AiM: Final Release Report

ENG 4000 Capstone Project

By Team I:

Jihal Patel
Joshua Amilhasan
Lukas Rose
Mike Shen
Matthew Tran
Samyak Jain
Sharujan Rajakumar
Varuhn Ruthirakuhan

Team Supervisor:

Dr. James Andrew Smith

Lassonde School of Engineering
York University

April 18, 2022

1 Executive Summary

Many schools require a strong web application that can help course directors deliver relevant content towards their students. It was seen during this covid pandemic that the need for these web applications has heavily increased. Not only do schools now require a strong web application, but they also require one with AI implementations that can help develop and improve their courses content.

The main objective our team looked to fulfill during the course of this project was designing a web application with the aim of improving existing online educational platforms. With this large objective in hand, it was important to divide the tasks down into smaller subgoals. For example, one critical feature the team looked into adding was allowing each user of the application to login to their own respective accounts where they will be granted certain privileges based on their specific role. The primary role with most privileges would be the course director. This is because the course director is responsible for uploading each lesson with additional resources that can help their student develop more knowledge

Apart from roles, another primary goal was designing courses with associated homework sections. This was done so the student and course director are both able to track the students progress. By implementing an AI framework, the application will be able to split each section of homework into separate parts so the professor can receive reports on students' individual performances.

This project can also be further evolved with the use of its AI implementation. As artificial intelligence is continually growing, our system will also be able to continuously expand and adapt. As other AI implementations get introduced and developed, their important uses will be added to this system. Similar to how the team was able to incorporate a speech-to-text feature with the help of the Google ML API, additional AI’s can be regularly introduced for further development of the system. The inclusion of AI has helped our team to develop many features for the board such as quiz QR scanner, image-to-text, speech-to-text, password encryption, and much more.

The incorporation of AI has also improved our systems' novelty and usefulness. The team always looks to fulfill where other systems lack. Where other web applications used by schools do not use any AI framework, our application is looking to include more AI frameworks that show relevance to our project's objectives.

2 Introduction Of High Level Design Flow Chart

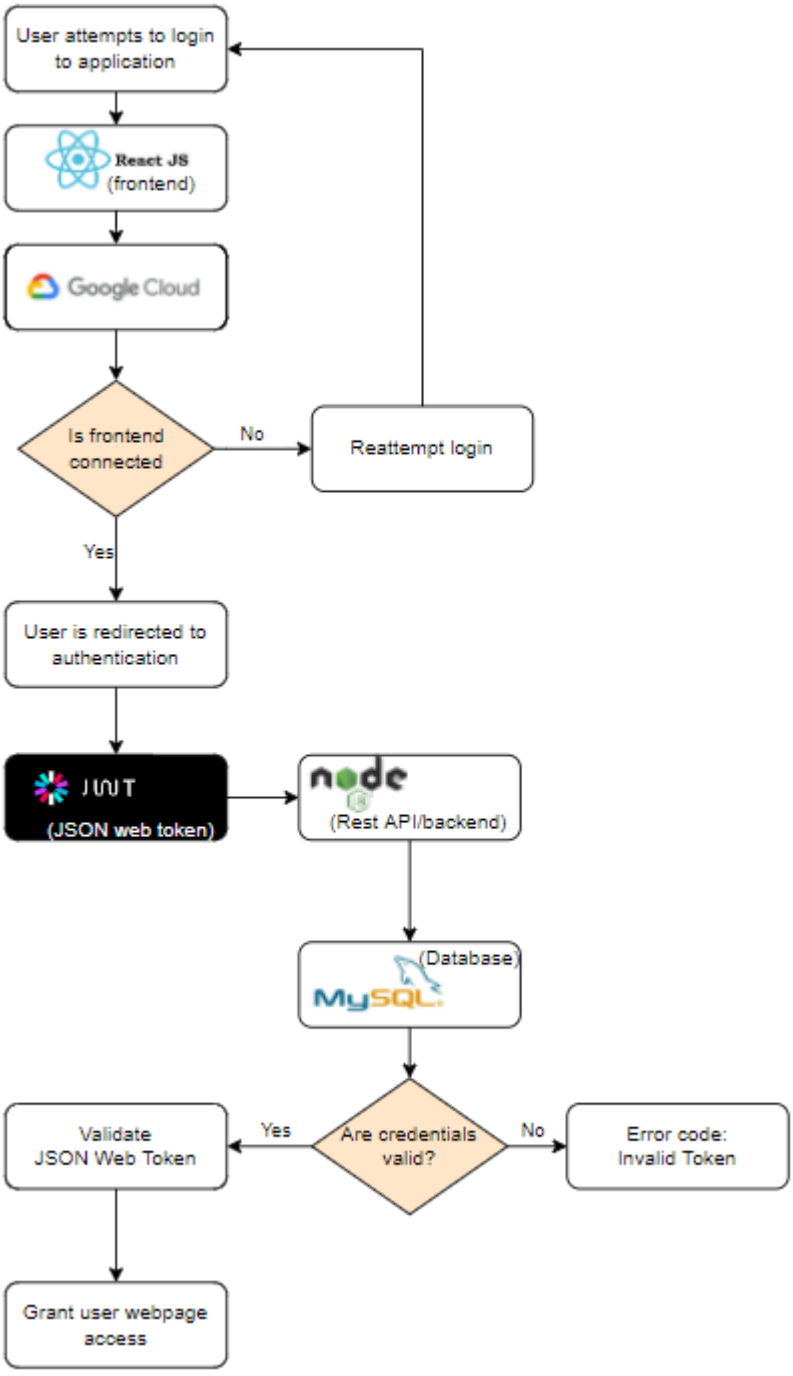


Figure 1: High-level design flow chart of the web application.

3 Technical Volume

3.1 Requirement Review

3.1.1 Product Vision Board

3.1.1.1 Purpose

Our software product looks to serve the purpose of improving the current educational platforms on the market that incorporate Artificial intelligence learning. The current education system being for online learning today contains significant room for improvement. This is caused by students not feeling that they are receiving the best possible education that their institution is providing. Students will then decide to understand their learning material using external sources.

As a team, we had discussed and found that every member felt that at some point in their education, they were not adequately learning the material to perform well on required assessments. Even when discussing with group members friends who attend different universities, this seemed to also be the case. To elaborate, this lack of learning and understanding of school material is shown to be consistent across multiple educational institutions.

This issue has caused our team to derive the goal of finding a solution to this problem by improving educational platforms with how students are receiving their respective content. We want to design the courses dynamically using AI to understand the different traits of each student and cater to their specific needs.

3.1.1.2 Interests

Since our team only consists of undergraduate Software and Computer engineering students, we were all able to find a common interest in software products and web development. The group was interested in using all aspects of web development to further develop our project so that it could reach its highest potential. This includes the front-end, back-end, database construction, and much more. Moreover, the technologies/programming languages the team had learnt and used to develop this project are some of the most used out in the field consisting of HTML, CSS, JS, React JS, and Node JS. Due to this, we feel that the project not only gives us an opportunity to improve the existing educational platforms but will also provide us with skills that will be helpful in landing a job out of university.

3.1.1.3 Target Group

When designing this project, it was important to consider the different users that would use our application along with their specific needs. The target audience that was primarily analyzed are professors and students of Lassonde and York University. Professors were considered as primary users for the application since they are the users who will be required to organize and set up their course content in the best way that they see fit. The students were considered as end-users of the product as they will be the ones experiencing the product and the benefits they offer over existing educational platforms.

As mentioned previously in section 3.1.1, the issues sought out to be solved were not only prevalent in York University but common across multiple educational institutions. This allows us to broaden the spectrum of users to a target group consisting of all educational institutions currently using web applications.

Another target group our group looked into were developers of educational software such as eClass, D2L, and Google Classroom. These products can benefit from our platform by potentially including the methods we use in order to improve the way that the students receive their education. Since our product incorporates AI implementation, we are able to demonstrate many features that course directors and students alike may find helpful towards their learning.

For example, some of AI's include useful features such as image-to-text, barcode and automatic checking, and speech-to-text. This was done with the help of the Google Natural Language API and Quiz API. Moreover, online learning platforms such as Khan Academy can also count as a target group as they serve to provide education but are not adequate in assessing the understanding of the people using their services.

3.1.1.4 Product Description, Feasibility, Uniqueness

Our product is a web application that sought out to improve the existing online educational platforms. For our application, each user of the product will have the ability to login into their own respective account and each account will be set to allow for unique permissions. The primary user (the educator, i.e. professor) is given the feature of setting up a course page that provides the necessary contents required for each course.

Each lesson that a professor uploads will have the option for additional resources that the professor can add if the student did not grasp all of the content from the professor's lecture. Each course will be associated with a homework section that the student can complete after watching the lesson. The goal in mind was to split each section of homework into separate parts for the professor so they could receive reports on the students' performances.

The existing platforms such as eClass allow professors to upload content and post-assessments but there is no way for the professors to receive reports on the performance of their class or to provide dynamic courses to students based on their current skill set using AI learning. Another aspect is the outsourcing of homework to different websites where the students are forced to buy a subscription. This application was worked to be able to integrate that into the student's primary educational platform.

Moreover, to improve over other external websites we designed a way for each educator to provide the intended time each question should take. The reports will provide feedback to the professors on whether or not the students are able to complete the work in the intended time so the professors can adjust the workload in a way that the students are not given an exhaustive amount of work. Using AI algorithms, students will know exactly what to learn and which course to take to learn a particular skill. This will help students to further reduce their workload.

The uniqueness of our product comes from 2 factors.

First, the ability of the professors to see the performance of the students as not just one number (their grade) but rather as reports that indicate the performance for a variety of predefined sections (given by the educator). Primarily, each report will consist of the average time that students take on each question and the areas that they need improvements on. This will give the educator a more in-depth view of where the set of students needs improvement and potentially provide additional content on those specific sections in order to help them improve.

Second, using Artificial intelligence algorithms to help students choose the right courses to learn a particular skill. Recommendations will be made by AI for the courses that are needed by the student to learn the desired skill based on their current skill set, previous searches and interests.

Additionally, our application also incorporates the use of simple AI integration that makes the learning process significantly easier for students and course directors alike. This could be from the speech-to-text feature supplied by the Google ML API or even the image-to-text feature supplied by our quiz API which makes it easier for professors to upload course content to their webpages.

3.1.1.5 Problem Solved By Product

The current educational platforms only provide a way for professors and teachers to post their course content and assessments. Course directors are not provided with instantaneous feedback on the performances of individual students and they also cannot provide additional courses to students so they can cover any missing skills to help them move forward with their

learning. For example, in a calculus course, the professors only have the options to post their slides (if any) and assessments (if online). The homework for that course has to be done on an external website for which students must buy a subscription or forfeit 10%+ of their grade. Professors assign enormous amounts of work to students on those websites where the questions are very repetitive and there often is a pattern for the answers where the students do not have to do the question to identify the correct answer. This results in students not being able to do the excessive work and receive the feedback they need on given assignments.

Our intention is to solve this issue and provide professors with a way to seamlessly integrate the homework into the educational platform and receive reports regarding each individual student's performance along with the class as a whole. We also looked to give students the ability to search through courses and choose them based on their requirements. Our artificial intelligence algorithms will provide the recommended courses to the students based on their current skill set, previous searches and interests so that it could help the students to learn a particular skill from end to end. The feedback given by the students in the comment section and in other forums will help our AI algorithm to improve and provide better course/lesson recommendations to the students that will help students to learn in a better way.

Some features our implemented AI algorithms can perform are fairly simple while others are more complex. For example, the addition of the Google ML API allows for the application to perform the speech-to-text feature. Whether users would like to use their microphone out of preference or possibly because of an issue regarding their keyboard, this implementation looked to solve that. An additional integrated AI includes the Google Natural Language AI. This is primarily used for text analysis because it is able to use machine learning that can extract, analyze, and store text which can be used to give feedback regarding each individual student. Our application also uses the Quiz API which allows for the use of image-to-text. This provides course directors with an easy way to scan and transfer quizzes directly to text form. Our quiz API also includes machine learning that provides barcode scanning and automatic analysis of long answers. Additionally, course directors will also receive feedback reports based on how individual students perform on these specific assessments.

3.1.1.6 Societal Benefits

When professors outsource work to external platforms, they sometimes provide an unreasonable amount of work for the students. Due to this, the students can be overworked or attempt to bypass the work by attempting to instantly find the answers. Our goal is to improve this by ensuring the educators provide a reasonable amount of work and monitor the amount of time it takes the students to complete it and adjust their plans accordingly.

Moreover, the students can receive a better educational experience after knowing what exactly to learn based on their personal skills. Also since educators will receive reports based on

which will be able to identify the areas where each individual student is lacking. Based on this, the educator can provide additional resources to students based on the sections they are not comfortable with. Overall, the product assures the educator that the student is doing the required work and benefits the students as they will be receiving an enhanced learning experience without being overworked.

3.2 As-built Design

3.2.1 Class Diagram

We see the following information from the class diagram:

- Each class represents a feature with its associated variables/methods.

1) If and only if we sign up, then we can login.
2) Once a teacher logs-in, they can search for a course with AI course recommendations.
3) After you search for a course, you will be redirected to the lesson page, where you can provide feedback in the student dashboard.
4) You can access the quiz api, through barcode and text can generated from the image you provide.

