conjunctions/Adverb/Adjectives Finder</p> <p>Group Number: 34</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Matric Number</th> </tr> </thead> <tbody> <tr> <td>Ching Fa Ying</td> <td>153463</td> </tr> <tr> <td>Yeong Ven Ting</td> <td>152664</td> </tr> </tbody> </table> <p>Date of Submission: 10<sup>th</sup> May 2024</p> </p>	Name	Matric Number	Ching Fa Ying	153463	Yeong Ven Ting	152664
Name	Matric Number					
Ching Fa Ying	153463					
Yeong Ven Ting	152664					

## 5.4. Model/Intelligent Component Evaluation

### 5.4.1. Lexical Analysis (Spelling & Grammar Error)

The evaluation of lexical analysis is based on 3 different assignments. The evaluation is done by evaluating every individual error, and the error is verified using Microsoft Word Spelling & Grammar feature. If the error shown is found in Word, then it is considered as a **True Positive**; If the error shown is not found in Word, and it really should not be considered as an error, then it will be considered as a **False Positive**. If there is an error shown in Word, but it's not found in the system, then it will be considered as a **False Negative**, otherwise it will be considered as **True Negative**.

With this information, a confusion matrix is formed to calculate the accuracy, false positive rate (FPR), precision, and recall. The definition and equation of accuracy, false positive rate (FPR), precision, and recall is shown below:

**Accuracy:** This is the percentage of accurate predictions, i.e., the ratio of number of correctly classified instances to the total number of instances

$$\text{Accuracy rate} = \frac{TP + TN}{TP + FN + FP + TN} \times 100\%$$

**False positive rate (FPR):** This measures the rate of wrongly classified instances.

$$\text{False Positive Rate (FPR)} = \frac{FP}{FP + TN} \times 100\%$$

**Precision:** This is the ratio of positively predicted instances among the retrieved instances

$$\text{Precision rate} = \frac{TP}{TP + FP} \times 100\%$$

**Recall:** this is the ratio of positively predicted instances among all the instances.

$$\text{Recall rate} = \frac{TP}{TP + FN} \times 100\%$$

The overall performance of the 3 different assignments is shown below:

#### Confusion Matrix:

*Table 5.1 Confusion Matrix*

Total: 2590		Actual	
		Positive	Negative
Predicted	Positive	32	13
	Negative	15	2530

$$\text{Accuracy rate} = \frac{32 + 2530}{2590} \times 100\% = \mathbf{98.92\%}$$

$$\text{False Positive Rate (FPR)} = \frac{13}{2543} \times 100\% = \mathbf{0.51\%}$$

$$\text{Precision rate} = \frac{32}{32 + 13} \times 100\% = \mathbf{71.11\%}$$

$$\text{Recall rate} = \frac{32}{32 + 15} \times 100\% = \mathbf{68.09\%}$$

The evaluation of lexical analysis across three different assignments reveals a high overall accuracy of 98.02%, indicating that the system is highly effective in identifying correct and incorrect lexical elements. The False Positive Rate, at a low 0.51%, suggests that the system rarely flags non-errors as errors, demonstrating its reliability in avoiding unnecessary corrections. However, the Precision rate of 71.11% and Recall rate of 68.09% indicate room for improvement in the system's precision and recall capabilities. A Precision rate of 71.11% means that when the system identifies an error, it is correct about 71.11% of the time, showing that a significant portion of the identified errors are indeed true positives. The Recall rate of 68.09% highlights that the system can identify approximately 68.09% of all actual errors, leaving about 31.91% of real errors undetected (false negatives). This balance between precision and recall suggests that while the system is adept at avoiding false alarms, it still misses a noticeable fraction of actual errors, indicating potential areas for further optimization and enhancement in the lexical analysis process.

### 5.4..1 Assignment 8:

#### Spelling Error:

*Table 5.2 Spelling error for assignment 8*

No.	Text	Result
1	Context: School of Computer Sciences Universiti Sains Malaysia CDS503 Machine Learning ..., Location: Offset 28, Length 10	TRUE
2	Context: School of Computer Sciences Universiti Sains Malaysia CDS503 Machine Learning 1) Stu..., Location: Offset 39, Length 5	TRUE
3	Context: ...redicted (classified) to be category B. Freq: Norm: c. Given the size of the data set..., Location: Offset 620, Length 5	TRUE
4	Context: ...ed (classified) to be category B. Freq: Norm: c. Given the size of the data set, whic..., Location: Offset 626, Length 5	TRUE
5	Context: ...Accuracy -- Accuracy Accuracy (error) - Precision, Recall rate - Recall, F1 score, Recall, Precisio..., Location: Offset 1280, Length 16	TRUE
6	Context: ...or) - Precision, Recall rate - Recall, F1 score, Recall, Precision and F1-score were used to ev..., Location: Offset 1314, Length 12	TRUE
7	Context: ...so,SVM is the best model. decision tree norm: decision tree freq: Svm norm: SVM freq: ..., Location: Offset 1864, Length 5	FALSE
8	Context: ...odel. decision tree norm: decision tree freq: Svm norm: SVM freq: 4) Is your overall ..., Location: Offset 1884, Length 5	FALSE
9	Context: ...decision tree norm: decision tree freq: Svm norm: SVM freq: 4) Is your overall best..., Location: Offset 1890, Length 3	TRUE
10	Context: ...sion tree norm: decision tree freq: Svm norm: SVM freq: 4) Is your overall best perfo..., Location: Offset 1894, Length 5	FALSE
11	Context: ...norm: decision tree freq: Svm norm: SVM freq: 4) Is your overall best performing mode..., Location: Offset 1904, Length 5	FALSE

## Grammar Error:

Table 5.3 Grammar error for assignment 8

No.	Text	Result
1	Context: ...refully and answer the questions below: a. Report the class distribution. Is this ..., Location: Offset 142, Length 2	FALSE
2	Context: ...oblem evaluation indicators:Accuracy -- Accuracy Accuracy (error) - Precision, Recall rate - Recal..., Location: Offset 1252, Length 17	TRUE
3	Context: ...al sample that predicted correct results,Precision:Of all the samples that were predicted ..., Location: Offset 1474, Length 10	TRUE
4	Context: ... be positive that were actually positive,Recall :The probability of being predicted to ..., Location: Offset 1566, Length 7	TRUE
5	Context: ...he balance between Precision and recall.When choosing between two different groups o..., Location: Offset 1719, Length 4	TRUE
6	Context: ..., consider the F-1 Score combining both.so,SVM is the best model. decision tree no..., Location: Offset 1824, Length 2	TRUE
7	Context: ...consider the F-1 Score combining both.so,SVM is the best model. decision tree norm: ..., Location: Offset 1826, Length 4	TRUE
8	Context: ...ombining both.so,SVM is the best model. decision tree norm: decision tree freq: Svm norm..., Location: Offset 1850, Length 8	TRUE
9	Context: ... very promising results. Suggestions: 1.In terms of data, increase the number of d..., Location: Offset 2256, Length 2	TRUE
10	Context: ...ansform the data and select features. 2.Improve performance algorithmically, filter alg..., Location: Offset 2389, Length 7	TRUE

## Error calculation:

Table 5.4 Error calculation for assignment 8

Detailed Calculation:
Total Words: 383
Spelling Error Percentage = (Number of Spelling Errors / Total Words) * 100 = (11 / 383) * 100 = 2.8720626631853787% Grammar Error Percentage = (Number of Grammar Errors / Total Words) * 100 = (10 / 383) * 100 = 2.610966057441253%
Round off:
<b>Lexical Performance:</b> <hr/> Spelling Error: 3% Grammar Error: 3%

## Confusion Matrix

*Table 5.5 Confusion Matix for assignment 8*

Total: 383		Actual	
		Positive	Negative
Predicted	Positive	16	5
	Negative	1	361

$$\text{Accuracy rate} = \frac{361 + 16}{383} \times 100\% = \mathbf{98.43\%}$$

$$\text{False Positive Rate (FPR)} = \frac{5}{366} \times 100\% = \mathbf{1.37\%}$$

$$\text{Precision rate} = \frac{16}{21} \times 100\% = \mathbf{76.19\%}$$

$$\text{Recall rate} = \frac{16}{17} \times 100\% = \mathbf{94.12\%}$$

### 5.4..2 Assignment 9:

#### Spelling Error:

*Table 5.6 Spelling error for assignment 9*

No.	Text	Result
1	Context: ... algorithms you are going to run on the Freq-PHOBinary and Norm-PHO-Binary respectively. Repor..., Location: Offset 1664, Length 14	TRUE
2	Context: ... going to run on the Freq-PHOBinary and Norm-PHO-Binary respectively. Report the machine learni..., Location: Offset 1683, Length 15	TRUE
3	Context: ... distance] algorithm: [auto, ball_tree, kd_tree, brute] p: [1-3] Decision Tree Sup..., Location: Offset 2245, Length 2	TRUE
4	Context: ...ctor Machine Hyperparameter criterion: [gini, entropy] C = [0.1, 1, 10, 100] splitte..., Location: Offset 2334, Length 4	TRUE
5	Context: ... [best, random] kernel = [linear, poly, rbf, sigmoid] max_depth: [1, 2, 3, 4, 5, 6,..., Location: Offset 2420, Length 3	TRUE
6	Context: ...dentify the best performing model using Freq-PHO-Binary and also the best performing model usin..., Location: Offset 2708, Length 15	FALSE
7	Context: ...nd also the best performing model using Norm-PHO-Binary (both the best performing models should..., Location: Offset 2765, Length 15	FALSE
8	Context: ...; degree=1; 0.636 0.488 gamma=auto Norm DecisionTree criterion=gini; splitter=best; 0.615 0...., Location: Offset 3217, Length 12	TRUE
9	Context: ... gamma=auto Norm DecisionTree criterion=gini; splitter=best; 0.615 0.502 max_depth=7..., Location: Offset 3240, Length 4	TRUE

#### Grammar Error:

*Table 5.7 Grammar error for assignment 9*

No.	Text	Result
1	Context: ...echnique is used. The data used is small and the small data size will directly affec..., Location: Offset 1101, Length 4	TRUE
2	Context: ... student to another will be imbalanced. o Noises are included if record all emoti..., Location: Offset 5518, Length 1	FALSE
3	Context: ...hanges. Having more data help the model learns the pattern well and predict the target..., Location: Offset 5604, Length 6	TRUE

### Error calculation:

*Table 5.8 Error Calculation for assignment 9*

Detailed Calculation:
<b>Total Words: 1007</b>
Spelling Error Percentage = (Number of Spelling Errors / Total Words) * 100 = (9 / 1007) * 100 = 0.893743793445879% Grammar Error Percentage = (Number of Grammar Errors / Total Words) * 100 = (3 / 1007) * 100 = 0.29791459781529295%
Round off:
<b>Lexical Performance:</b>  Spelling Error: 1% Grammar Error: 0%

### Confusion Matrix

*Table 5.9 Confusion Matrix for assignment 9*

Total: 1007		Actual	
		Positive	Negative
Predicted	Positive	9	3
	Negative	12	983

$$\text{Accuracy rate} = \frac{9 + 983}{1007} \times 100\% = \mathbf{98.51\%}$$

$$\text{False Positive Rate (FPR)} = \frac{3}{986} \times 100\% = \mathbf{0.30\%}$$

$$\text{Precision rate} = \frac{9}{12} \times 100\% = \mathbf{75\%}$$

$$\text{Recall rate} = \frac{9}{21} \times 100\% = \mathbf{42.86\%}$$

### 5.4..3 Assignment 10:

#### Spelling Error:

*Table 5.10 Spelling error for assignment 10*

No.	Text	Result
1	Context: ...to run in this assignment are K Nearest Neighbour, Naive Bayes and Support Vector Machine..., Location: Offset 3376, Length 9	TRUE
2	Context: .... These three algorithms are run on the Freq-PHO-binary and Norm-PHO-Binary respectively. The f..., Location: Offset 3465, Length 15	FALSE
3	Context: ...thms are run on the Freq-PHO-binary and Norm-PHO-Binary respectively. The first step is data cl..., Location: Offset 3485, Length 15	FALSE
4	Context: ...der and depression which is categorical nominal data. Next, we use value count function..., Location: Offset 3874, Length 8	TRUE
5	Context: ...cy. It indicates the model suffers from underfitting. The result shows that model is sufferi..., Location: Offset 6410, Length 12	FALSE

#### Grammar Error:

*Table 5.11 Grammar error for assignment 10*

No.	Text	Result
1	Context: ...refully and answer the questions below: a. Report the class distribution. Is this ..., Location: Offset 64, Length 2	FALSE
2	Context: ... this assignment, the target feature is ‘Depression’, with “No” represents no dep..., Location: Offset 656, Length 1	TRUE
3	Context: ...nment, the target feature is ‘Depression’, with “No” represents no depression and..., Location: Offset 667, Length 1	TRUE
4	Context: ...se k- fold cross validation. Percentage split splits the data into training, validation, and..., Location: Offset 2499, Length 12	TRUE
5	Context: ...g fold. The advantage of this method is less data is required but needs more computa..., Location: Offset 2934, Length 4	TRUE
6	Context: ...s suffering from high bias low variance.T The high bias low variance suggests the..., Location: Offset 6493, Length 1	TRUE
7	Context: ...s to an exponent. It can help the model fits data better. For example, a new column ..., Location: Offset 7143, Length 4	FALSE

### Error calculation:

*Table 5.12 Error Calculation for assignment 10*

Detailed Calculation:
<pre>Total Words: 1200  Spelling Error Percentage = (Number of Spelling Errors / Total Words) * 100 = (5 / 1200) * 100 = 0.416666666666667% Grammar Error Percentage = (Number of Grammar Errors / Total Words) * 100 = (7 / 1200) * 100 = 0.583333333333334%</pre>
Round off:
<b>Lexical Performance:</b> <hr/> <p>Spelling Error: 0%</p> <p>Grammar Error: 1%</p>

### Confusion Matrix

*Table 5.13 Confusion Matrix for assignment 10*

Total: 1200		Actual	
		Positive	Negative
Predicted	Positive	7	5
	Negative	2	1186

$$\text{Accuracy rate} = \frac{7 + 1186}{1200} \times 100\% = \mathbf{99.42\%}$$

$$\text{False Positive Rate (FPR)} = \frac{5}{1191} \times 100\% = \mathbf{0.42\%}$$

$$\text{Precision rate} = \frac{7}{12} \times 100\% = \mathbf{58.33\%}$$

$$\text{Recall rate} = \frac{7}{9} \times 100\% = \mathbf{77.78\%}$$

### 5.4.2. Feedback Suggestion Generation

The evaluation for the feedback suggestion generation uses 3 new assignments based on the historical comments of 7 existing assignments. In the existing 7 assignments, there are a total of 74 comments which are separated in the list below:

- Assignment 1: 9 comments
- Assignment 2: 10 comments
- Assignment 3: 10 comments
- Assignment 4: 14 comments
- Assignment 5: 13 comments
- Assignment 6: 7 comments
- Assignment 7: 11 comments

The evaluation is done by evaluating the highlighted text, the actual feedback and the top 3 predicted feedback. If there is any feedback among the top 3 feedback that has a similar context with the actual feedback, the result will be considered as accurate; If none of the predicted feedback has the similar feedback, then it will be considered as not accurate. Out of 28 new comments, 15 comments are accurate, and 13 comments are inaccurate.

The summary of the 3 new assignments can be shown in the table below:

*Table 5.14 Summary of 3 new assignments*

	Comments	Accurate	Not Accurate
<b>Assignment 8:</b>	7	4	3
<b>Assignment 9:</b>	9	4	5
<b>Assignment 10:</b>	12	7	5
<b>Total</b>	28	15	13

The overall accuracy of this evaluation is shown as below:

$$\text{Accuracy} = \frac{15}{28} = 0.5357 = 53.57\%$$

The detailed evaluation is shown in the next page.

Figure 5.17 Before evaluation.

Assignment 1 Classification							
Submission Status							
Submission		Grading Progress					
First Name	Last Name	Email Address	Student ID	Grading	Grade	Download	Release Submission
Amirah	Abdullah	amirah.abdullah@student.usm.my	150001	Not graded	52 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Hafiz	Ibrahim	hafiz.ibrahim@student.usm.my	150002	Not graded	76 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Nurul	Aiziz	nurul.aiziz@student.usm.my	150003	Not graded	68 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Muhammad	Hassan	muhammad.hassan@student.usm.my	150004	Not graded	64 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Siti	Mohamad	siti.mohamad@student.usm.my	150005	Not graded	52 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Azim	Wong	azim.wong@student.usm.my	150006	Not graded	48 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Fatin	Tan	fatin.tan@student.usm.my	150007	Not graded	50 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Aisyah	Lim	aisyah.lim@student.usm.my	150008	Not graded	-	<a href="#">Download</a>	<a href="#">Release Submission</a>
Syahir	Lee	syahir.lee@student.usm.my	150009	Not graded	-	<a href="#">Download</a>	<a href="#">Release Submission</a>
Amir	Ng	amir.ng@student.usm.my	150010	Not graded	-	<a href="#">Download</a>	<a href="#">Release Submission</a>

Figure 5.18 After evaluation.