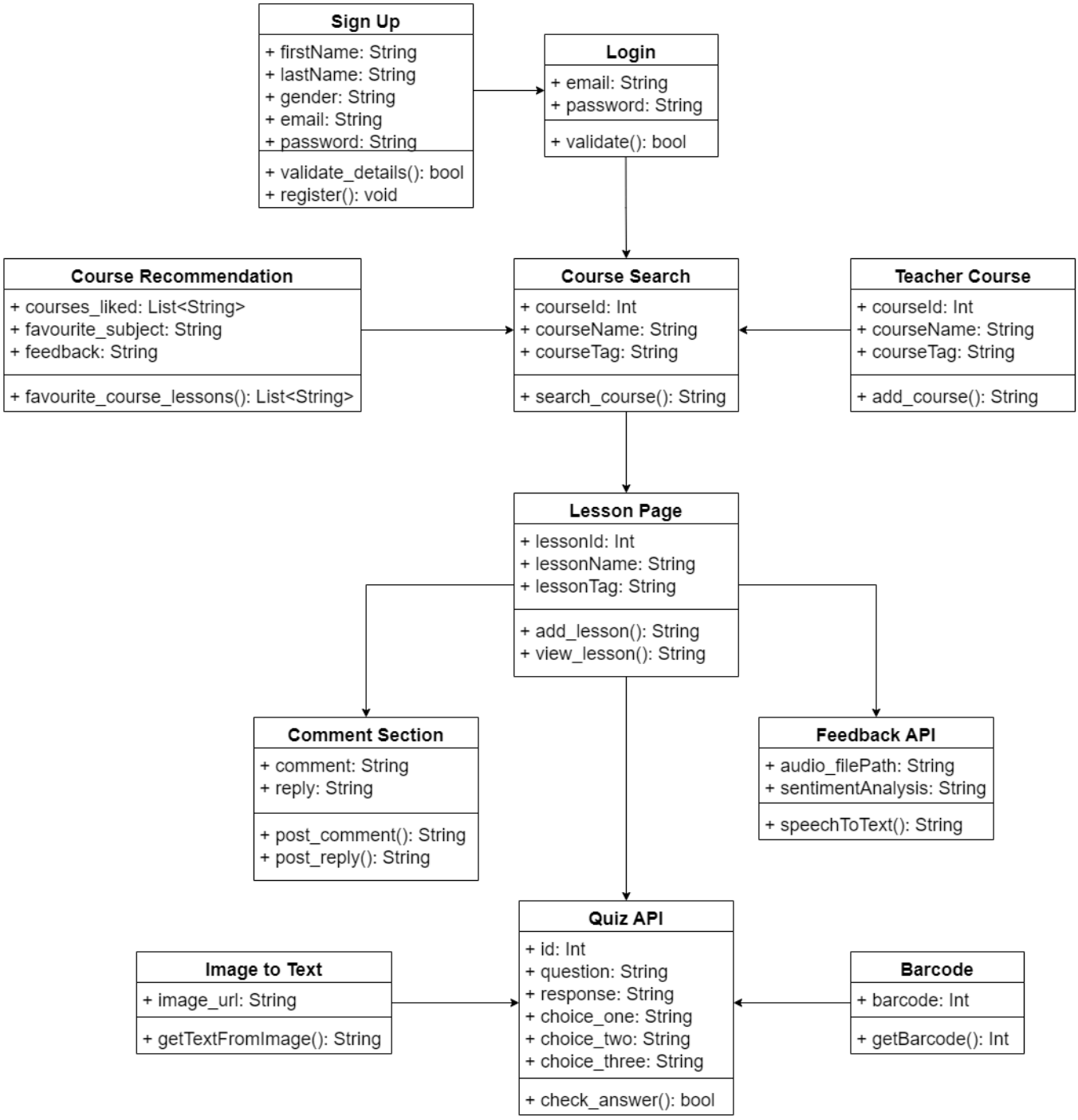


Figure 2: Class UML diagram.

3.2.2 Block Diagram

The block diagram depicts how the frontend, backend, database and AI technology are all interconnected with each other.
Based on the diagram, it is evident that backend development is the most crucial part of our development since it deals with the web server, database connectivity, and interacts with the frontend, and incorporates AI technology.

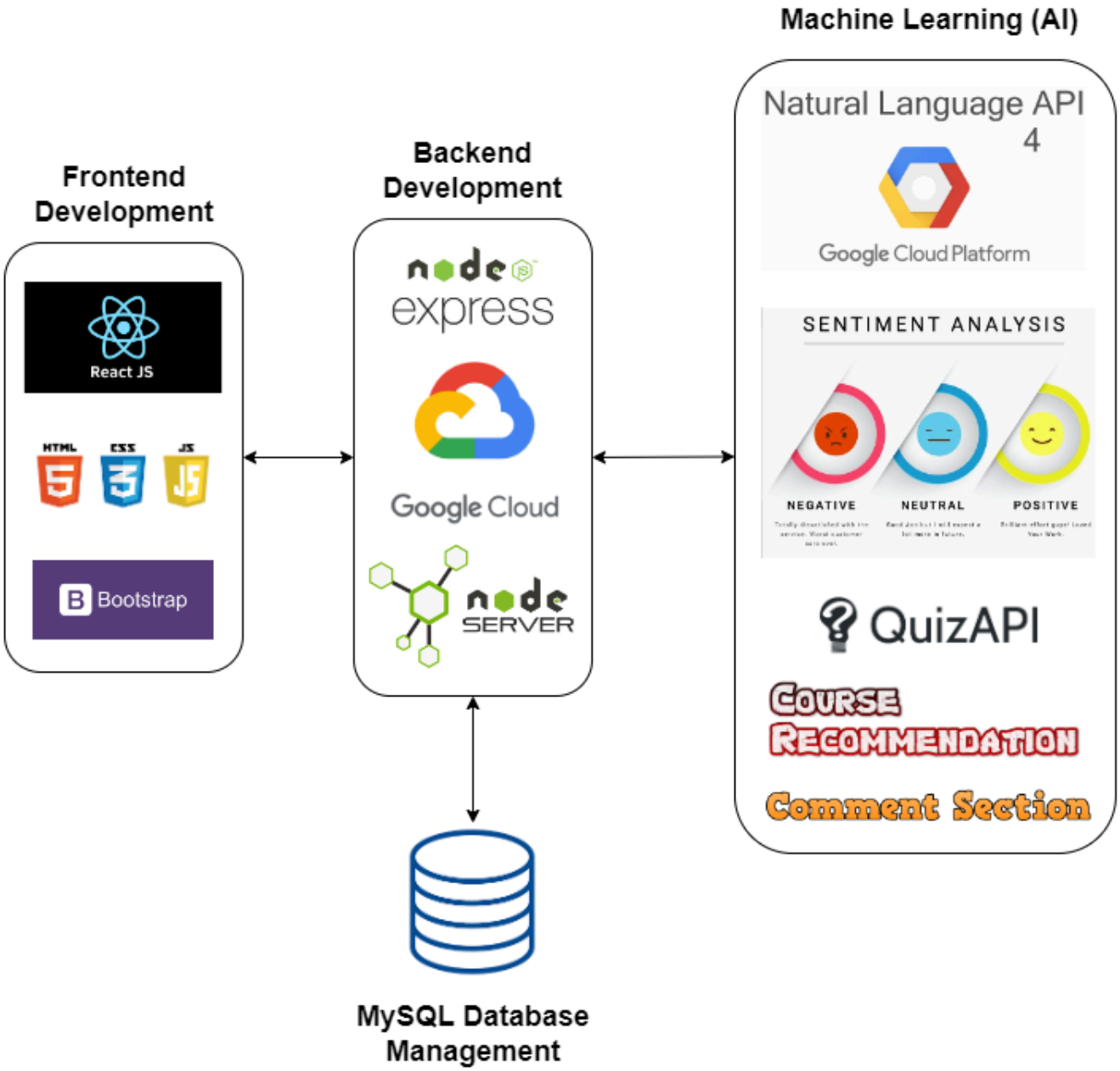


Figure 3: Block diagram of interconnected technologies.

3.2.3 Activity Diagrams

3.2.3.1 Login and Sign Up page

- The login and sign up page works by having the user enter their email and password, and once validated they are redirected to a home page. Otherwise, if their credentials are invalid, they are redirected to enter their email and password again.

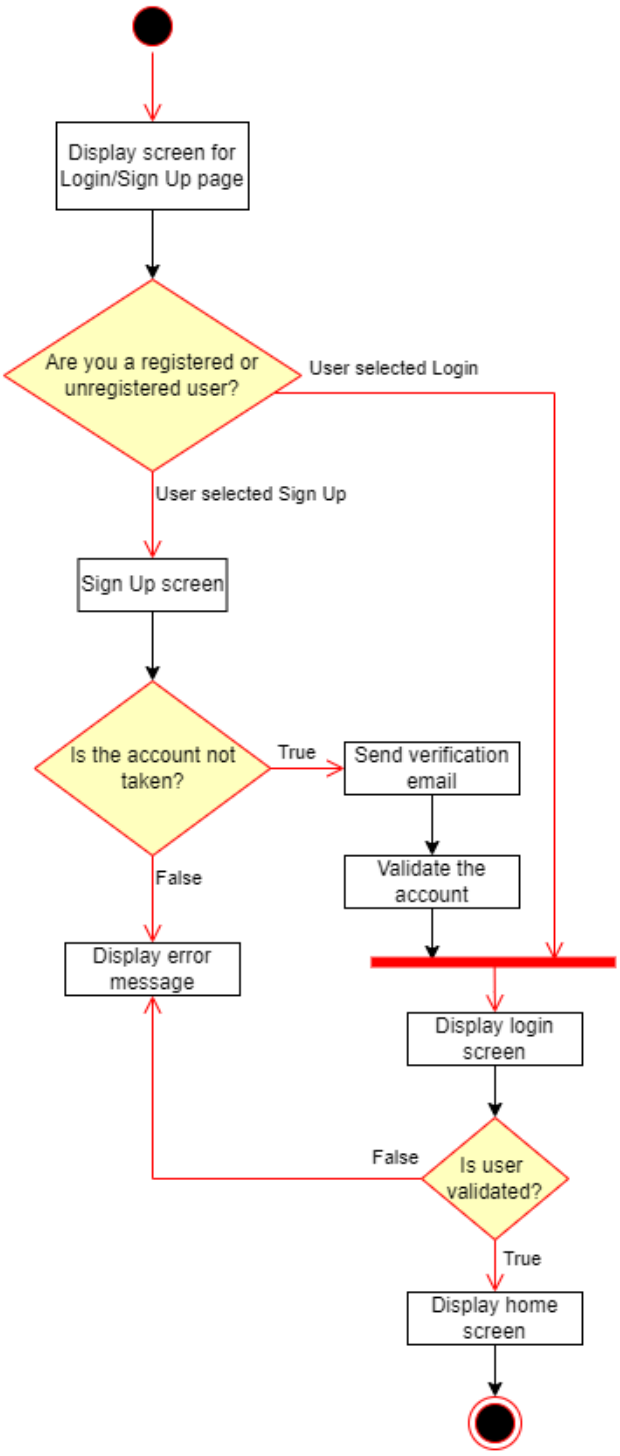


Figure 4: Activity diagram of the sign-up and login process.

3.2.3.2 Barcode, Image to Text, and Quiz API

- This activity diagram works by asking the user if they want to access the quiz on their mobile phone or desktop.

1) User selects mobile phone
    - If so, they would scan their barcode and if successful, they will be redirected to the quiz.
    - If unsuccessful, they will be asked to scan their barcode again.
2) User selects desktop
    - They will be redirected to the quiz.

- The user will then handwrite or type their answers and take a picture. Once the picture is sent, text will be generated from the image.
- Finally, AI will compare the answers to the correct answers using Google's natural learning algorithm and record their results to the quiz.

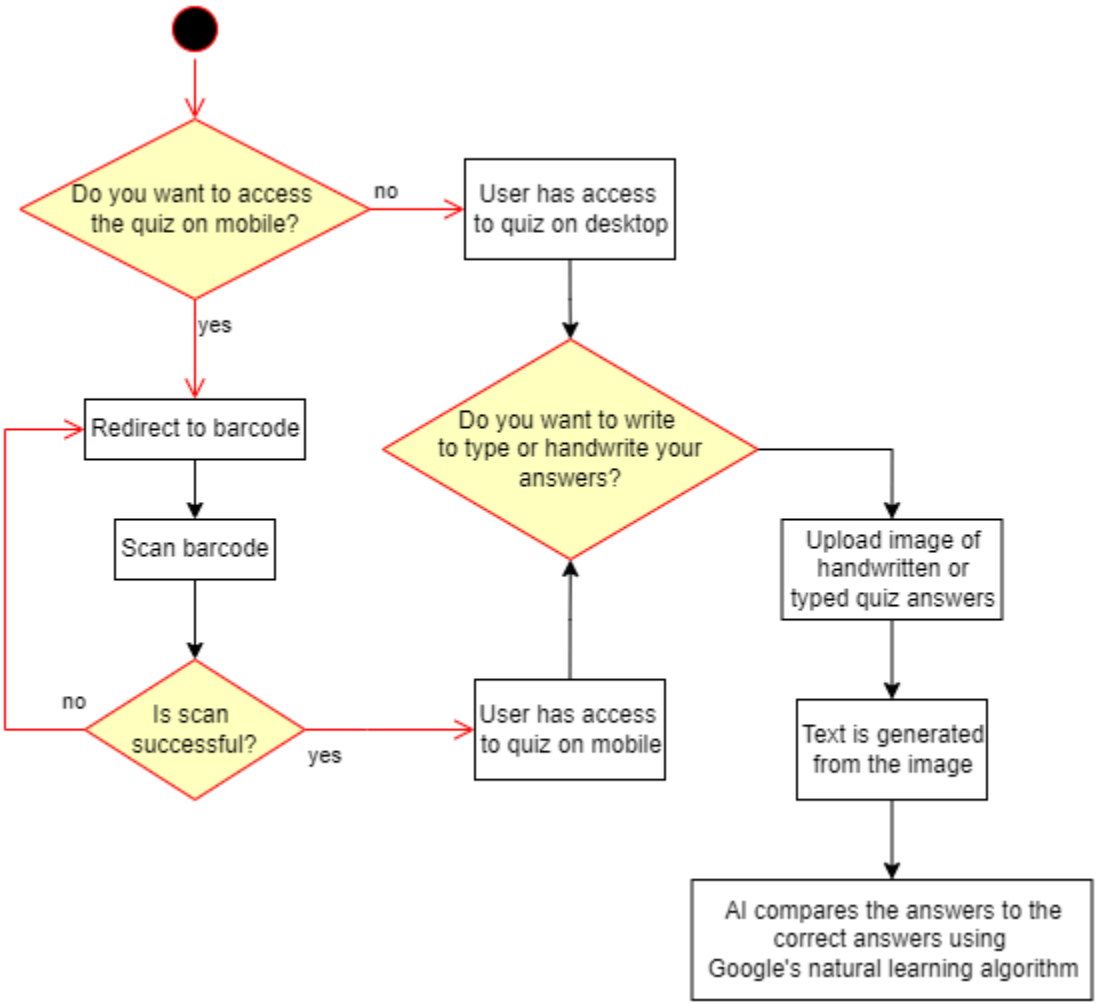


Figure 5: Activity diagram of the barcode, image-to-text, and quiz features.

3.2.3.3 Teacher course addition and course lesson page (teacher dashboard)

- The teacher dashboard works by having the teacher login to the web application with an email and password. Once their credentials are validated, the teacher can search for a course and add lessons to it.

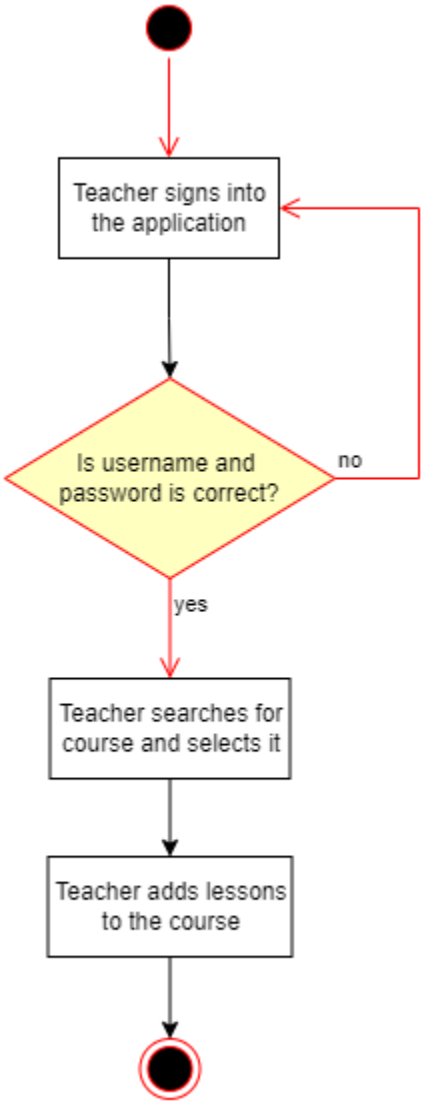


Figure 6: Activity diagram of the course and lesson creation and page from the teacher side.