Assignment 1 Classification							
Submission Status							
Submission		Grading Progress					
First Name	Last Name	Email Address	Student ID	Grading	Grade	Download	Release Submission
Amirah	Abdullah	amirah.abdullah@student.usm.my	150001	Graded	52 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Hafiz	Ibrahim	hafiz.ibrahim@student.usm.my	150002	Graded	76 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Nurul	Aiziz	nurul.aiziz@student.usm.my	150003	Graded	68 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Muhammad	Hassan	muhammad.hassan@student.usm.my	150004	Graded	64 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Siti	Mohamad	siti.mohamad@student.usm.my	150005	Graded	52 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Azim	Wong	azim.wong@student.usm.my	150006	Graded	48 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Fatin	Tan	fatin.tan@student.usm.my	150007	Graded	50 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Aisyah	Lim	aisyah.lim@student.usm.my	150008	Graded	34 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Syahir	Lee	syahir.lee@student.usm.my	150009	Graded	66 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>
Amir	Ng	amir.ng@student.usm.my	150010	Graded	60 / 100	<a href="#">Download</a>	<a href="#">Release Submission</a>

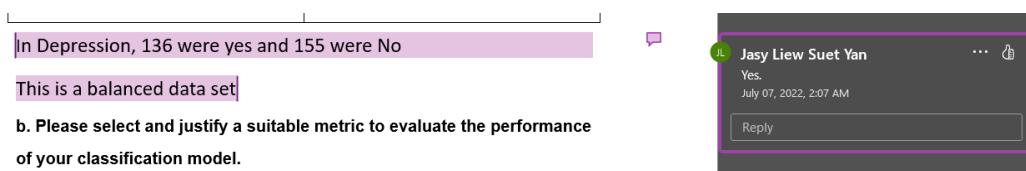
### 5.4..1 Assignment 8:

In the 8th assignment, we can identify that out of 7 comments, 4 is considered as accurate, while 3 does not provide accurate results. The accuracy for this assignment is 57.14%. The evaluation can be seen below:

Table 5.15 Assignment 8 evaluation for feedback suggestion generation

Text	Actual feedback	Predicted feedback	Result
In Depression, 136 were yes and 155 were No This is a balanced data set	yes	<p>Class Distribution</p> <p><b>Considered to be balanced.</b></p> <p><b>The dataset can be considered roughly balanced</b></p>	Accurate

Actual Feedback:



Predicted Feedback:

A screenshot of a software interface for generating feedback suggestions. At the top, there are two text boxes: one containing "In Depression, 136 were yes and 155 were No" and another containing "This is a balanced data set". Below these is a large text area with the following content:

```

evaluate the performance
ance of classification
of which is to count the
predicted (classified) to

```

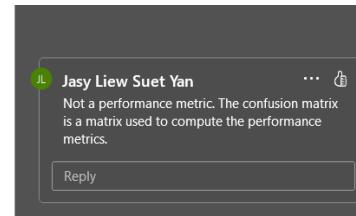
At the bottom of this area are "Add" and "Cancel" buttons. To the left of the text area, there is a "Class Distribution" section with the text "Considered to be balanced." and "The dataset can be considered roughly balanced".

Text	Actual feedback	Predicted feedback	Result
Confusion Matrix	<p>Not a performance metric.</p> <p>The confusion matrix is a matrix used to compute the performance metrics.</p>	<p><b>The confusion matrix is not a performance metric. It is a matrix used to compute the performance metrics.</b></p>	Accurate
	<p>Select one primary performance metric and justification</p>		
	<p>Is this the F1 score for just the positive class or macro-F1 or weighted-F1?</p>		

## Actual Feedback:

The method to evaluate the performance of classification models is **confusion matrix**, the general idea of which is to count the number of times category A instances are predicted (classified) to be category B.

Freq:



## Predicted Feedback:

The method to evaluate the performance of classification models is **confusion matrix**, the general idea of which is to count the number of times category A instances are predicted (classified) to be category B.

Freq:

The confusion matrix is not a performance metric. It is a matrix used to compute the performance metrics.

Select one primary performance metric and justification

Is this the F1 score for just the positive class or macro-F1 or weighted-F1?

Text	Actual feedback	Predicted feedback	Result
When choosing between two different groups of accuracy and recall, consider the F-1 Score combining both	Consider F1 as your primary performance metric. Justify why you have selected F1.	Are these the results for Freq or Norm?	Not Accurate
		Incorrect. F1 is the harmonic mean between precision and recall.	
		Did not report recall, precision and F1 in the experiment sheet.	

Actual Feedback:

is The balance between Precision and recall. When choosing between two different groups of accuracy and recall, consider the F-1 Score combining both. so, SVM is the best model.

Jasy Liew Suet Yan  
Consider F1 as your primary performance metric.  
Justify why you have selected F1.  
July 07, 2022, 2:12 AM  
Reply

Predicted Feedback:

is The balance between Precision and recall. When choosing between two different groups of accuracy and recall, consider the F-1 Score combining both. so, SVM is the best model.

Add Cancel

Are these the results for Freq or Norm?

Incorrect. F1 is the harmonic mean between precision and recall.

Did not report recall, precision and F1 in the experiment sheet.

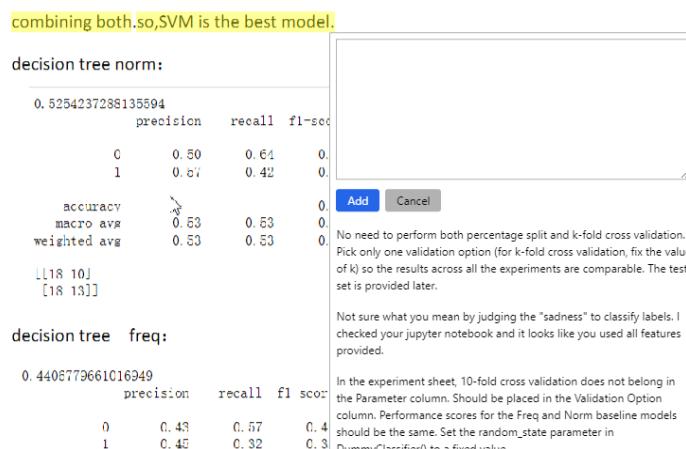
Support	28
	31
	69
	69
	59

Text	Actual feedback	Predicted feedback	Result
	<p>How did you come to the conclusion that SVM is the best performing model? Norm or Freq? Provide more detailed descriptions of the experiments and results leading to your conclusion of the best performing model.</p>	<p>No need to perform both percentage split and k-fold cross validation. Pick only one validation option (for k-fold cross validation, fix the value of k) so the results across all the experiments are comparable. The test set is provided later.</p> <p>Not sure what you mean by judging the "sadness" to classify labels. I checked your jupyter notebook and it looks like you used all features provided.</p> <p>In the experiment sheet, 10-fold cross validation does not belong in the Parameter column. Should be placed in the Validation Option column.</p> <p>Performance scores for the Freq and Norm baseline models should be the same. Set the random_state parameter in DummyClassifier() to a fixed value.</p>	Not Accurate

Actual Feedback:



Predicted Feedback:



Text	Actual feedback	Predicted feedback	Result
	<p>To achieve best performing model able to achieve very promising results</p> <p>Performance scores for the Freq and Norm baseline models should be the same. Set the random_state parameter in DummyClassifier() to a fixed value.</p> <p>LightBGM is not a supervised learning algorithm covered in the course.</p>	<p>Can only determine if underfitting or overfitting occurs if you compare training recall and validation recall.</p> <p>In the experiment sheet, 10-fold cross validation does not belong in the Parameter column. Should be placed in the Validation Option column.</p> <p><b>Performance scores for the Freq and Norm baseline models should be the same. Set the random_state parameter in DummyClassifier() to a fixed value.</b></p> <p>Cannot verify from your experiment sheet. There are missing values.</p>	Accurate

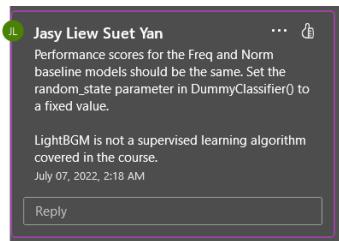
## Actual Feedback:

To achieve best performing model able to achieve very promising results.

Suggestions:

1.In terms of data, increase the number of data, collect more real data, generate more data, transform the data and select features.

2.Improve performance algorithmically, filter algorithms, improve



Jasy Liew Suet Yan  
Performance scores for the Freq and Norm baseline models should be the same. Set the random\_state parameter in DummyClassifier() to a fixed value.  
LightBGM is not a supervised learning algorithm covered in the course.  
July 07, 2022, 2:18 AM

## Predicted Feedback:

To achieve best performing model able to achieve very promising results.

Suggestions:

- 1.In terms of data, increase the number of data, collect more real data, generate more data, transform the data and select features.
- 2.Improve performance algorithmically, filter algorithms, improve performance from algorithm tuning.

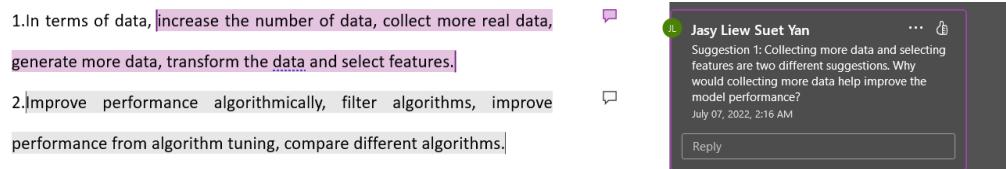
Can only determine if underfitting or overfitting occurs if you compare training recall and validation recall.

In the experiment sheet, 10-fold cross validation does not belong in the Parameter column. Should be placed in the Validation Option column. Performance scores for the Freq and Norm baseline models should be the same. Set the random\_state parameter in DummyClassifier() to a fixed value.

Cannot verify from your experiment sheet. There are missing values.

Text	Actual feedback	Predicted feedback	Result
In terms of data, increase the number of data, collect more real data, generate more data, transform the data and select features.	Collecting more data and selecting features are two different suggestions. Why would collecting more data help improve the model performance?	<p>Yes</p> <p><b>This is not a legit reason why collecting more data can help improve the performance of the model.</b></p> <p>The dataset can be considered roughly balanced</p>	Accurate

Actual Feedback:



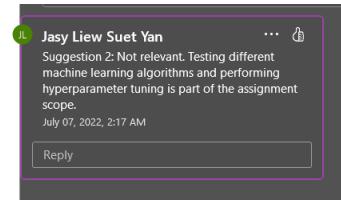
Predicted Feedback:

<p>1.In terms of data, <b>increase the number of data, collect more real data, generate more data, transform the data and select features.</b></p> <p><b>Add</b> <b>Cancel</b></p> <p>Yes</p> <p>This is not a legit reason why collecting more data can help improve the performance of the model.</p> <p>The dataset can be considered roughly balanced</p>	<p>2.Improve performance algorithmically, filter algorithms, improve performance from algorithm tuning, compare different algorithms.</p>
---	---

Text	Actual feedback	Predicted feedback	Result
Improve performance algorithmically, filter algorithms, improve performance from algorithm tuning, compare different algorithms.	Not relevant. Testing different machine learning algorithms and performing hyperparameter tuning is part of the assignment scope.	This should be part of the scope of the assignment.  Also no evidence of high variance. More training data can help the model learn more pertinent patterns from the data.  Ok but can you explain why the model would work out more accurately with more training data?	Not Accurate

## Actual Feedback:

2. Improve performance algorithmically, filter algorithms, improve performance from algorithm tuning, compare different algorithms.



## Predicted Feedback:

2. Improve performance algorithmically, filter algorithms, improve performance from algorithm tuning, compare different algorithms.

Not relevant. Testing different machine learning algorithms and performing hyperparameter tuning is part of the assignment scope

Add Cancel

This should be part of the scope of the assignment.  
Also no evidence of high variance. More training data can help the model learn more pertinent patterns from the data.  
Ok but can you explain why the model would work out more accurately with more training data?

### 5.4..2 Assignment 9:

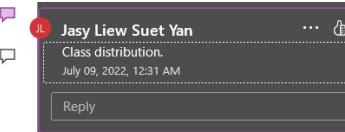
In the 9th assignment, we can identify that out of 9 comments, 4 is considered as accurate, while 5 does not provide accurate results. The accuracy for this assignment is 44.44%. The evaluation can be seen below:

*Table 5.16 Assignment 9 evaluation for feedback suggestion generation*

Text	Actual feedback	Predicted feedback	Result
Figure 1 shows 136 individuals (46.7%) who have depression and 155 individuals (53.3%)	Class distribution.	<b>Class Distribution.</b>  Yes  Considered to be balanced.	Accurate

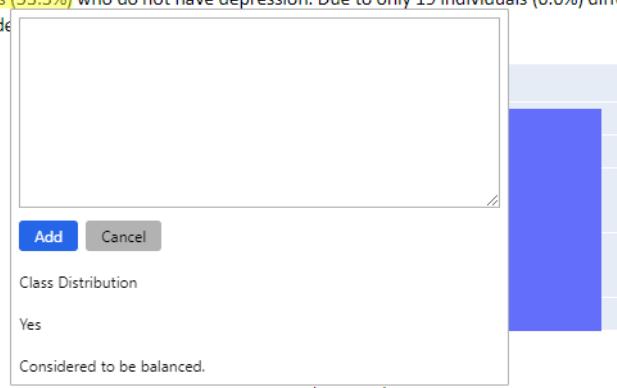
#### Actual Feedback:

It is a balanced dataset. Figure 1 shows 136 individuals (46.7%) who have depression and 155 individuals (53.3%) who do not have depression. Due to only 19 individuals (6.6%) difference, it is still considered a balanced dataset.



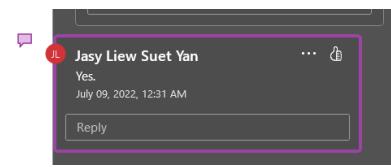
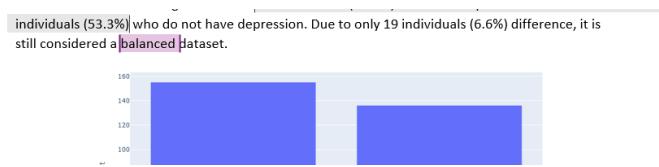
#### Predicted Feedback:

It is a balanced dataset. Figure 1 shows 136 individuals (46.7%) who have depression and 155 individuals (53.3%) who do not have depression. Due to only 19 individuals (6.6%) difference, it is still consider



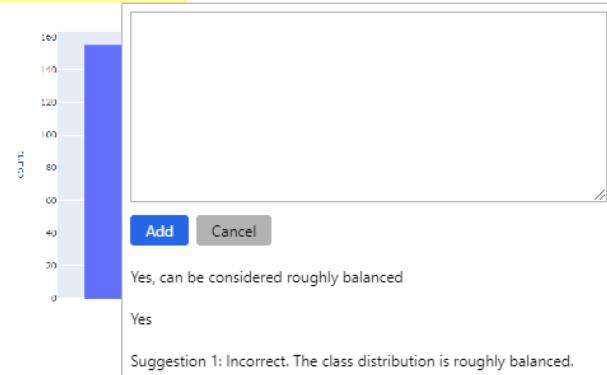
Text	Actual feedback	Predicted feedback	Result
it is still considered a balanced dataset.	Yes.	Yes, can be considered roughly balanced	Accurate
		Yes	
		Suggestion 1: Incorrect. The class distribution is roughly balanced.	

## Actual Feedback:



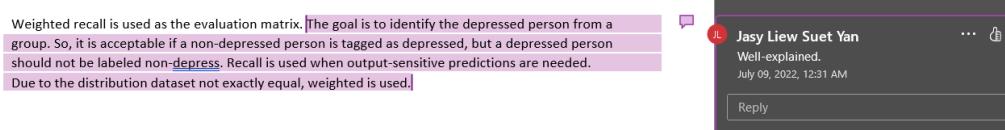
## Predicted Feedback:

individuals (53.3%) who do not have depression. Due to only 19 individuals (6.6%) difference, it is still considered a balanced dataset.



Text	Actual feedback	Predicted feedback	Result
<p>The goal is to identify the depressed person from a group. So, it is acceptable if a non-depressed person is tagged as depressed, but a depressed person should not be labeled non-depress.</p> <p>Recall is used when output-sensitive predictions are needed. Due to the distribution dataset not exactly equal, weighted is used.</p>	<p>Well-explained.</p>	<p><b>Yes</b></p> <p>Justification does not make sense. Accuracy is also suitable when the class distributions is balanced so why did you not choose accuracy?</p>	<p>Accurate</p>
		<p>Yes but how does this help improve the model's performance?</p>	

Actual Feedback:



Predicted Feedback:

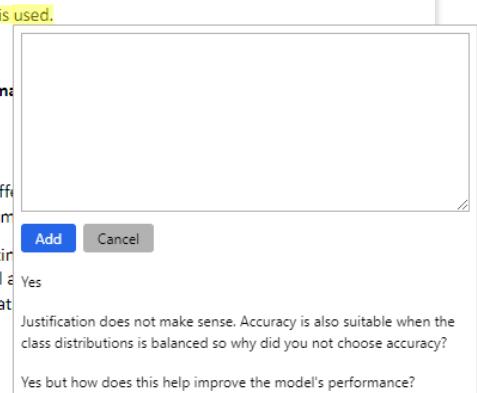
Weighted recall is used as the evaluation matrix. The goal is to identify the depressed person from a group. So, it is acceptable if a non-depressed person is tagged as depressed, but a depressed person should not be labeled non-depress. Recall is used when output-sensitive predictions are needed. Due to the distribution dataset not exactly equal, weighted is used.

c) Specify the validation option you are selecting for your model. Explain the reason for using the validation option.

5-fold cross-validation technique is used.

The data used is small and the small data size will directly affect the validation. k-fold validation is used to avoid the overfitting and bias that may occur.

In theory, a higher k value leads to less bias and less overfitting. However, the runtime and the computational cost will increase. 5 is chosen as the K value to balance the computation time and the validation accuracy.



Text	Actual feedback	Predicted feedback	Result
K value to balance the computation cost and the overfitting issue.	<p>Ok. Another food for thought. For such a small dataset, if you use say 10-fold cross validation, the subset used for validation in each fold contains only 29 instances. Such a small set will cause the performance values to fluctuate for quite a bit across the folds. Smaller k can lead to more stable values.</p>	Can only determine if underfitting or overfitting occurs if you compare training recall and validation recall.	Not Accurate
		Present your top results for comparison in a table. Need to make a conclusion which is the ONE best model.	
		Cannot make such conclusion as the comparison should be done on training accuracy versus validation accuracy. Same comment applies to all similar statements you made below.	

## Actual Feedback:

The screenshot shows a messaging interface. A user message from "Jasy Liew Suet Yan" is displayed, containing the text: "Ok. Another food for thought. For such a small dataset, if you use say 10-fold cross validation, the subset used for validation in each fold contains only 29 instances. Such a small set will cause the performance values to fluctuate for quite a bit across the folds. Smaller k can lead to more stable values." Below the message is a timestamp: "July 09, 2022, 12:43 AM". There is also a "Reply" button at the bottom of the message box.