3.2.3.4 Student course recommendation, comment section and course lesson page (student dashboard)

- The student dashboard works by having a student search for a course, and they will be prompted to either search through: (1) AI course recommendations or (2) normal search.
- If an AI course recommendation is selected, then they will be redirected to the lessons page and AI will filter the lessons for them based on their recommendations. To improve the AI recommendation process, students will then provide feedback by entering their comments in the comment section.
- If normal search is selected, then the students can just view their lessons and add comments in the comment section in the form of feedback.

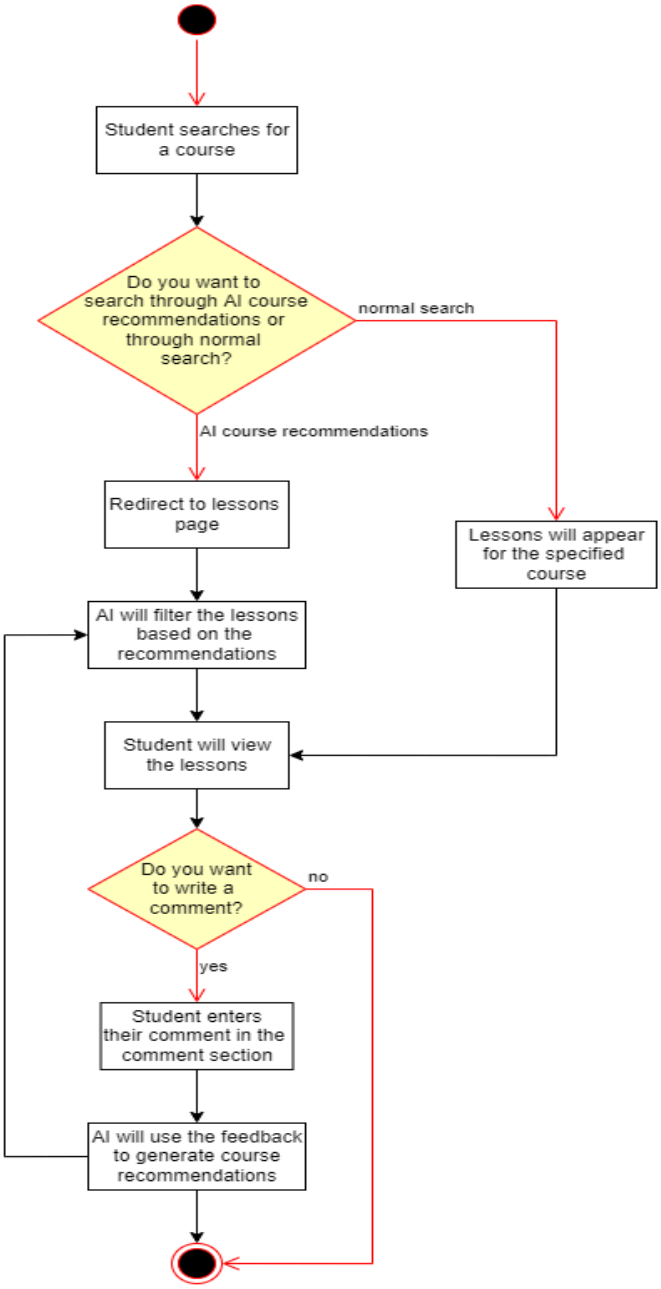


Figure 7: Activity diagram of the course recommendation, comment section, and course lesson page from the student side.

3.2.4 Database Diagram

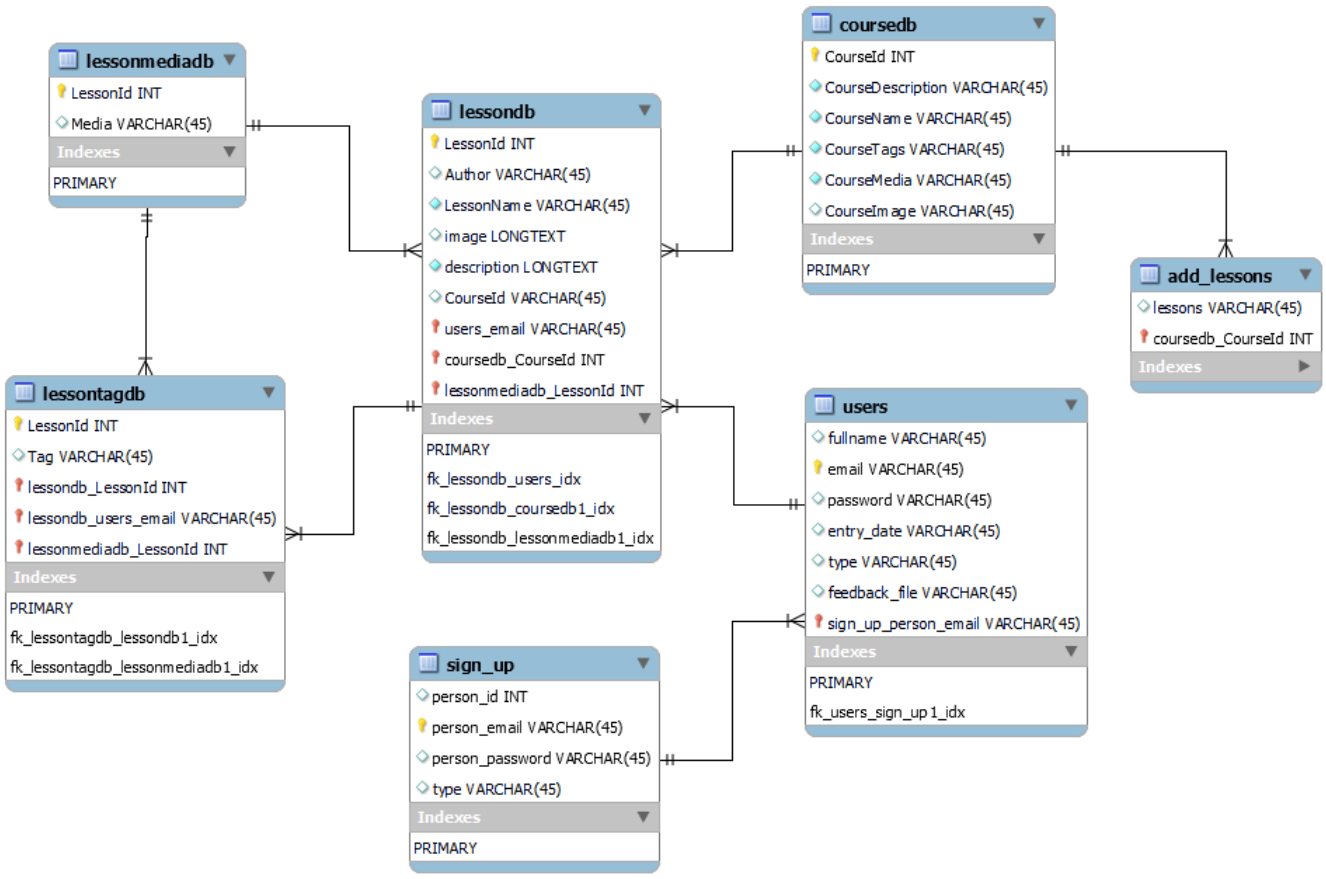


Figure 8: Database diagram.

The database diagram illustrates the following relationships between tables:

- lessonmediadb has a one-to-many relationship with lessontagdb and lessondb
- lessondb has a one-to-many relationship with lessontagdb
- sign_up has a one-to-many relationship with users
- users has a one-to-many relationship with lessondb
- coursedb has a one-to-many relationship with lessondb and add_lessons

3.2.6 Course Recommendation Algorithm

The course recommendation algorithm takes previous user searches for their desired courses and uses those to recommend the user new courses that might fit their preferences. A high-level diagram of how this learning algorithm works is shown below.

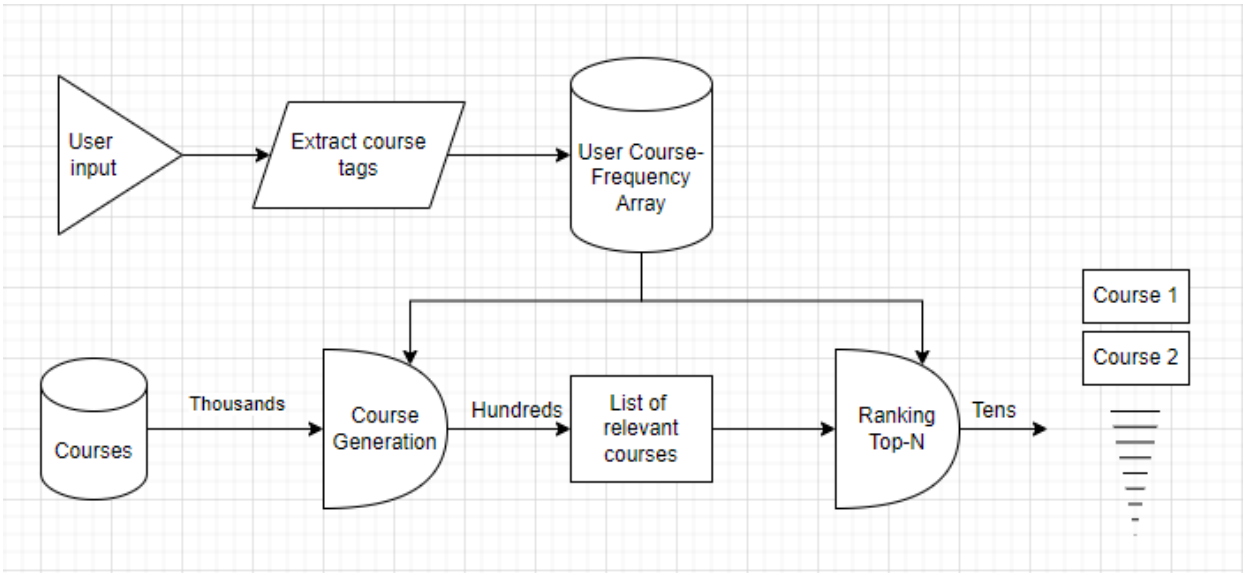


Figure 14: A high-level representation of how the frequency-based learning algorithm works.

Each course has tags that describe its contents. So a course about physics will have the “physics” tag, and more broad courses will have multiple tags. Whenever a student searches for a course or accesses a course, the system will extract the associated course tags to these actions and record them to the user’s course-frequency array. This course-frequency array records how frequent, and thus how relevant, a course tag is to a student. So if a student accesses many physics courses, the “physics” tag will have a high frequency count. The frequency count will decrease overtime or when the student searches another tag more recently and frequently. When a tag has a frequency count of 0, it will be removed from the array to save space.

The “Course Generation” acts as a filter, filtering out courses that do not contain the course tags the student is interested in. The “Ranking Top-N” further filters out the courses based on the Top-N frequently accessed course tags according to the course-frequency array of the user. N is calculated by simply having the requirement to display five courses. So if we only need the Top-1 frequent course tag to get 5 courses, then N is equal to 1. However, if the student is particularly interested in more obscure course tags, then N may be as high as (or more than) 10 to display five courses.

This learning algorithm learns and adapts to the user's preference over use. This is useful for the nature of this web application because it promotes learning of other courses and topics for the student. Since the aim of the project is to encourage a better, more advanced learning platform, then this feature is essential in achieving this goal.

3.2.7 AI generated feedback visualization

With our previously mentioned AI feedback generation, our next goal is to present our extracted data in a user-friendly format. For now, our user sentiment scores are still ranged within -1 to 1 (as demonstrated in Figure 1). Generally, every category of feedback would be presented within a bar graph to demonstrate its overall user sentiment. For our current graphs, positive evaluations skew towards the right and negative would skew towards the left.

3.2.7.1 Raw Data

Our first provided chart would be the raw sentiment data extracted from feedback. This feedback is purely the results that the Google API has extracted from all student evaluations.

All the raw sentiment score values would be taken and categorized accordingly to its sentiment range.

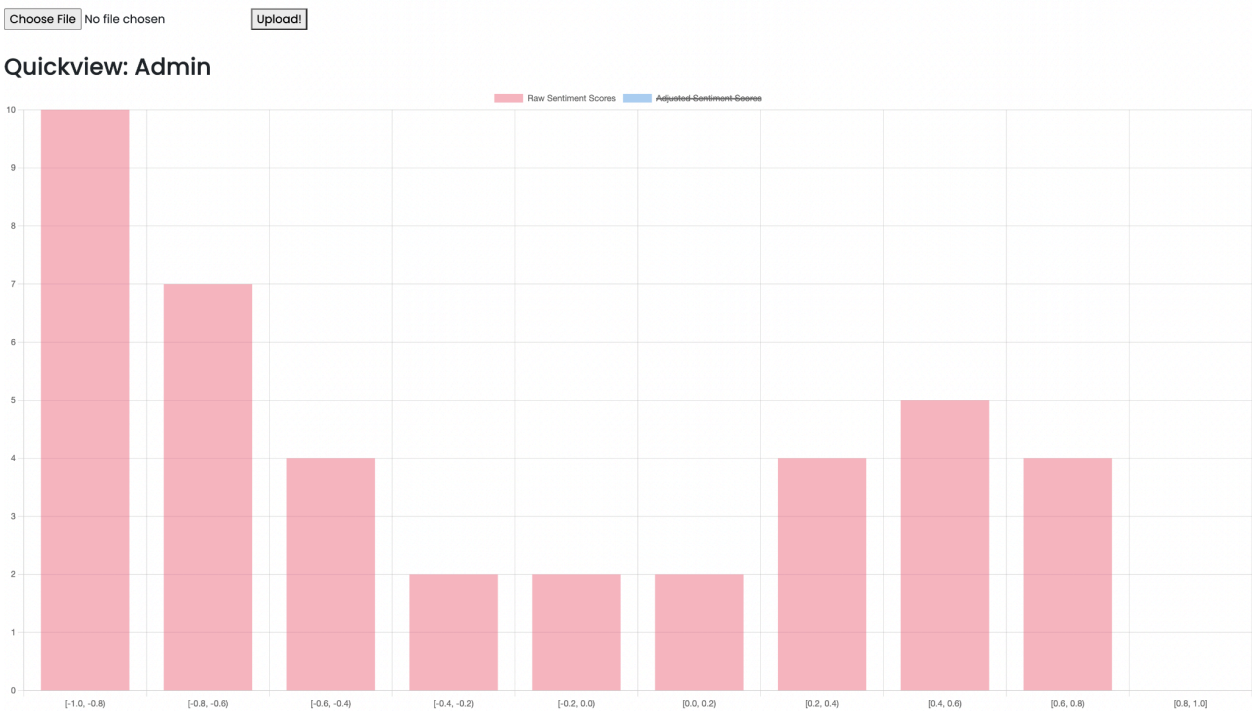


Figure 15: The raw sentiment score provided for the class

3.2.7.2 Adjusted Data:

Oftentimes, the raw generated data could be subject to bias. Either through personal means or emotional means, the data generated through Google's API can sometimes be misleading. Writing conventions such as sarcasm can oftentimes lead to misleading sentiment evaluations.