## Predicted Feedback:

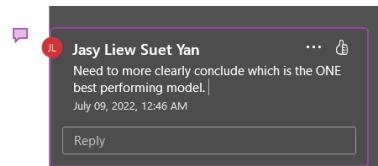
The screenshot shows a software application window with a title bar and a main content area. The content area contains several sections of text and buttons:

- b) Please select and justify a model.** This section includes a note: "Weighted recall is used as the group. So, it is acceptable if a group should not be labeled non-deaf. Due to the distribution data". Buttons for "Add" and "Cancel" are present.
- c) Specify the validation option explain the reason for using 5-fold cross-validation technique.** This section includes a note: "The data used is small and the fold validation is used to avoid overfitting. In theory, a higher k value leads to better generalization. However, the runtime and the computational cost will also increase when the value of k increases. 5 is chosen as the K value to balance the computation cost and the overfitting issue." A note below states: "Can only determine if underfitting or overfitting occurs if you compare training recall and validation recall."
- A note: "Present your top results for comparison in a table. Need to make a conclusion which is the ONE best model."
- A note: "Cannot make such conclusion as the comparison should be done on training accuracy versus validation accuracy. Same comment applies to all similar statements you made below."

Text	Actual feedback	Predicted feedback	Result
SVM in frequency representation performs slightly better	Need to more clearly conclude which is the ONE best performing model.	<p><b>How did you come to the conclusion that SVM is the best performing model?</b> Norm or Freq? Provide more detailed descriptions of the experiments and results leading to your conclusion of the best performing model.</p> <p>Good Observation!</p> <p>Considered to be balanced.</p>	Accurate

## Actual Feedback:

Between the two best performing model in the frequency and normalized representation, the performance of **SVM in frequency representation performs slightly better**. Due to there is no significant difference between the model performance in frequency representation and the normalized representation, a threshold of 60% of validation recall is set as the baseline. There have 13 hyperparameter sets in frequency representation able to hit this target whereas only 7 hyperparameter sets in normalized representation can this target. Overall, the frequency presentation produces a better model.



## Predicted Feedback:

Identify the best using Norm-PHO your random ba

Table 2 shows th best performing

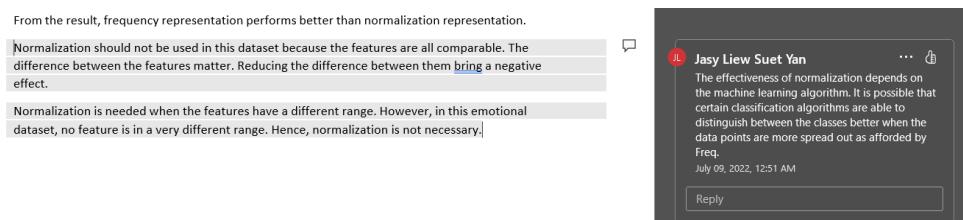
Feature Represent	Add	Cancel
Freq	How did you come to the conclusion that SVM is the best performing model? Norm or Freq? Provide more detailed descriptions of the experiments and results leading to your conclusion of the best performing model.	
Norm	Good Observation!	

Which feature r Considered to be balanced.

Between the two best performing model in the frequency and normalized representation, the performance of **SVM in frequency representation performs slightly better**. Due to there is no

Text	Actual feedback	Predicted feedback	Result
<p>Normalization should not be used in this dataset because the features are all comparable. The difference between the features matter. Reducing the difference between them bring a negative effect.</p> <p>Normalization is needed when the features have a different range. However, in this emotional dataset, no feature is in a very different range. Hence, normalization is not necessary.</p>	<p>The effectiveness of normalization depends on the machine learning algorithm. It is possible that certain classification algorithms are able to distinguish between the classes better when the data points are more spread out as afforded by Freq.</p>	<p>Inaccurate. Normalization does not guarantee to reduce the trained model error.</p>	Not Accurate
		<p>Why would this be helpful? Justify.</p>	
		<p>Again no evidence of high variance. Outliers and the number of dimensions mean two different things.</p>	

## Actual Feedback:



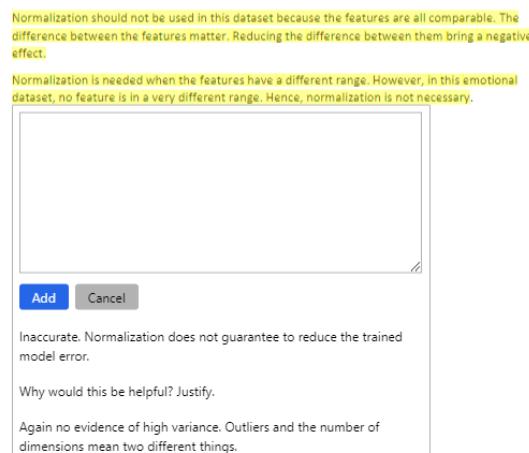
From the result, frequency representation performs better than normalization representation.

Normalization should not be used in this dataset because the features are all comparable. The difference between the features matter. Reducing the difference between them bring a negative effect.

Normalization is needed when the features have a different range. However, in this emotional dataset, no feature is in a very different range. Hence, normalization is not necessary.

Jasy Liew Suet Yan ... 12:51 AM  
The effectiveness of normalization depends on the machine learning algorithm. It is possible that certain classification algorithms are able to distinguish between the classes better when the data points are more spread out as afforded by Freq.  
July 09, 2022, 12:51 AM  
Reply

## Predicted Feedback:



Normalization should not be used in this dataset because the features are all comparable. The difference between the features matter. Reducing the difference between them bring a negative effect.

Normalization is needed when the features have a different range. However, in this emotional dataset, no feature is in a very different range. Hence, normalization is not necessary.

Add Cancel

Inaccurate. Normalization does not guarantee to reduce the trained model error.

Why would this be helpful? Justify.

Again no evidence of high variance. Outliers and the number of dimensions mean two different things.

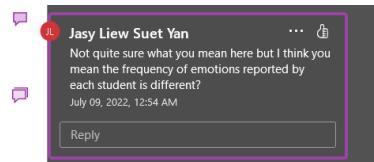
Text	Actual feedback	Predicted feedback	Result
The dataset from one student to another will be imbalanced.	Not quite sure what you mean here but I think you mean the frequency of emotions reported by each student is different?	Inaccurate. Normalization does not guarantee to reduce the trained model error.	Not Accurate
		Considered to be balanced.	
		Suggestion 1: Incorrect. The class distribution is roughly balanced.	

Actual Feedback:

There are two problems:

- o The dataset from one student to another will be imbalanced.
- o Noises are included if record all emotions changes.

Having more data help the model learns the pattern well and predict the target value correctly. However, if the student records every emotional change and some are just tiny little things that the student will forget very soon, those data will be considered as noises. Hence, it is better if every student only records 1 to 2 emotions per day. If emotion changes frequently that day, the student only needs to choose the emotion that impacts him/her the most.



Predicted Feedback:

There are two problems:

- o The dataset from one student to another will be imbalanced.
- o Noises are included if record all emotions changes.

Having more data help the model learns the pattern well correctly. However, if the student records every emotional little things that the student will forget very soon, those. Hence, it is better if every student only records 1 to 2 emotions frequently that day, the student only needs to choose the most.

- Feature Selection.  
There are eight types of emotion recorded in this database relevant and some may bring a negative impact on the model to filter those emotions that are irrelevant or did not contribute to the prediction. This process not only increases the accuracy of the model.

Add Cancel

Inaccurate. Normalization does not guarantee to reduce the trained model error.

Considered to be balanced.

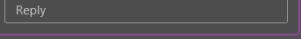
Suggestion 1: Incorrect. The class distribution is roughly balanced.

Text	Actual feedback	Predicted feedback	Result
those data will be considered as noises	I would use the term noise here. Noise usually means meaningless data. Perhaps the more appropriate term is data inconsistency.	Considered to be balanced.	Not Accurate
		Collecting more data and selecting features are two different suggestions. Why would collecting more data help improve the model performance?	
	Yes		

## Actual Feedback:

Correctly, however, if the student records every emotional change and some are just tiny little things that the student will forget very soon, those data will be considered as noises. Hence, it is better if every student only records 1 to 2 emotions per day. If emotion changes frequently that day, the student only needs to choose the emotion that impacts him/her the most.

- Feature Selection.  
There are eight types of emotion recorded in this dataset. Not all these emotions are relevant and some may bring a negative impact on the model. Domain knowledge is needed

Jasy Liew Suet Yan ... 

I would use the term noise here. Noise usually means meaningless data. Perhaps the more appropriate term is data inconsistency.  
July 09, 2022, 12:56 AM

Add Cancel

Considered to be balanced.

Collecting more data and selecting features are two different suggestions. Why would collecting more data help improve the model performance?

Yes

Text	Actual feedback	Predicted feedback	Result
If emotion changes frequently that day, the student only needs to choose the emotion that impacts him/her the most.	I am interpreting your first suggestions as standardizing the data collection procedure.	Inaccurate. Normalization does not guarantee to reduce the trained model error.	Not Accurate
		Suggestion 1: More data does not necessarily mean stronger correlation.	
		Why would accuracy be a suitable performance measure for this problem?	

## Actual Feedback:

Hence, it is better if every student only records 1 to 2 emotions per day. If emotion changes frequently that day, the student only needs to choose the emotion that impacts him/her the most.

- Feature Selection.  
There are eight types of emotion recorded in this dataset. Not all these emotions are relevant and some may bring a negative impact on the model. Domain knowledge is needed to filter those emotions that are irrelevant or did not contribute more to the target value.

Jasy Liew Suet Yan ...  
Suggestion 1: I am interpreting your first suggestions as standardizing the data collection procedure.  
July 09, 2022, 12:57 AM  
Reply

## Predicted Feedback:

Inaccurate things that the student will forget very soon, those data will be considered as noises. Hence, it is better if every student only records 1 to 2 emotions per day. If emotion changes frequently that day, the student only needs to choose the emotion that impacts him/her the most.

Add Cancel

Inaccurate. Normalization does not guarantee to reduce the trained model error.

Suggestion 1: More data does not necessarily mean stronger correlation.

Why would accuracy be a suitable performance measure for this problem?

#### 5.4..3 Assignment 10:

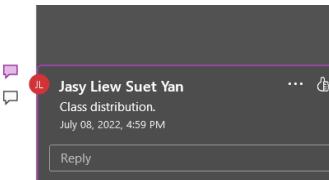
In the 10th assignment, we can identify that out of 12 comments, 7 is considered as accurate, while 5 does not provide accurate results. The accuracy for this assignment is 58.33 %. The evaluation can be seen below:

Table 5.17 Assignment 10 evaluation for feedback suggestion generation

Text	Actual feedback	Predicted feedback	Result
155 No and 136 Yes	Class distribution.  Yes  Considered to be balanced	Class distribution.	Accurate
		Yes	
		Considered to be balanced	

Actual Feedback:

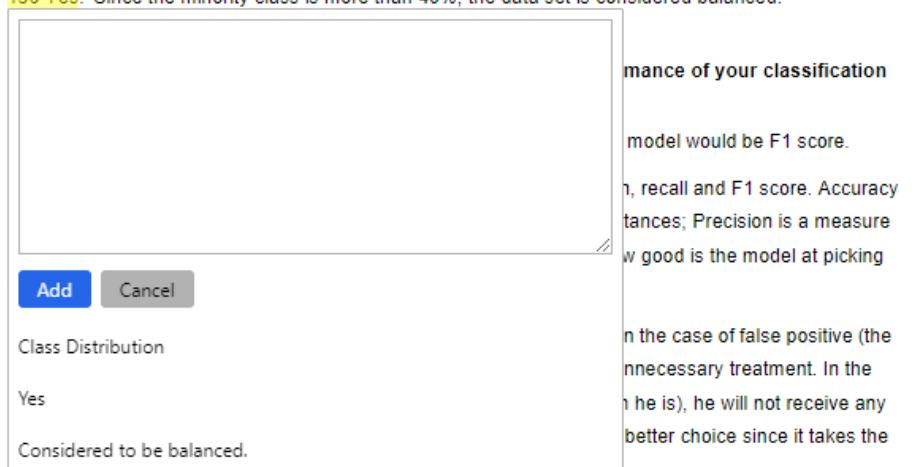
In this assignment, the target feature is 'Depression', with "No" represents no depression and "Yes" represents depression. It later transforms into 0 and 1 respectively using label encoder. To check the data distribution, the method "value count" is used on the Depression feature, getting the result of 155 No and 136 Yes. Since the minority class is more than 40%, the data set is considered balanced.



b. Please select and justify a suitable metric to evaluate the performance of your classification model.

Predicted Feedback:

distribution, the method "value count" is used on the Depression feature, getting the result of 155 No and 136 Yes. Since the minority class is more than 40%, the data set is considered balanced.



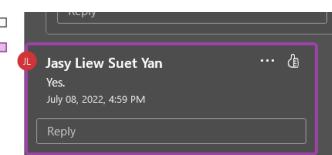
Text	Actual feedback	Predicted feedback	Result
the data set is considered balanced	Yes	Yes	Accurate
		Yes	
		The dataset can be considered roughly balanced	

## Actual Feedback:

distribution, the method "value count" is used on the Depression feature, getting the result of 155 No and 136 Yes. Since the minority class is more than 40%, the data set is considered balanced.

- b. Please select and justify a suitable metric to evaluate the performance of your classification model.

The suitable metric to evaluate the performance of classification model would be F1 score.



## Predicted Feedback:

distribution, the method "value count" is used on the Depression feature, getting the result of 155 NO and 136 Yes. Since the minority class is more than 40%, the data set is considered balanced.

- b. Please select and justify a suitable metric to evaluate the performance of your classification model.

The suitable metric to evaluate the performance of classification model would be F1 score.

There are four types of performance metrics:  
Precision is measure of how many instances are correctly predicted by the model.  
Recall is measure of how good is the model at whatever it predicted; Recall measures the proportion of the correct items; F1 is harmonic mean of precision and recall.

In this assignment our task is to predict the depression status of the patient. If the patient is diagnosed with Depression when he is not, it is a case of false positive (the patient is not depressed but is diagnosed as such). If the patient is not diagnosed with Depression when he is, it is a case of false negative (the patient is diagnosed as Normal but is actually depressed). We don't want any of the situation happen. Therefore, we need to find a metric which can measure both precision and recall. F1 score is harmonic mean between recall and precision.

Text	Actual feedback	Predicted feedback	Result
<p>In this assignment our task is to predict the depression patient. In the case of false positive (the patient is diagnosed with Depression when he is not), he will undergo unnecessary treatment. In the case of false negative (the patient is diagnosed as No Depression when he is), he will not receive any treatment. We don't want any of the situation happen, therefore F1 is a better choice since it takes the harmonic mean between recall and precision</p>	<p>You have provided convincing justification of your primary performance metric selection. Just to get you thinking a bit more... If the class distribution is roughly balanced, why not just go with accuracy?</p>	<p>Incorrect. F1 is the harmonic mean between precision and recall</p>	Not Accurate
	<p>Yes</p>		
	<p>Justification is not convincing</p>		

Actual Feedback:

In this assignment our task is to predict the depression patient. In the case of false positive (the patient is diagnosed with Depression when he is not), he will undergo unnecessary treatment. In the case of false negative (the patient is diagnosed as No Depression when he is), he will not receive any treatment. We don't want any of the situation happen, therefore F1 is a better choice since it takes the harmonic mean between recall and precision.

Jasy Liew Suet Yan ... ⓘ  
You have provided convincing justification of your primary performance metric selection. Just to get you thinking a bit more... If the class distribution is roughly balanced, why not just go with accuracy?  
July 08, 2022, 5:01 PM  
Reply

Predicted Feedback:

In this assignment our task is to predict the depression patient. In the case of false positive (the patient is diagnosed with Depression when he is not), he will undergo unnecessary treatment. In the case of false negative (the patient is diagnosed as No Depression when he is), he will not receive any treatment. We don't want any of the situation happen, therefore F1 is a better choice since it takes the harmonic mean between recall and precision.

Add Cancel

Incorrect. F1 is the harmonic mean between precision and recall.

Yes

Justification is not convincing.

Text	Actual feedback	Predicted feedback	Result
decreasing the performance of the model .	Not the most appropriate justification. Link it back to model being biased due to limited data.	Also no evidence of high variance. More training data can help the model learn more pertinent patterns from the data.	Not Accurate
		Why would accuracy be a suitable performance measure for this problem?	
		The confusion matrix is not a performance metric. It is a matrix used to compute the performance metrics.	

## Actual Feedback:

Only 291, no splitting of data will result in only small range of data been used and over usage, decreasing the performance of the model.

K-fold cross validation suggests that data is divided into K fold, one serves as training, remaining serve as testing and continue till every fold has been testing fold. The advantage of this method is less data is required but needs more computation power. Considering we have only 291 entries, it will be ideal in this case

## Predicted Feedback:

decreasing the performance of the model.

K-fold cross validation suggests that serve as testing and continue till every fold data is required but needs more computation ideal in this case

**2) Which feature representation produces a better performing model based on the performance of the model? Explain why one feature representation is better than the other.**

The classification algorithms that Naïve Bayes and Support Vector Machine use are Norm-PHO-Binary respectively.

The first step is data cleaning and there are missing values in the data set.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 291 entries, 0 to 290
Data columns (total 10 columns):
```

Add Cancel

Also no evidence of high variance. More training data can help the model learn more pertinent patterns from the data.

Why would accuracy be a suitable performance measure for this problem?

The confusion matrix is not a performance metric. It is a matrix used to compute the performance metrics.

Text	Actual feedback	Predicted feedback	Result
		<p>Need to fix on using only one value of k for k-fold cross validation to make your results comparable across all experiments.</p> <p>Just need one validation set. In your Jupyter notebook, no need to split to a dev set and then test set. The test set is provided later.</p> <p><b>What is the value of k you have selected for k-fold cross validation and why?</b></p> <p>No need to perform both percentage split and k-fold cross validation. Pick only one validation option (for k-fold cross validation, fix the value of k) so the results across all the experiments are comparable. The test set is provided later.</p>	Accurate

## Actual Feedback:

K-fold cross validation suggests that data is divided into K fold, one serves as training, remaining serve as testing and continue till every fold has been testing fold. The advantage of this method is less data is required but needs more computation power. Considering we have only 291 entries, it will be ideal in this case

Jasy Liew Suet Yan  
What is the value of k you have selected for k-fold cross validation and why?  
July 08, 2022, 5:03 PM  
Reply

## Predicted Feedback:

K-fold cross validation suggests that data is divided into K fold, one serves as training, remaining serve as testing and continue till every fold has been testing fold. data is required but needs more computation power. Considering ideal in this case

2) Which feature representation produces a better model? Explain performing model based on the performance metric you have feature representation is better than the other?

The classification algorithms that are going to run in this Naive Bayes and Support Vector Machine. These three algorithm Norm-PHO-Binary respectively.

The first step is data cleaning and data preparation. After is there any missing value in the data set using df.info() function.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 291 entries, 0 to 290
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   Gender            291 non-null    object  
 1   Emotion_Joy       291 non-null    float64 
 2   Emotion_Sadness   291 non-null    float64 
 3   Emotion_Anger     291 non-null    float64 
 4   Emotion_Disgust   291 non-null    float64 
 ..   ...              ...             ...    