It was found that Gendered language was also a big factor for this review bias. Depending on the subject being taught, student review oftentimes seemed to have a gendered bias for Male and Female teachers.

Therefore, it would be necessary to implement an Adjusted Data feature for the user to counteract the mentioned user bias.

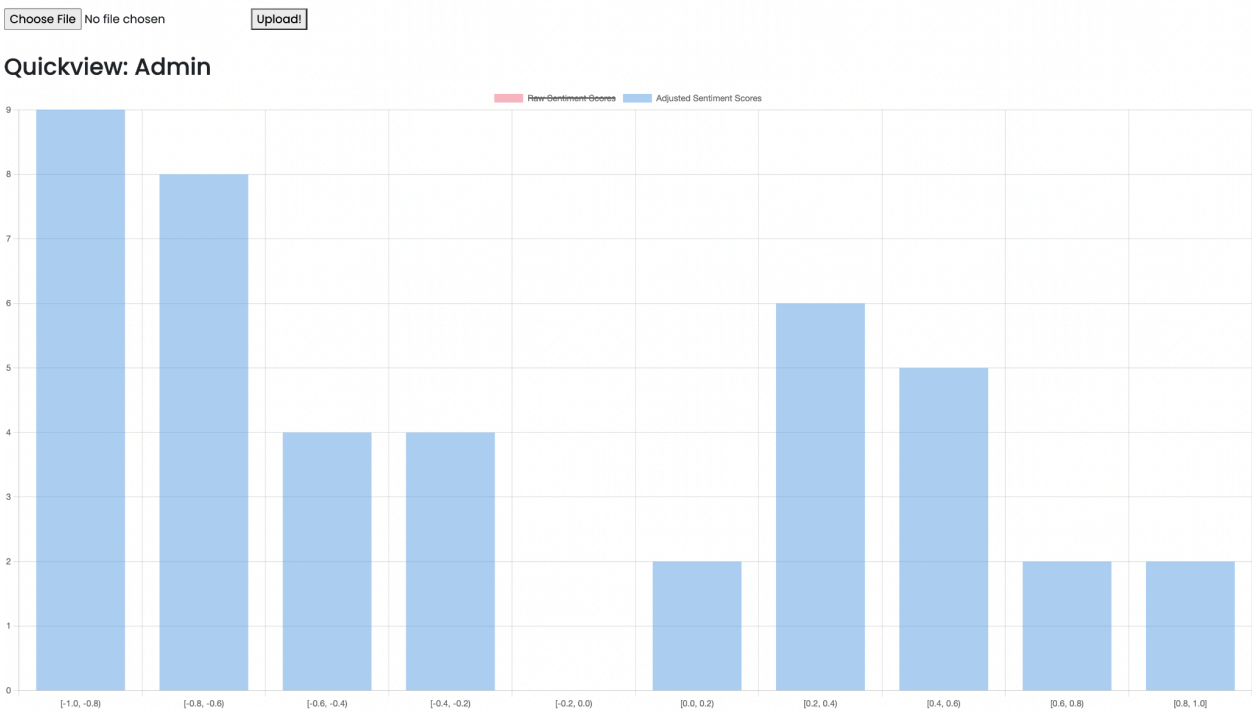


Figure 16: The adjusted sentiment score for the corresponding class

3.2.7.3 Raw & Adjusted Data:

Finally, it would be in our best interest to implement a feature that would display both Raw and Adjusted data for easy comparison. This in turn would make it easy for the user to find the most accurate results on how students feel about their courses.

Figure 17: All data is presented

3.2.7.4 Future Developments

Our goal for the future is to take our current data presentation and provide the user/teacher with enough flexibility to find the information that they want. Features such as what the feedback is directed towards. Things such as: “the teacher was not clear” or “the teacher was teaching too fast” would be prime examples of overall directed feedback that would be provided for the user rather than just a sentiment score.

Sentiment analysis System

Our goal for sentiment analysis was to utilize the GCP in two different ways as mentioned above.

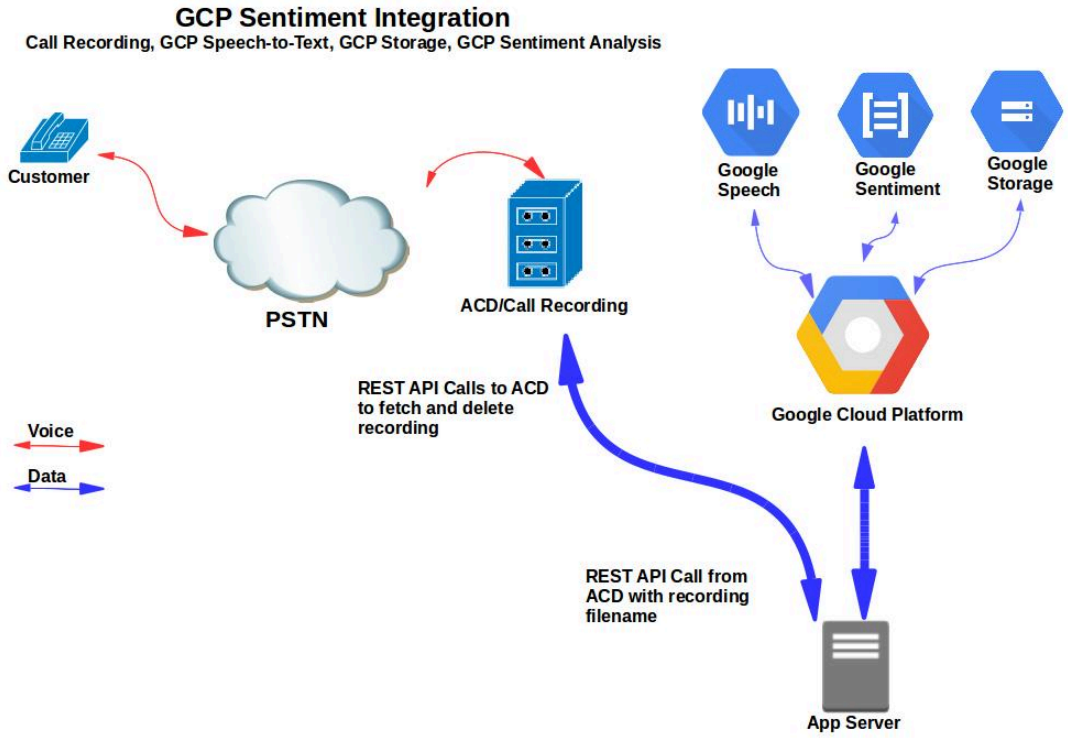


FIgure 18: GCP Sentiment Integration.

Originally, the GCP would use a PSTN phone line to extract user audio data and store it in a phone line.

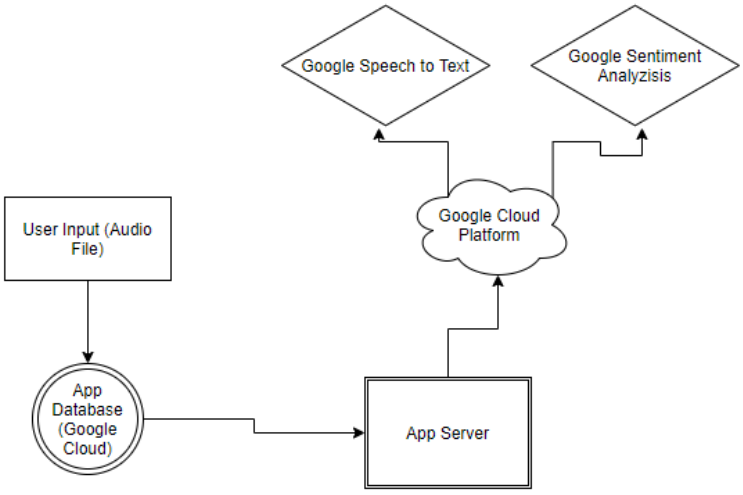


Figure 19: Data flow for sentiment analysis through web application server to Google Cloud Platform.

However, our user input is directly stored within our cloud storage and in our app server. The Text-Speech and Sentiment analysis still remain the same.

3.2.8 Quiz API, Image to Text and Barcode Algorithm

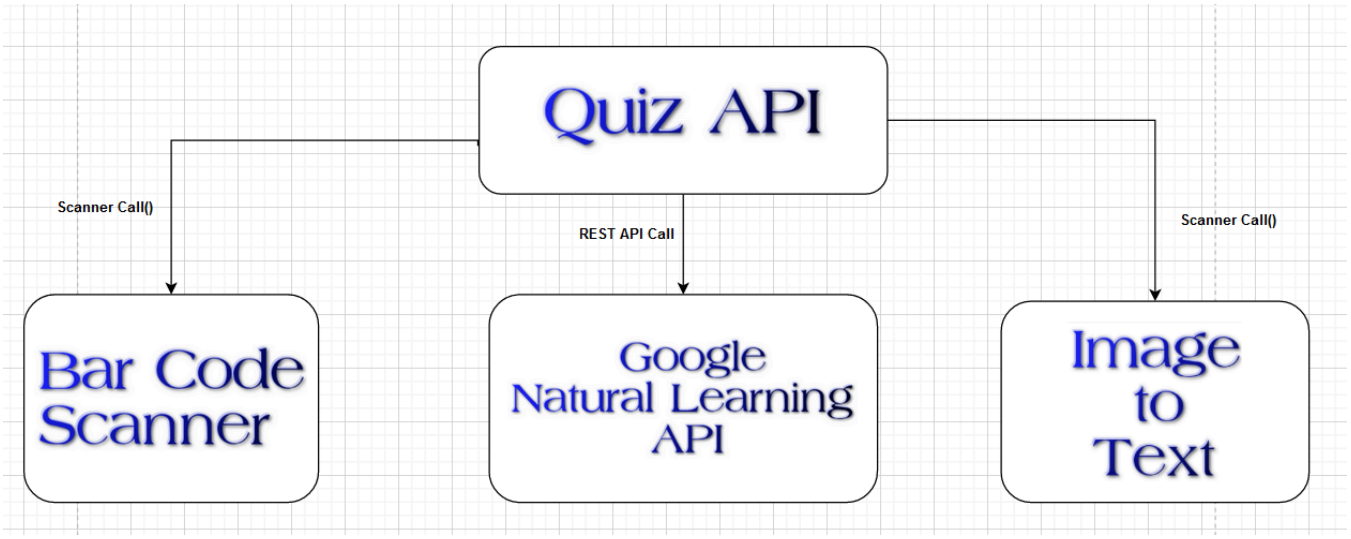


Figure 20: Data flow for Quiz API to test User,s Understanding

The quiz api is used to test the understanding of the users. The quiz feature of our application is made in such a way that it can be accessed by any device like mobile or laptop by using a barcode scanner.

Google Natural Learning API helps our quiz api to automatically mark the long answers so that teachers don't have to manually check each and every long answer.

Image to Text feature supports our quiz api by giving the flexibility to the students to upload their answers in the handwritten form or by typing digitally. The handwritten answers are later converted into digital format so that they can be automatically checked by the Google Natural Learning API.

3.2.9 Performance Report

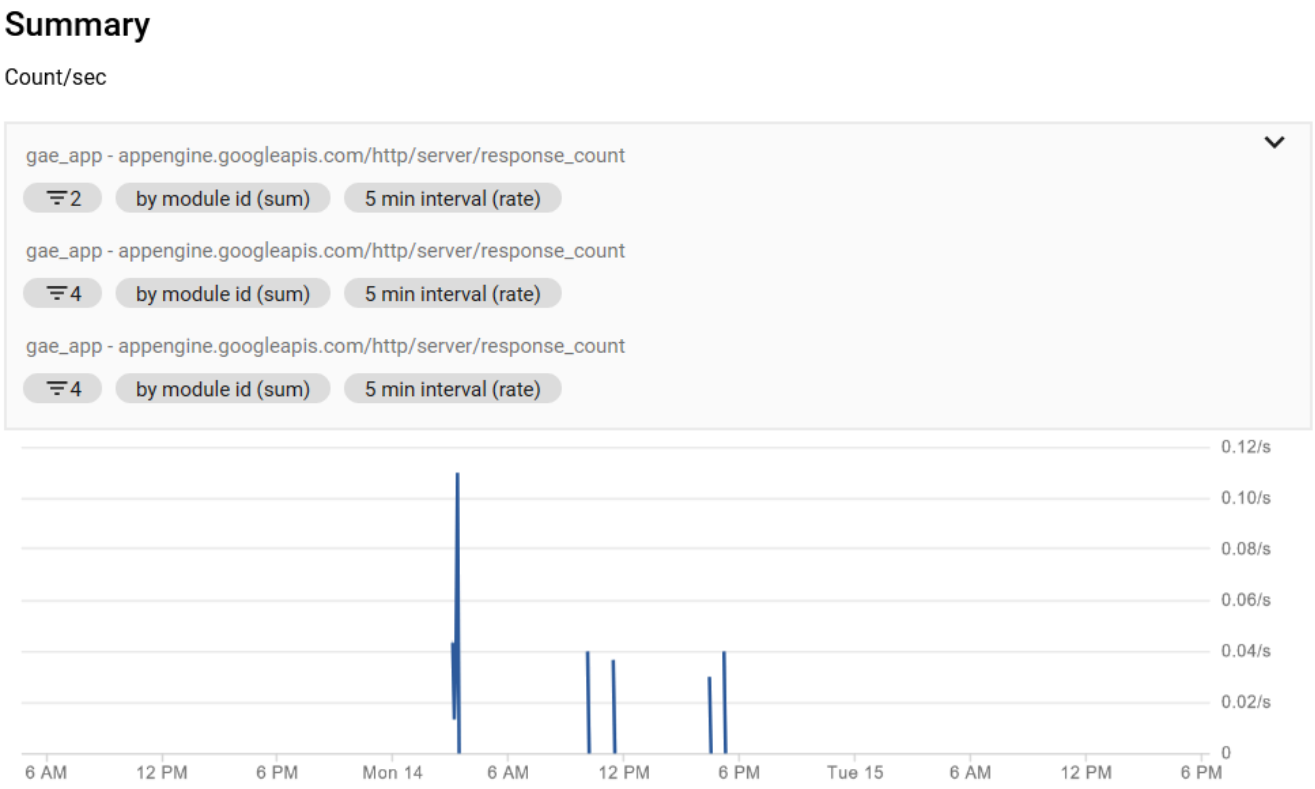


Figure 9: This graph shows the performance report summary for Monday, April 14th where the website is active from the intervals: 5am, 11am, 12pm, 5pm, and 6pm.