```

Need to fix on using only one value of k for k-fold cross validation to make your results comparable across all experiments. Just need one validation set. In your Jupyter notebook, no need to split to a dev set and then test set. The test set is provided later.

What is the value of k you have selected for k-fold cross validation and why?

No need to perform both percentage split and k-fold cross validation. Pick only one validation option (for k-fold cross validation, fix the value of k) so the results across all the experiments are comparable. The test set is provided later.

Text	Actual feedback	Predicted feedback	Result
The K -fold cross validation is used to compute the accuracy, precision, recall and F1 score of the model.	<p>Need to fix on using only one value of k for k-fold cross validation to make your results comparable across all experiments.</p> <p>The goal is not to compare different values of k in k-fold cross validation. Performance improvement should not be a product of the validation option. Rather, it should come from the properties of the model.</p>	<p>What is the value of k you have selected for k-fold cross validation and why?</p> <p><b>Need to fix on using only one value of k for k-fold cross validation to make your results comparable across all experiments.</b></p> <p>Just need one validation set. In your Jupyter notebook, no need to split to a dev set and then test set. The test set is provided later.</p>	Accurate

## Actual Feedback:

Since the assignment mentions the test data will be released later, all data will be used as training set. The K-fold cross validation is used to compute the accuracy, precision, recall and F1 score of the model. The result is shown as below:

Feature Representat - Machine Learning Al - Parameters		Validation Option	Validation Accuracy	Validation Precision	Validation Recall (W - Validation F1 (Weigh : 1 Avg)
Freq	DummyClassifier	K-fold cross validation (k=10)	0.529	0.497	0.400
	NN	K-fold cross validation (k=11)	0.534	0.499	0.419
	Naive Bayes	K-fold cross validation (k=2)	0.567	0.526	0.575
	SVM	Linear, C = 100	0.623	0.702	0.595
		K-fold cross validation (k=8)	0.596	0.596	0.591
		Poly, C = 1	0.596	0.596	0.591
		Ridge, C = 1	0.549	0.441	0.367
		Sigmoid, C = 1	0.524	0.699	0.242
		K-fold cross validation (k=7)	0.513	0.479	0.429
		K-fold cross validation (k=10)	0.503	0.479	0.429
		NN	0.603	0.588	0.618
		Naive Bayes	0.591	0.531	0.575
		SVM	0.591	0.589	0.592
		Linear, C = 1	0.545	0.499	0.551
		Poly, C = 1	0.526	0.595	0.371
		Ridge, C = 1	0.426	0.032	0.598
		Sigmoid, C = 1	0.540	0.496	0.391

Jasy Liew Suet Yan ... ⓘ

Need to fix on using only one value of k for k-fold cross validation to make your results comparable across all experiments.

The goal is not to compare different values of k in k-fold cross validation. Performance improvement should not be a product of the validation option. Rather, it should come from the properties of the model.

July 08, 2022, 5:06 PM

Reply

## Predicted Feedback:

set. The K-fold cross validation is used to compute the accuracy, precision, recall and F1 score of the model. The result is shown as below:

Feature Representat - Validation Recal (W - Validation F1 (Weigh : 1 Avg)	
0.4371	0.4887
0.5371	0.419
0.5271	0.5728
0.7302	0.5465
0.84	0.5051
0.7441	0.192
0.699	0.367
0.4739	0.4429
0.9088	0.5052
0.5331	0.525
0.5991	0.4951
0.5758	0.6967
0.0302	0.5688
0.4966	0.3199
	0.3821

e. The best performing algorithm represented Model and SVM using F1 score, we know that the % (exclude Dummy Classifier) is score of 53% (exclude Dummy representation model also performs

What is the value of k you have selected for k-fold cross validation and why?

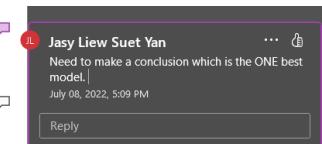
Need to fix on using only one value of k for k-fold cross validation to make your results comparable across all experiments. Just need one validation set. In your Jupyter notebook, no need to split to a dev set and then test set. The test set is provided later.

Did not report recall, precision and F1 in the experiment sheet.

Text	Actual feedback	Predicted feedback	Result
<p>The best performing algorithm has been highlighted in yellow which is Naïve Bayes in Frequency Feature Represented Model and SVM Polynomial in Normalized Feature Representation model.</p>	<p>Need to make a conclusion which is the ONE best model.</p>	<p><b>Need to more clearly conclude which is the ONE best performing model.</b></p> <p>How did you come to the conclusion that SVM is the best performing model? Norm or Freq? Provide more detailed descriptions of the experiments and results leading to your conclusion of the best performing model.</p> <p>Again justify why. Better to provide feature selection methods you have learnt as examples.</p>	<p>Accurate</p>

#### Actual Feedback:

The best performing model is determined by looking at F1 score. The **best** performing algorithm has been highlighted in yellow which is Naïve Bayes in Frequency Feature Represented Model and SVM Polynomial in Normalized Feature Representation model. By comparing F1 score, we know that the Normalized Feature Representation model with mean F1 score of 43% (exclude Dummy Classifier) is better than Frequency feature Represented Model with mean F1 score of 53% (exclude Dummy Classifier). The best performing algorithm in Normalized Feature Representation model also performs



#### Predicted Feedback:

The best performing model is determined by looking at F1 score. **The best performing algorithm has been highlighted in yellow which is Naïve Bayes in Frequency Feature Represented Model and SVM Polynomial in Normalized Feature Representation model.** By comparing F1 score, we know that the Normalized Feature Representation model with mean F1 score of 43% (exclude Dummy Classifier) is better than Frequency feature Represented Model with mean F1 score of 53% (exclude Dummy Classifier). The best performing algorithm in Normalized Feature Representation model also performs

feature allows the dataset to use the larger feature will affect result

Add Cancel

Need to more clearly conclude which is the ONE best performing model.

How did you come to the conclusion that SVM is the best performing model? Norm or Freq? Provide more detailed descriptions of the experiments and results leading to your conclusion of the best performing model.

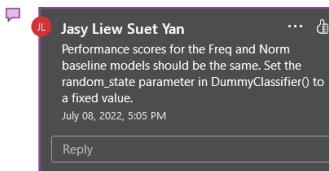
Again justify why. Better to provide feature selection methods you have learnt as examples.

Text	Actual feedback	Predicted feedback	Result
By comparing F1 score, we know that the Normalized Feature Representation model with mean F1 score of 43% (exclude Dummy Classifier) is better than Frequency feature Represented Model with mean F1 score of 53% (exclude Dummy Classifier).	Performance scores for the Freq and Norm baseline models should be the same. Set the random_state parameter in DummyClassifier() to a fixed value.	<b>Performance scores for the Freq and Norm baseline models should be the same. Set the random_state parameter in DummyClassifier() to a fixed value.</b>	Accurate
		Incorrect. F1 is the harmonic mean between precision and recall.	
		Is this the F1 score for just the positive class or macro-F1 or weighted-F1?	

#### Actual Feedback:

Normalized Feature Representation model with mean F1 score of 43% (exclude Dummy Classifier) is better than Frequency feature Represented Model with mean F1 score of 53% (exclude Dummy Classifier). The best performing algorithm in Normalized Feature Representation model also performs better than the one in Frequency Represented Model.

The normalized feature produces better result. The normalized feature allows the dataset to use a common scale. When one of the features is far greater than another, the larger feature will affect result



#### Predicted Feedback:

Polynomial in Normalized Feature Representation model. By comparing F1 score, we know that the Normalized Feature Representation model with mean F1 score of 43% (exclude Dummy Classifier) is better than Frequency feature Represented Model with mean F1 score of 53% (exclude Dummy Classifier). The best performing algorithm in Normalized Feature Representation model also performs

feature allows the dataset to use the larger feature will affect result

3

Add Cancel

Performance scores for the Freq and Norm baseline models should be the same. Set the random\_state parameter in DummyClassifier() to a fixed value.

Incorrect. F1 is the harmonic mean between precision and recall.

Is this the F1 score for just the positive class or macro-F1 or weighted-F1?

Text	Actual feedback	Predicted feedback	Result
<p>It indicates the model suffers from underfitting. The result shows that model is suffering from high bias low variance</p>	<p>There is no evidence comparing training error and validation or test error to prove this statement.</p>	<p>Can only determine if underfitting or overfitting occurs if you compare training recall and validation recall.</p>	Not Accurate
		<p>How did you come to the conclusion that SVM is the best performing model? Norm or Freq? Provide more detailed descriptions of the experiments and results leading to your conclusion of the best performing model.</p>	
		<p>Do not just provide a laundry list of performance metrics. Specify one primary performance metric and justify why you have selected it to measure the performance of your models.</p>	

#### Actual Feedback:

The best prediction model produces not more 60% of accuracy. It indicates the model suffers from underfitting. The result shows that model is suffering from high bias low variance.