From this graph we can conclude that:
Average Response Time = (0.042/s + 0.11/s + 0.04/s + 0.038/s + 0.03/s + 0.04/s) / 6
= 0.05 count/s

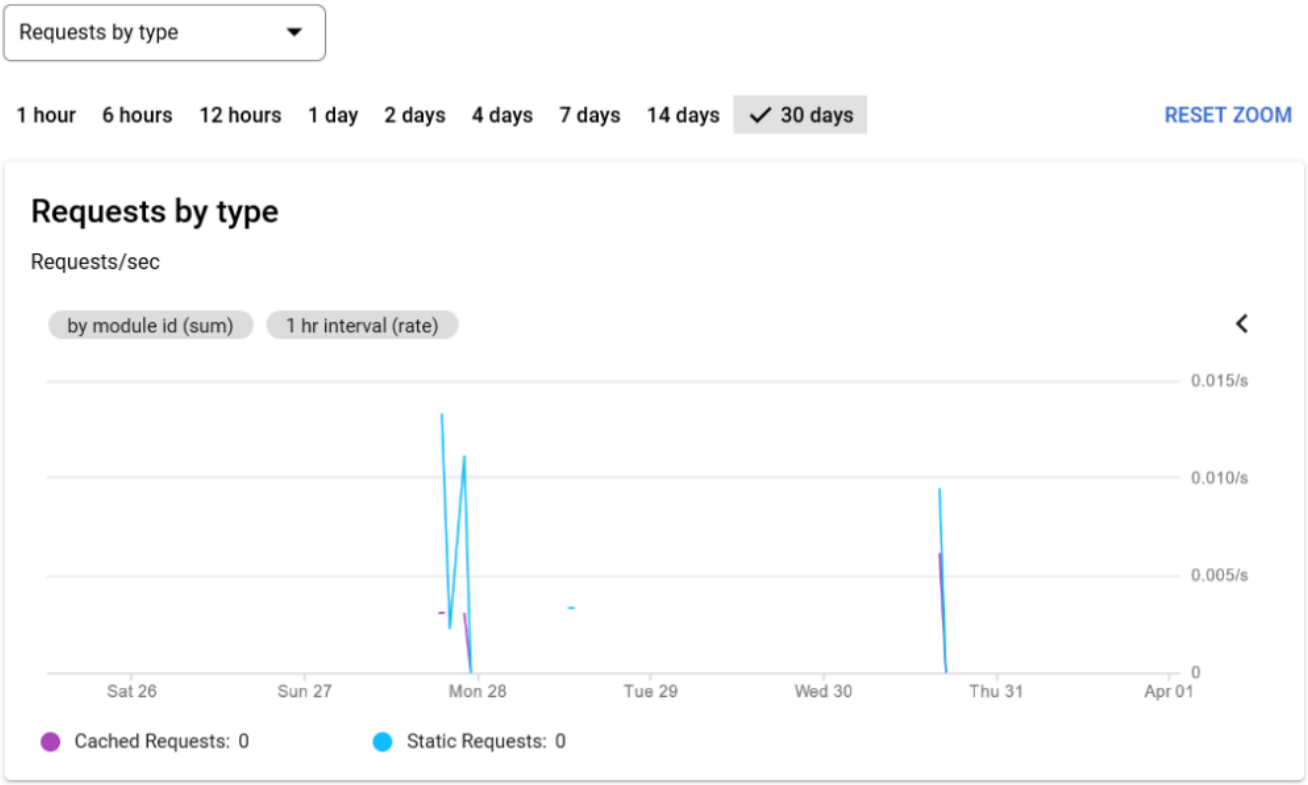


Figure 10: This graph shows the Requests by type for dates between Saturday, March 26th to April 1st.

From this graph we can see that:
Average Requests by type = (0.013/s + 0.003/s + 0.011/s + 0/s + 0.0095/s + 0/s) / 6
= 0.0061 requests/s

Figure 11: This graph shows the average latency of the web application in the last 30 days.

From this graph we can see that:
Average Latency = (3ms + 26ms + 22ms + 5ms + 3ms + 26ms) / 6
= 14.17ms

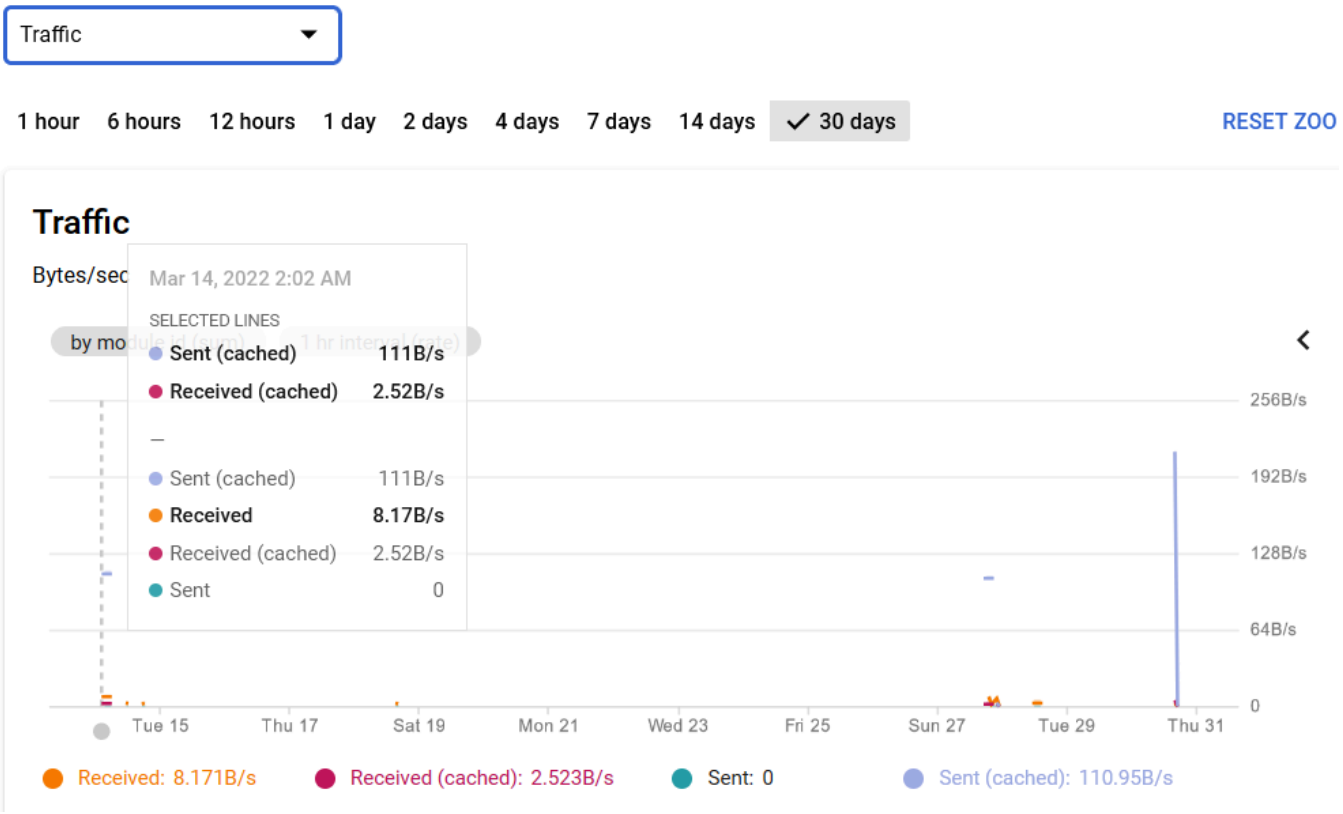


Figure 12: This graph shows the average traffic of our web application in the last 30 days.

From this graph we can see that the average traffic is 224 B/s.

Current load

URI	Requests/Minute current	Requests last 24 hours	Runtime MCycles last hour	Average latency last hour
/static/js/main.55df0d9e.js	0.8	35	0	1.25 ms
/static/css/main.3460e470.css	0	18	0	1 ms
/favicon.ico	0	14	0	1 ms
/manifest.json	0	14	0	1 ms
/signin	0	13	0	7 ms
/logo192.png	0	12	0	1 ms
/static/media /svg-1.7e7149ff42aae8450441fa26f205a056.svg	0	8	0	ms
/static/media /svg-3.68bf86f3e5bb3a34f1cf1984fd548f3f.svg	0	7	0	ms
/	0	7	0	ms
/static/media /svg-5.b8e417785e2f2402ede1cb79a71f5e6f.svg	0	7	0	ms
/static/media /svg-3.68bf86f3e5bb3a34f1cf1984fd548f3f.svg	0	7	0	ms
/	0	7	0	ms
/static/media /svg-5.b8e417785e2f2402ede1cb79a71f5e6f.svg	0	7	0	ms
/static/media /svg-6.037927c3b015ea52fcb35ac5bf0c296b.svg	0	7	0	ms
/static/media /svg-4.dfbca26e8f125c6d4a613f24666120c2.svg	0	7	0	ms
/static/media /svg-2.92dc1e6ed62eae8f9754aabc3b45eced.svg	0	7	0	ms
/static/js/main.55df0d9e.js.map	0	4	0	ms
/static/css/main.3460e470.css.map	0	3	0	ms
/studentdash	0	2	0	ms

Figure 13: This graph shows the current load of the web application which indicates information about the Request/Minute, Requests, Runtime MCycles, and Average Latency in the last hour.

For the AI portion of the project, one implementation we decided on was for feedback. Using Google ML API, we utilize speech-to-text and sentiment analysis to quickly give and analyze feedback for courses. The speech-to-text allows the student to simply vocalize their opinions on a lesson while our web app transcribes that to a script. This script is then passed to a Sentiment Analysis to create an estimate of how positively or negatively the student reacted to the lesson.

A score is calculated corresponding to the tone of sentence or paragraph, where a positive sentiment indicates a score above 0.05, a negative sentiment indicates a score below -0.05, and a neutral sentiment indicates a score in between. The Sentiment Analysis also creates a graphical representation of all comments and their overall “mood/sentiment” to give the teacher a clearer view of how their lessons are perceived.

3.3 As-built Design Compliance Analysis

3.3.1 Final NCR Disposition

During the project development process, the team was able to fully appreciate the agile methodology as it allowed us to modify and cancel requirements that did not meet the project scope. In an agile methodology, the needs of the project are constantly changing and adapting to changing user and team needs. As a result, some requirements were modified and canceled to adapt to this change. This section can be broken down into three NCR dispositions: (1) Modified Requirements, (2) Canceled Requirements, (3) General Change.

(1) Modified Requirements:

During the course of the project, the team experienced many evolving changes and needed to modify the existing requirements in order to successfully meet these changes. Some of the changes that were modified from the list of requirements (see appendix) was changing the load response time of the web application from 2-5 seconds to 5-10 seconds. As a group, we decided to make this change because as we completed various sprints, the number of features we had grew exponentially, which reduced the overall performance of the website. And, to take account of this change we decided to increase the range of reasonable load response time.

(2) Cancelled Requirements:

In addition to the modified requirements, the team found that the scope that was defined earlier on did not correctly reflect the MVP and end product that we were trying to reach.

The majority of the requirements that we cancelled belonged to non-functional requirements that were listed in the appendix.

The cancelled requirements include:

- The system should be recoverable. This means that the web application has the ability to recover from a crash or a failure in the system and return to full operations.
- The system should be portable, which means that there is less effort to move the software to a different target platform.

As a group we decided to cancel the non-functional requirement of making the system recoverable because recovery files can take up a lot of memory space and can slow down the overall performance of the web application. Also, as a group we decided to cancel the non-functional requirement of making the system portable because we thought that for the purpose of this particular project scope it would be more suitable to have the project running on a desktop platform and not on a mobile platform.

(3) General Change:

Database Design: We used MySQL database which provides the ability to scale data both vertically and horizontally. Throughout the development process, we expected the database to undergo a lot of change. However, the first schema posed multiple limitations for querying and filtering. To resolve this issue, we used the LIMIT sql command in the statement/query to help save execution time and improve query performance.

Calendar Feature: The calendar was initially developed at first glance to look for reminders and see upcoming events. However, as the project progressed, we decided not to include the idea of a calendar because it already existed and did not add value to our project. For example, since we have to push notifications for reminders, the calendar feature did not add value.

3.4 Final Verification Status (Front end Testing)

3.4.1 Using Perfecto

Using Perfect, we are able to conduct basic testing for accessing the web application. Accessing the web application and logging into the website using the cloud deployed website.

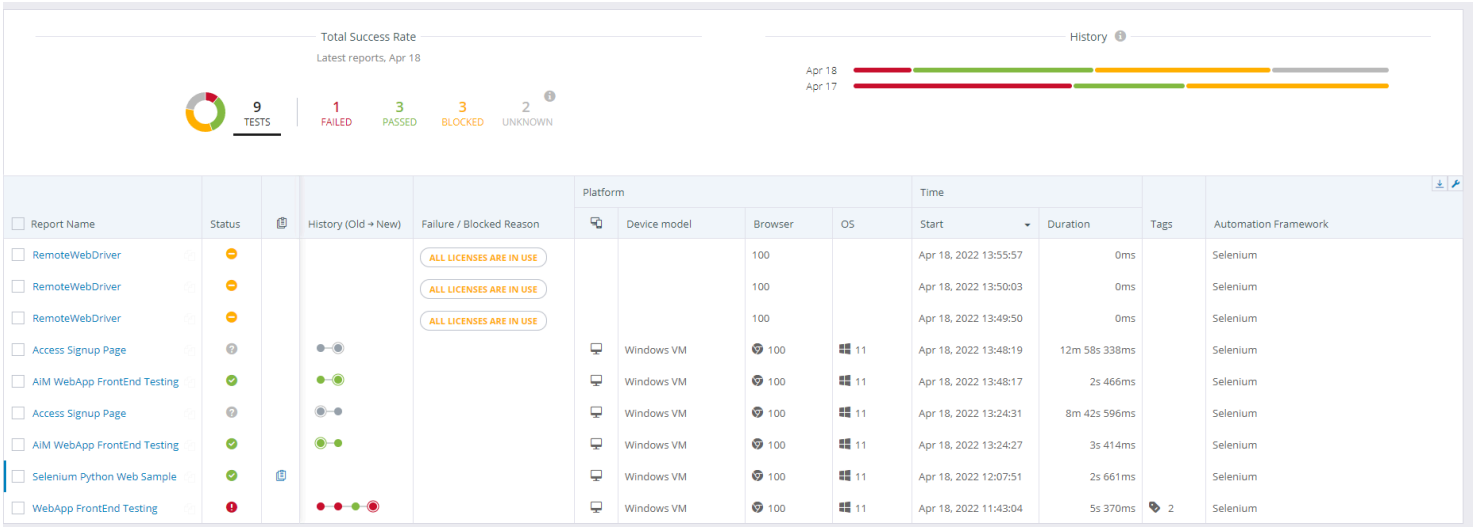


Figure 20: Test Cases Using Perfecto.

Shown in the image above is the comprehensive testing done using Perfecto's service to conduct testing on our web application. Certain Test cases were not passed originally but after refining the test cases, it was proven that the websites are accessible.

3.4.2 End to End Testing

End-to-end testing was conducted and proved that the user is able to access the signup page and create an account.

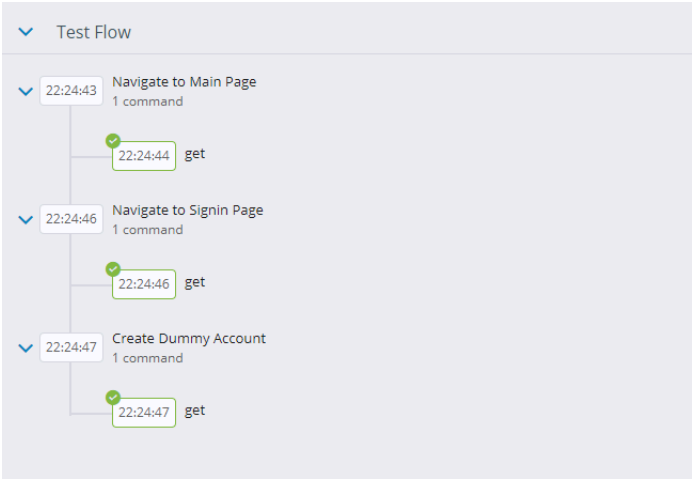


Figure 21: Test Flow.

The flow of the test case follows the end-to-end structure that the user would typically access. As seen in Figure 2, we first start from entering the website, making an account.

3.4.3 Using Unit Testing

Because of technical constraints, I could not access the full version of the perfecto platform. Therefore, anything beyond website accessing was done through selenium web scraping.

```
(base) C:\Users\mshen55\Downloads\Perfecto-Testing-Python\web>python webApptest.py
....
----------------------------------------------------------------------
Ran 4 tests in 3.073s

OK

(base) C:\Users\mshen55\Downloads\Perfecto-Testing-Python\web>
```

Figure 22: Running Front End Test Cases using Selenium.

As shown in Figure 1, we ran 4 test cases. Each test case is an end-to-end test for each stakeholder in the project. The stakeholders being: Students, Teachers, and Admin.

```
class testSignup(unittest.TestCase):
    def test_ping(self):
        # Configure Selenium to keep web open
        options = webdriver.ChromeOptions()
        options.add_experimental_option("detach", True)
        # Enable Logging
        options.add_experimental_option('excludeSwitches', ['enable-logging'])
        # options.add_experimental_option("detach", True)
        driver = webdriver.Chrome(options = options, executable_path = PATH)
        driver.get("https://clever-cyclist-320200.uc.r.appspot.com/signin")
        driver.find_element(By.XPATH, value='//*[@id="root"]/div/div/div/form/div/div[2]/div/a[2]/a').click()
        driver.find_element(By.XPATH, value = '//*[@id="root"]/div/div/div/form/div/div[2]/div/form/input[1]').send_keys("name")
        driver.find_element(By.XPATH, value = '//*[@id="root"]/div/div/div/form/div/div[2]/div/form/input[2]').send_keys("mail@mail.com")
        driver.find_element(By.XPATH, value = '//*[@id="root"]/div/div/div/form/div/div[2]/div/form/input[3]').send_keys("password")
```

Figure 23: Example test case.

As shown in figure 2, signup testing was also conducted to ensure that users had the ability to create accounts using the appropriate parameters.

3.5 Tests (Back End)

3.5.1 Logistical Testing

Originally, we had conducted vertical testing on our web application. This meant we had to split our project components to fit a cohesive linear structure. The order of testing that we would approach would be: Login->Signup->Dashboard.

The first component of testing would be the login. We would test this by creating a “switchToSignIn” function. We would test this with an expected term like a dummy login. The Signup feature would mainly test to see if the validation message for account creation would display. The dashboard would then test the user’s input.

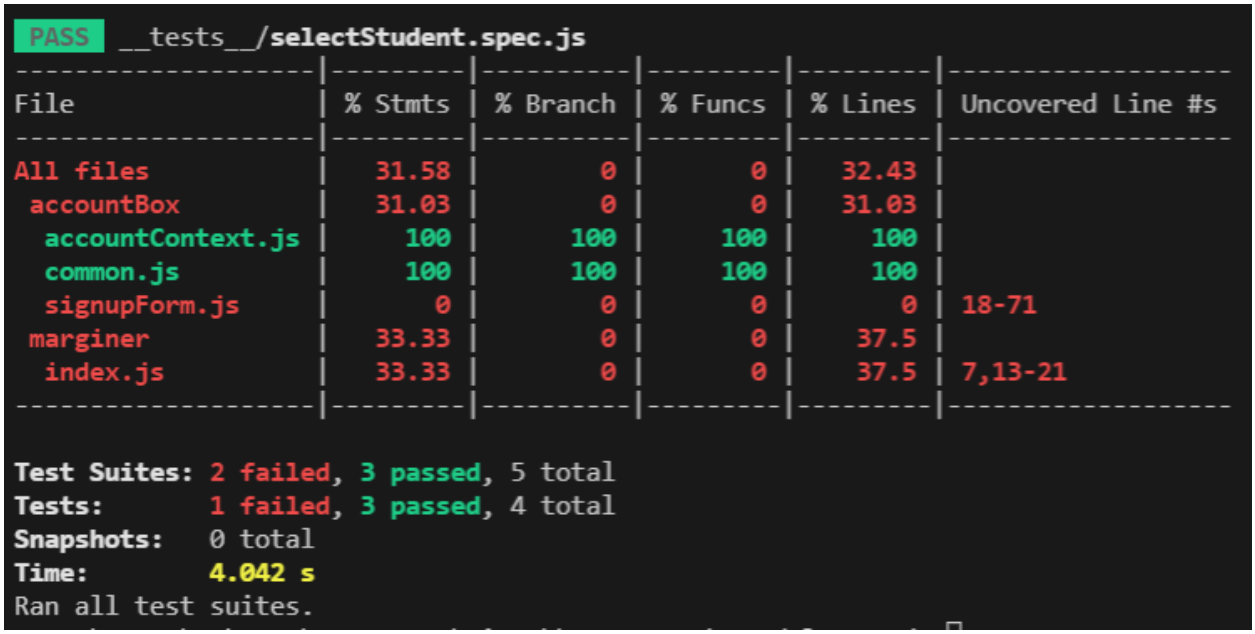


Figure 24: Jest Unit Test Code Coverage

We can see above that the signup form test case was not passed as this time, we had to account for tokenization of the login user information which was unfortunately not completely implemented on time. This problem furthermore expanded into our accountBox as we can see a part of it succeeded, but did not include the necessary security features that we had in mind.

4 Management Volume

4.1 As-build Sprint Structure

4.1.1 Reflection on Agile Evolution

4.1.1.1 Overview of our Feature roadmap:

During our 1 year of product development life cycle we followed our feature roadmap to keep track of different features that needed to be developed and when they needed to be developed. As overview of our feature roadmap is described below:

October 2021: (Initial Setup of the Web Application)

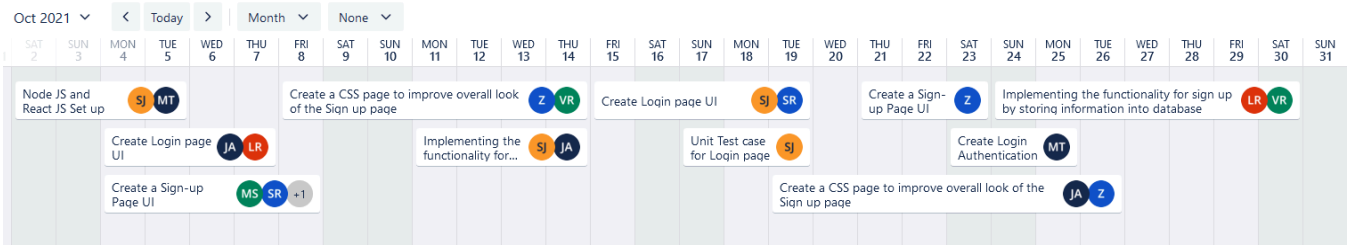


Figure 25: October 2021 roadmap.

November 2021: (Implementing Key Features for the Website such as teacher dash)

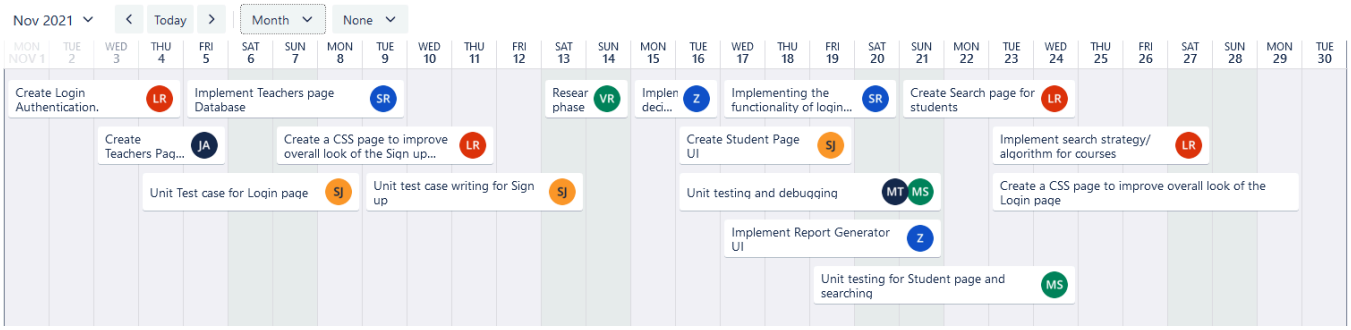


Figure 26: November 2021 roadmap.

January 2022: (Implemented the baseline AI feature and AI Deployment Release)

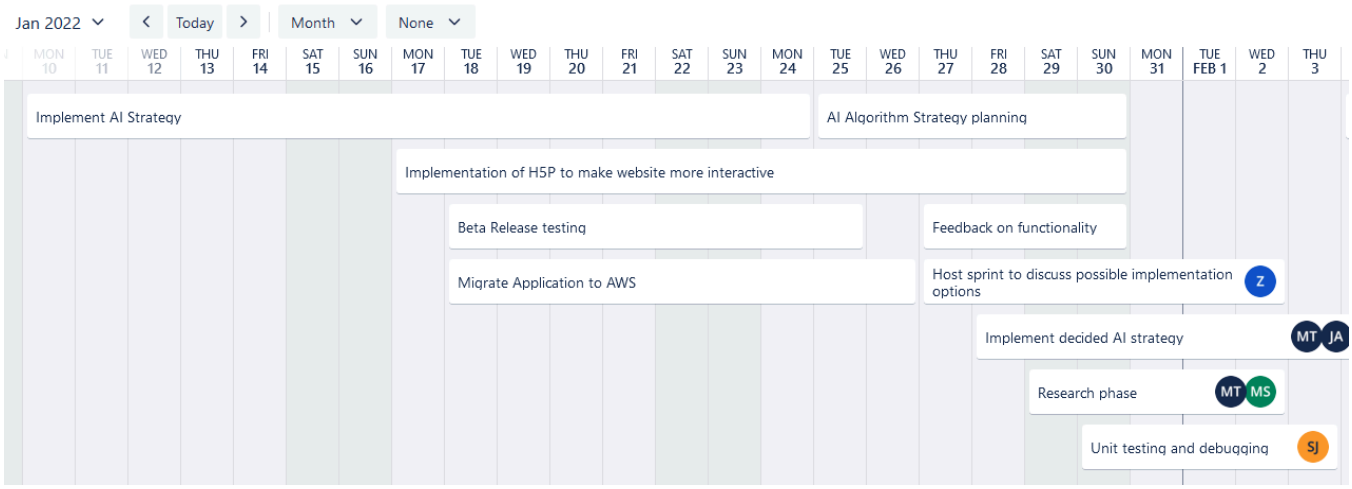


Figure 27: January 2022 roadmap.

February 2022: (Implemented the Student Dashboard Features and Alpha Release)

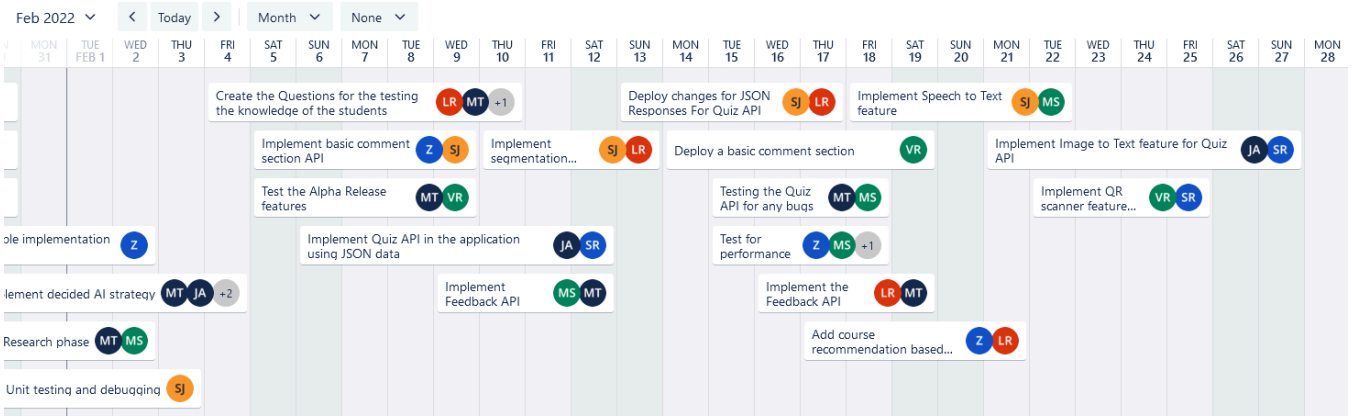


Figure 28: February 2022 roadmap.

March 2022: (Beta release)

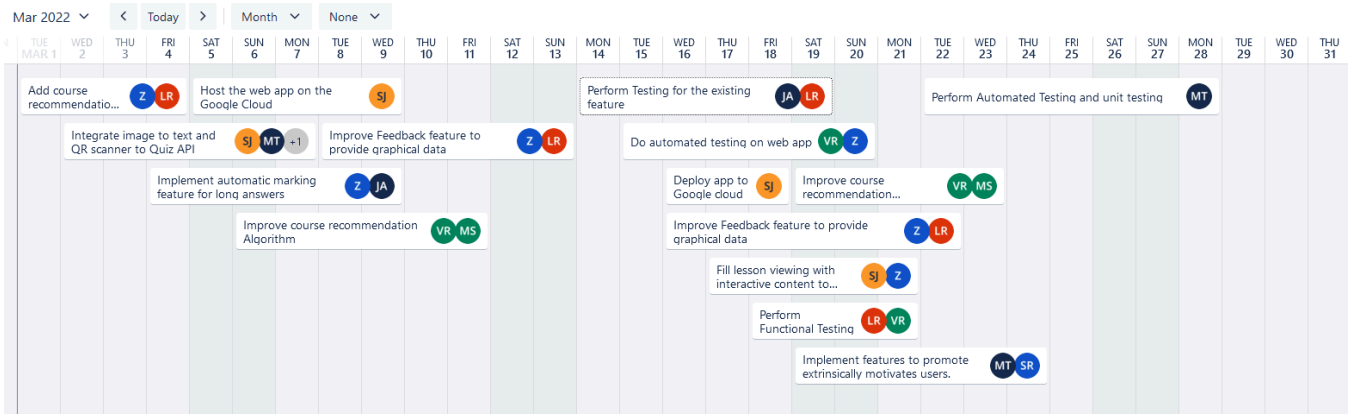


Figure 29: March 2022 roadmap.