The high bias low variance suggests the model is inaccurate but produce consistence result. The model is unable to learn the relationship between variables. In this case, adding more features will help the model to improve. This is due to increasing feature means the model has more data to learn from. Examples of features that can be added into this data set are demographic data such as age, gender,

Jasy Liew Suet Yan ... ⓘ

There is no evidence comparing training error and validation or test error to prove this statement.

July 08, 2022, 5:11 PM

Reply

#### Predicted Feedback:

The best prediction model produces not more 60% of accuracy. It indicates the model suffers from underfitting. The result shows that model is suffering from high bias low variance.

t produce consistence result. The e, adding more features will help del has more data to learn from. aphic data such as age, gender, (BDI).

features are features which are el fits data better. For example, a n\_Joy" into exponential of power

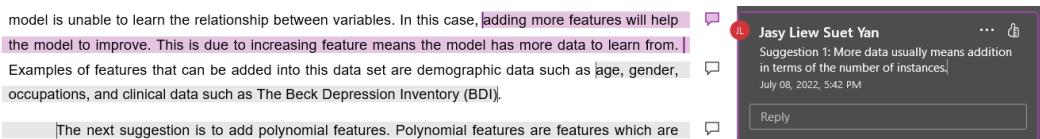
Can only determine if underfitting or overfitting occurs if you compare training recall and validation recall.

How did you come to the conclusion that SVM is the best performing model? Norm or Freq? Provide more detailed descriptions of the experiments and results leading to your conclusion of the best performing model.

Do not just provide a laundry list of performance metrics. Specify one primary performance metric and justify why you have selected it to measure the performance of your models.

Text	Actual feedback	Predicted feedback	Result
<p>In this case, adding more features will help the model to improve. This is due to increasing feature means the model has more data to learn from</p>	<p>More data usually means addition in terms of the number of instances.</p>	<p>Again, there is no evidence showing that your model is too simple. Also no evidence of high variance. More training data can help the model learn more pertinent patterns from the data. Need to make a conclusion which is the ONE best model.</p>	Not Accurate

Actual Feedback:



Predicted Feedback:

model is unable to learn the relationship between variables. In this case, adding more features will help the model to improve. This is due to increasing feature means the model has more data to learn from.

aphic data such as age, gender, (BDI).

features are features which are el fits data better. For example, a n\_Joy" into exponential of power

Add Cancel

Again, there is no evidence showing that your model is too simple.

Also no evidence of high variance. More training data can help the model learn more pertinent patterns from the data.

Need to make a conclusion which is the ONE best model.

Text	Actual feedback	Predicted feedback	Result
Examples of features that can be added into this data set are demographic data such as age, gender, occupations, and clinical data such as The Beck Depression Inventory (BDI)	Gender is already included as a feature. The class label is partially based on BDI so cannot be used as features.	Collecting more data and selecting features are two different suggestions. Why would collecting more data help improve the model performance?	Not Accurate
		The dataset can be considered roughly balanced	
	Yes		

## Actual Feedback:

Examples of features that can be added into this data set are demographic data such as age, gender, occupations, and clinical data such as The Beck Depression Inventory (BDI).

The next suggestion is to add polynomial features. Polynomial features are features which are created by raising existing features to an exponent. It can help the model fits data better. For example, a new column can be created by increasing the values in column "Emotion\_Joy" into exponential of power 2. The exponential offers greater flexibility to model, lowering its bias.

Jasy Liew Suet Yan  
Gender is already included as a feature. The class label is partially based on BDI so cannot be used as features.  
July 08, 2022, 5:43 PM  
Reply

## Predicted Feedback:

Examples of features that can be added into this data set are demographic data such as age, gender, occupations, and clinical data such as The Beck Depression Inventory (BDI).

features are features which are el fits data better. For example, a n\_Joy" into exponential of power

Add Cancel

Collecting more data and selecting features are two different suggestions. Why would collecting more data help improve the model performance?

The dataset can be considered roughly balanced

Yes

Text	Actual feedback	Predicted feedback	Result
<p>The next suggestion is to add polynomial features. Polynomial features are features which are created by raising existing features to an exponent. It can help the model fits data better. For example, a new column can be created by increasing the values in column "Emotion_Joy" into exponential of power</p> <p>2. The exponential offers greater flexibility to model, lowering its bias.</p>	<p>For this to work, you must first provide evidence that the current hypothesis function learnt is too simple and the model suffers from high bias (underfitting).</p>	<p><b>Again, there is no evidence showing that your model is too simple.</b></p> <p>Need to make a conclusion which is the ONE best model.</p> <p>Again no evidence of high variance. Outliers and the number of dimensions mean two different things.</p>	Accurate

## Actual Feedback:

The next suggestion is to add polynomial features. Polynomial features are features which are created by raising existing features to an exponent. It can help the model fits data better. For example, a new column can be created by increasing the values in column "Emotion\_Joy" into exponential of power

2. The exponential offers greater flexibility to model, lowering its bias.

Jasy Liew Suet Yan  
Suggestion 2: For this to work, you must first provide evidence that the current hypothesis function learnt is too simple and the model suffers from high bias (underfitting).  
July 08, 2022, 5:44 PM

## Predicted Feedback:

The next suggestion is to add polynomial features. Polynomial features are features which are created by raising existing features to an exponent. It can help the model fits data better. For example, a new column can be created by increasing the values in column "Emotion\_Joy" into exponential of power

2. The exponential offers greater flexibility to model, lowering its bias.

Add Cancel

Again, there is no evidence showing that your model is too simple.

Need to make a conclusion which is the ONE best model.

Again no evidence of high variance. Outliers and the number of dimensions mean two different things.

## CHAPTER 6: CONCLUSION & FUTURE WORK

In conclusion, the development of this web application for semi-automatic grading represents a significant step towards addressing the challenges instructors face in providing timely and consistent feedback to students on written assignments. The integration of lexical analysis allows for a comprehensive evaluation of language and grammatical quality of the assignments. By leveraging text extraction and natural language processing techniques, particularly employing TF-IDF for feedback generation, the system enables instructors to efficiently manage large volumes of assignments while still delivering personalized feedback to students. This not only enhances the learning experience by facilitating continuous improvement but also contributes to fair and consistent grading practices.

In the future, there are promising paths to explore to improve the functionality and user experience of the system. Firstly, the feedback generated by the system can be archived and utilized for subsequent batches of students, fostering continuity, and ensuring that valuable insights are not lost over time. Additionally, implementing robust backup mechanisms can safeguard against data loss, providing reassurance to instructors and administrators in the event of accidental deletion or system failure.

Furthermore, future iterations of the web application could explore additional features such as integration with learning management systems, real-time collaboration tools, and advanced analytics for performance tracking. By continuously refining and expanding the capabilities of the system, we can further optimize the teaching and learning process, ultimately contributing to improved academic outcomes for both instructors and students.

## REFERENCES

- [1] D. L. B. MUHAMMAD, "Malaysia Educational Statistic - Quick Facts 2023," Educational Macro Data Planning Sector, Putrajaya, 2023.
- [2] M. o. H. E. M. Policy Planning and Research Division, "Higher Education Report: [MALAYSIA]," UNESCO National Commission, Paris, 2022.
- [3] D. Nicol, "From monologue to dialogue: improving written feedback processes in mass higher education," *Assessment & Evaluation in Higher Education*, vol. 35, p. 17, 2010.
- [4] E. Boudreau, "How teachers can help students receive and learn from comments and critiques," 21 January 2022. [Online]. Available: <https://www.gse.harvard.edu/ideas/usable-knowledge/22/01/importance-feedback>.
- [5] "FeedbackFruits | Digital Teaching Tools for Higher Ed," [Online]. Available: <https://feedbackfruits.com/>.
- [6] "Gradescope | Save time grading," [Online]. Available: <https://www.gradescope.com/>.
- [7] "Formative | Questions & Hints Powered by ChatGPT," ChatGPT, [Online]. Available: <https://www.formative.com/ai-powered>.
- [8] "codePost: Autograder and code review for computer science courses," codepost.io, [Online]. Available: codePost: Autograder and code review for computer science courses.
- [9] S. G. Sumit Sharma, "A Correction Model for Real-word Errors," *Procedia Computer Science*, vol. 70, pp. 99-106.

- [10] S. Li, "Natural Language Processing for Fuzzy String Matching with Python," Medium, 6 12 2018. [Online]. Available: <https://towardsdatascience.com/natural-language-processing-for-fuzzy-string-matching-with-python-6632b7824c49>.
- [11] R. N. K. I. Kazuaki Hanawa, "Analyzing Methods for Generating Feedback Comments for Language Learner," *Journal of Natural and Processing*, vol. 29, no. 3, 2022.
- [12] J. Moris, "language-tool-python: Checks grammar using LanguageTool.," PyPI, 18 4 2022. [Online]. Available: <https://pypi.org/project/language-tool-python/>.
- [13] L. Ramadhan, "TF-IDF Simplified," Medium, 4 2 2021. [Online]. Available: <https://towardsdatascience.com/tf-idf-simplified-aba19d5f5530>.