April 2022: (Final release)

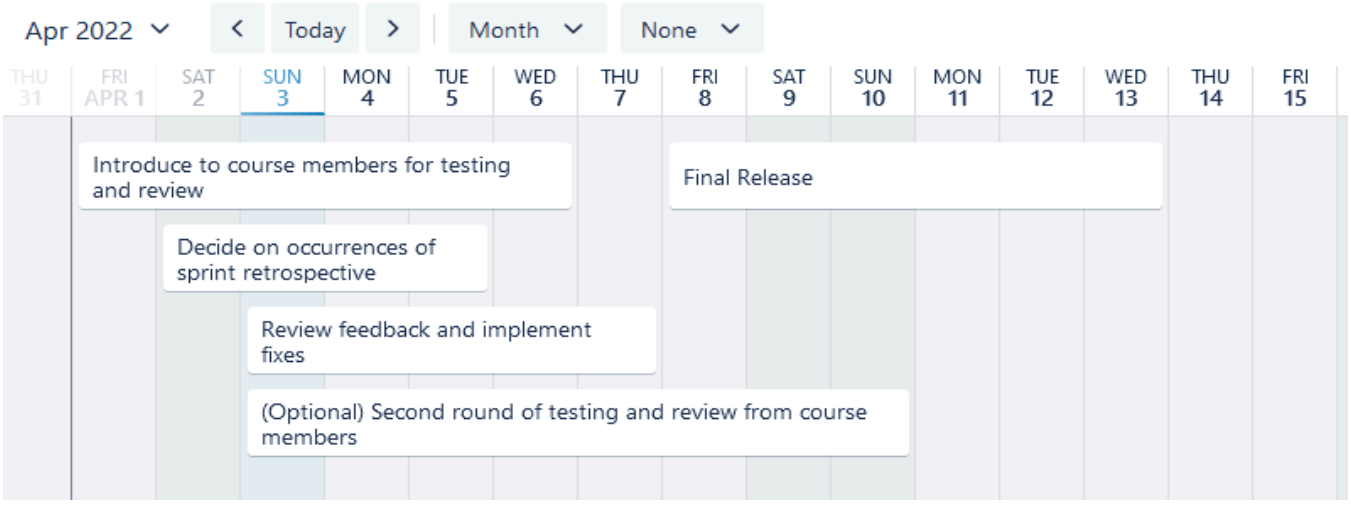


Figure 30: April 2022 roadmap.

4.1.1.2 Changes to Feature Roadmap and Release Plan

During our software development life cycle we had 5 releases. First release was Webapp Deployment, second release was AI implementation deployment, third release was alpha release, fourth release was beta release and final maintenance release was feedback & maintenance release.

Our first release was scheduled for 13 November 2021 and we were able to release our first changes on time that included the basic features such as login, signup and student dashboard. This release was important to be deployed on time because this is the minimum viable product for our project. Some of the AI features such feedback from students were pushed to the next release due to other high priority tasks that needed to be done first.

Our second release was scheduled for 20 January 2022 but we ended up releasing the changes by 30 January 2021. The delay occurred due to the unit testing that needs to be done on our newly added features in our MVP. Unit testing took a little bit longer because testing of newly added AI algorithms is a rigorous process. This release involved the important features of AI such as feedback features from the students and segmentation score implementation.

Third release was our alpha release and was scheduled for 13th february 2022. We were able to release our changes as per the plan because testing for the features was mostly done in the 2nd release. Alpha release involved end to end testing of the software and performance testing of the application using perfecto software. The optimization tasks for the course

recommendation feature was moved to beta release because initial QA testing was of high priority.

Fourth release was scheduled for 15 March 2022 and we were able to release our changes as planned. This was our beta release and included important optimisation changes for some of our features. It includes a feedback feature that generates the bar graph for easy analyzing of the student feedback. Our course recommendation feature was also improved in this release and now it provides more refined recommendations for the courses.

The final release was scheduled for 1st April 2022. This release focuses on the improvement we made after collecting the feedback from the users. This release provides a more stable version for our application and reduces the maintenance tasks of the application. This release also included the optimization changes that were left out in the beta release.

4.1.1.3 Changes to Agile Procedures

At the beginning of the project we planned to run a 2 weeks long spring and we followed the 2 weeks sprint pattern for most of our sprints. Some of the modifications were made to the sprint length to accommodate the high priority changes and to make sure features are tested before releases

Sprint 5:

For instance, our sprint 5 was almost 20 days long starting on 10th January 2022. This sprint was long due to many new AI algorithms that were implemented in the application. All of these new AI algorithms need to be tested before the alpha release to make sure the outputs generated are correct. It took some extra time to find the optimal process to test these algorithms and in order to accommodate these testing for the new features we made a sprint 5, 20 days long.

4.2 As-built Sprint Plans

Our Sprint backlogs were created during the sprint planning at the beginning of every sprint. The Sprint backlog helped our team by narrowing down the tasks and setting up clear objectives for a sprint. We have used a trello board to keep track of backlog for each sprint.

4.2.1 Sprint 1 Backlog:

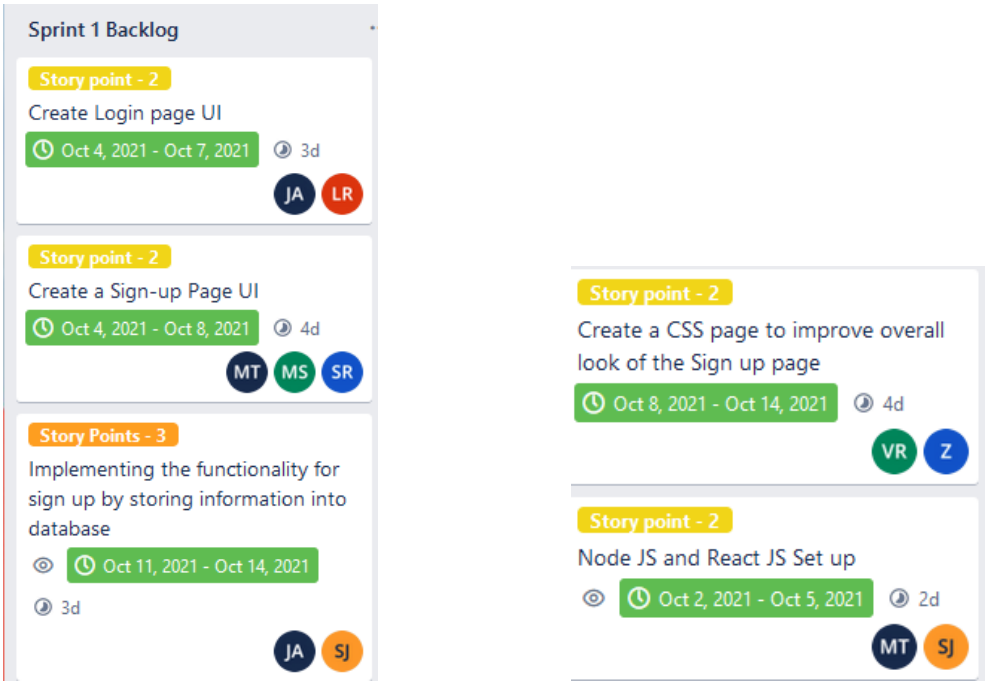


Figure 31: Sprint 1 backlog.

For sprint 1 we targeted to develop a basic UI for sign-up and login page because this was part of the basic structure of our AIM website. For this sprint we focused on setting up a back Node Js and React Js website so that we can have basic websites working.

4.2.2 Sprint 2 Backlog:

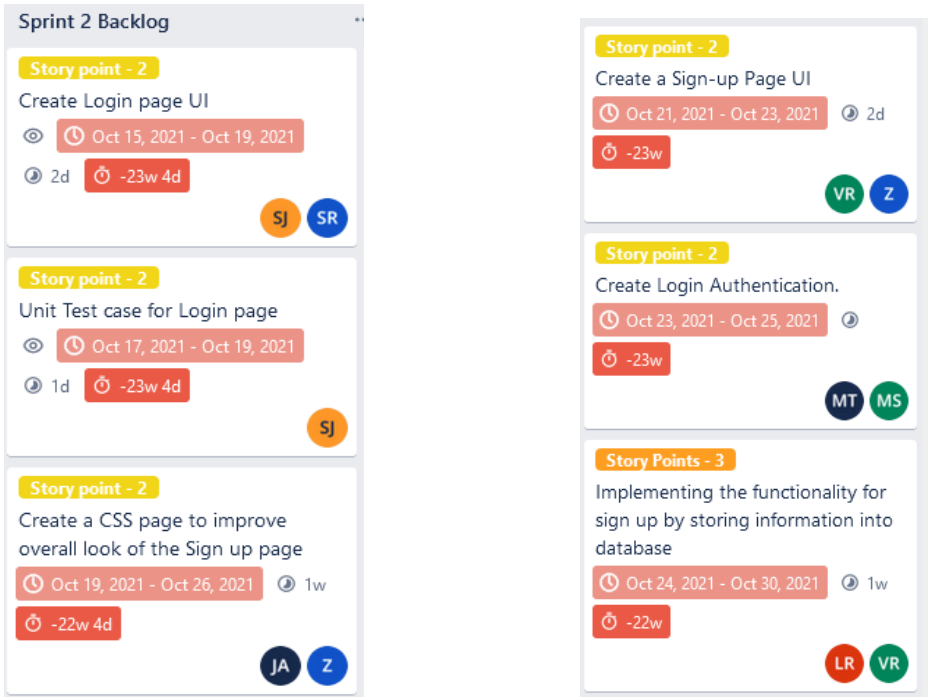


Figure 32: Sprint 2 backlog.

In this sprint our main objective was to implement the functionality of the sign-up and login page. Login authentication was previously planned to be completed in this sprint but due to time limitations it was moved to the next sprint. Unit testing of the newly developed UI was done in this sprint.

4.2.3 Sprint 3 Backlog:

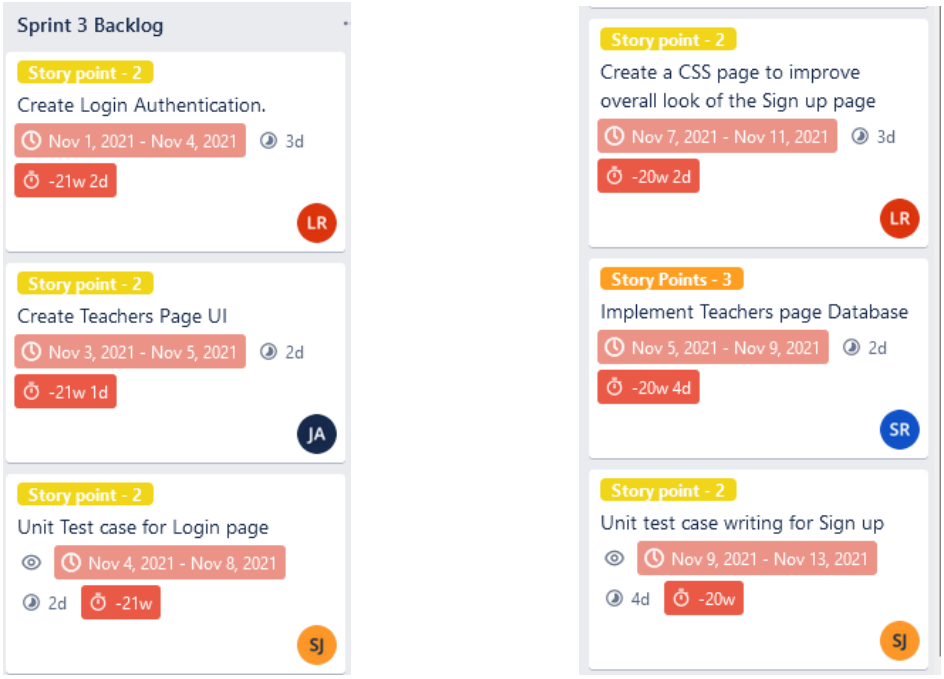


Figure 33: Sprint 3 backlog.

In sprint 3 we were able to implement the teachers UI page along with improving the overall looks of the sign-up page. The left over unit testing of the login authentication implemented in the previous sprint was also finished.

This was important because we did our first release of the application after sprint 3.

4.2.4 Sprint 4 Backlog:

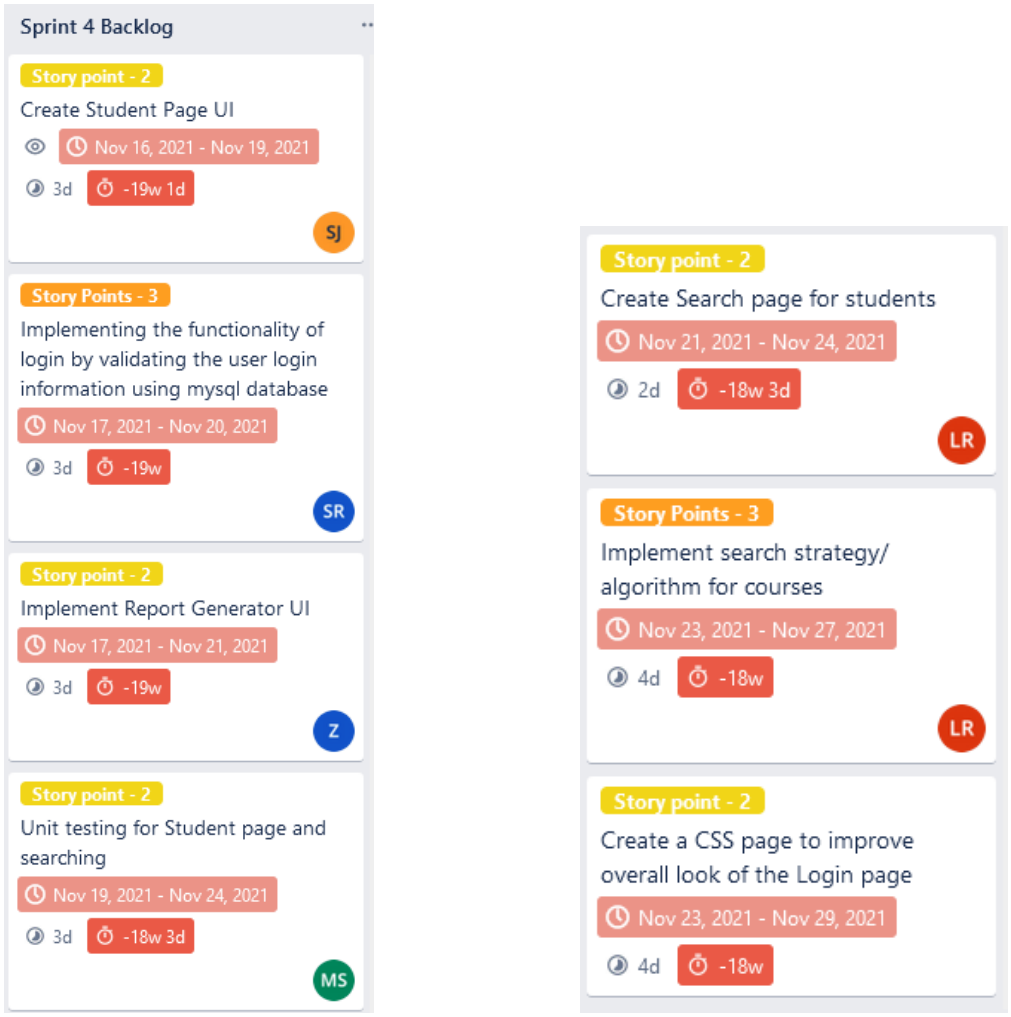


Figure 34: Sprint 4 backlog.

In this sprint we implemented the login authentication to validate the user before allowing signing of the user. Course search feature for searching the available courses was implemented along with its unit testing.

4.2.5 Sprint 5 Backlog:

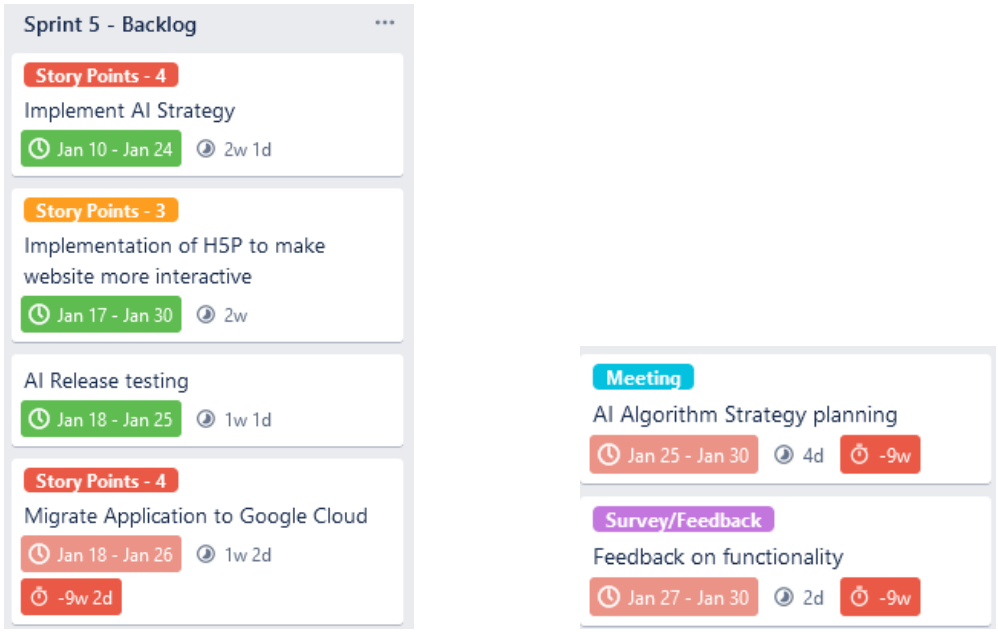


Figure 35: Sprint 5 backlog.

AI algorithm for course recommendation was implemented in the student dashboard and to make the application more robust we migrated our application to Google cloud. Testing for AI release was also finished in this sprint.

4.2.6 Sprint 6 Backlog:

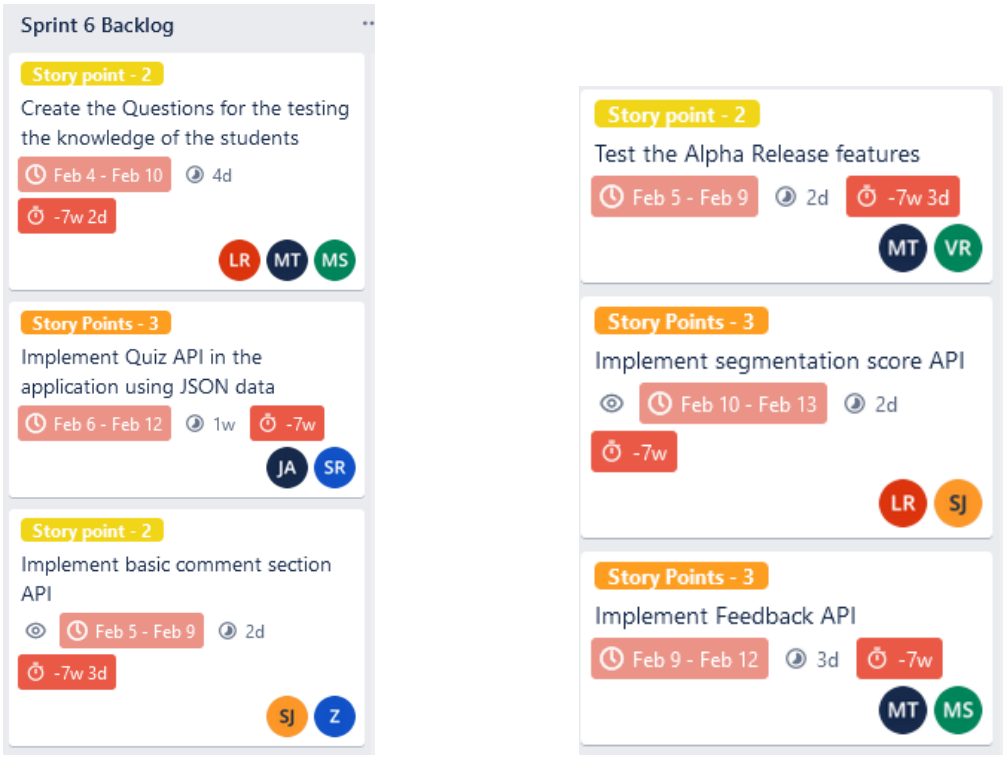


Figure 36: Sprint 6 backlog.

Since Sprint 6 was the last release before the alpha release so we performed the regression testing and unit testing for all the features we finished in the MVP. Segmentation score API which is used in our feedback feature is implemented in this sprint. Quiz API was implemented for testing the knowledge of the students.

4.2.7 Sprint 7 Backlog:

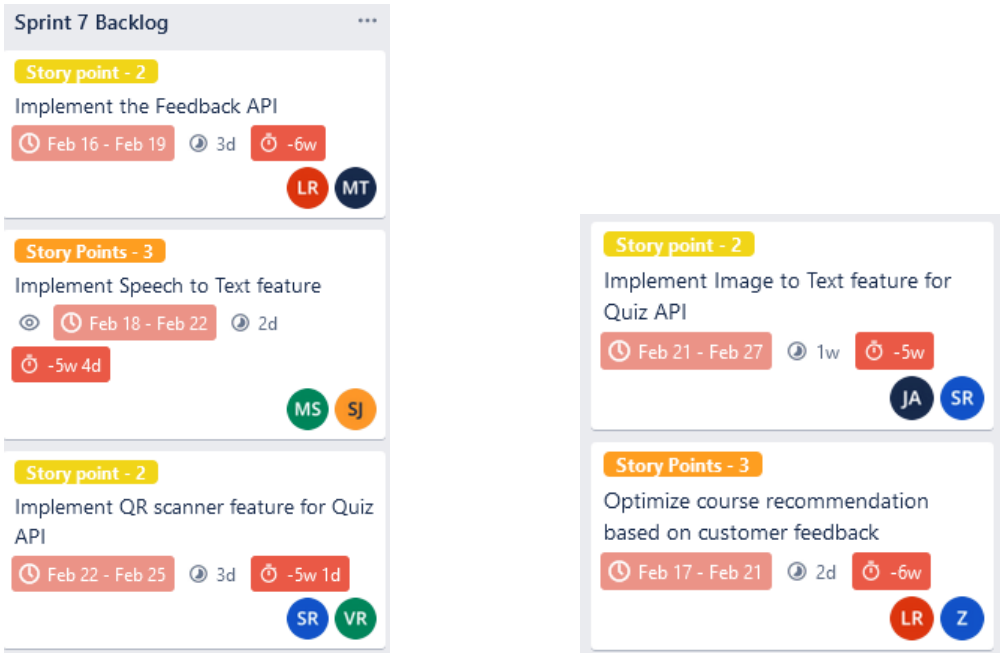


Figure 37: Sprint 7 backlog.

Our final version of the student feedback feature along with its speech to text functionality was implemented in this sprint. To make quizzes easily accessible we also added barcode feature and image to text feature.

4.2.8 Sprint 8 Backlog:

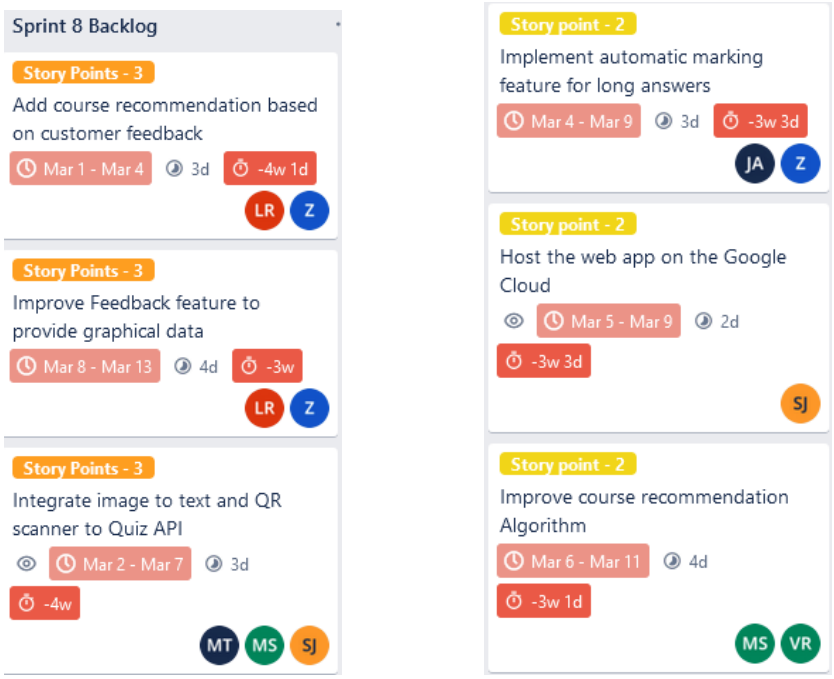


Figure 38: Sprint 8 backlog.

To reduce the efforts of the instructors in marking the long questions we implemented the automatic marking feature using google natural learning algorithm. We also improved the course recommendation feature in this sprint to better provide the course recommendations. Beta release testing was also finished in this print.

4.2.9 Sprint 9 Backlog:

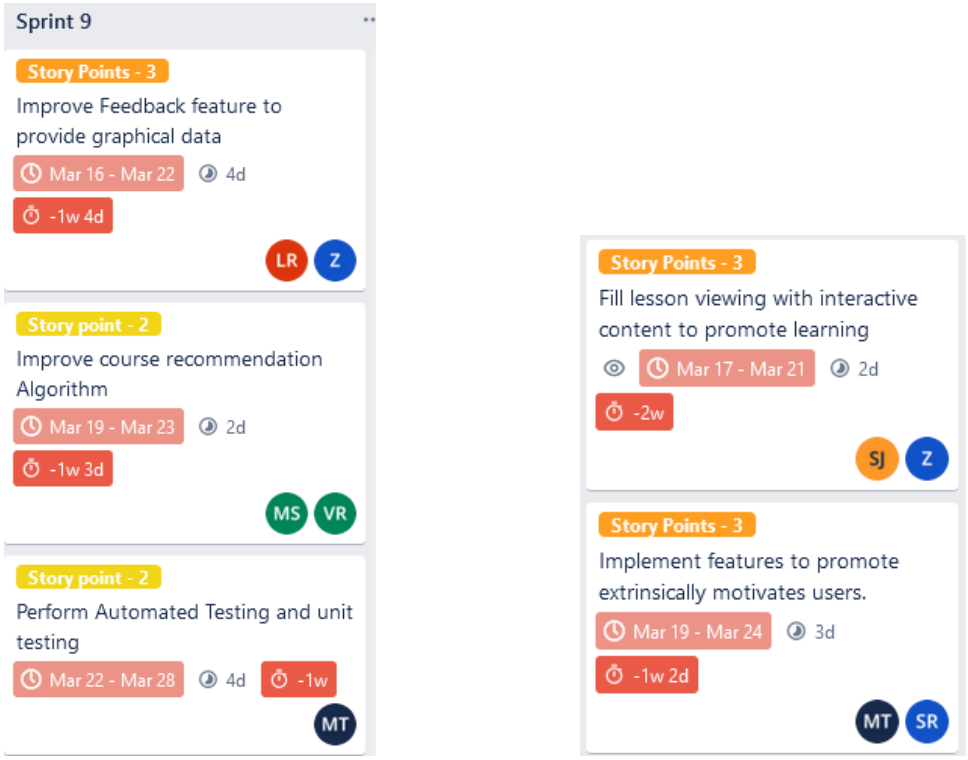


Figure 39: Sprint 9 backlog.

Since this was our final sprint our main focus was to improve the performance of the system and to perform automated tests. Some of the additional features such as in-web video players were implemented in this sprint.

4.3 Changes Made In The Initial Sprint Plan

In some of the sprints the amount of work we added to the sprint backlog was not finished within its respective sprint due other high priority tasks, performing end to end testing and due to other course work.

The changes made to the sprint planning are as follows:

4.3.1 In Sprint 2

In sprint 2 we planned to finish the implementation of the login and sign-up page but due to the initial phase of the project, many of the members were not familiar with the technologies of the project so we had to move the implementation task into the next sprint.

4.3.2 In Sprint 5

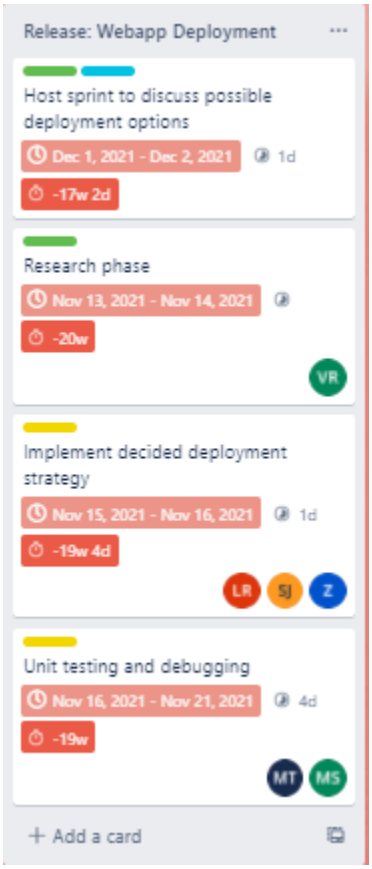
To incorporate the MVP testing and AI release testing the implementation of the text to speech feature for Quiz API was shifted to Sprint 6 so that we could focus more on our MVP release.

4.3.3 In Sprint 7

In sprint 7 we planned to finish the feedback API entirely but due to its dependency on speech to text and segmentation score we were not able to finish this feature entirely and some parts of this feature was transferred to sprint 8. The implementation of the segmentation score was new to all the members so it took more than the anticipated time.

4.4 As-Build Release Description

This section will summarize each release of the project, discussing the main focus of the sprint preceding the release, the achieved goals and deliverables of said release, and the team’s reflection of the completed sprint. It may also discuss the nature of any changes made to the initial release plan if any arise during the sprint.



4.4.1 Web App Deployment Release

The first release has sprints 1 to 3 leading up to it. Each sprint had goals and deliverables dedicated to what the team wanted to have implemented by the first release. The first release's goal was to get the web application deployed locally and get the foundations and basic features set up for both the teacher and student pages.

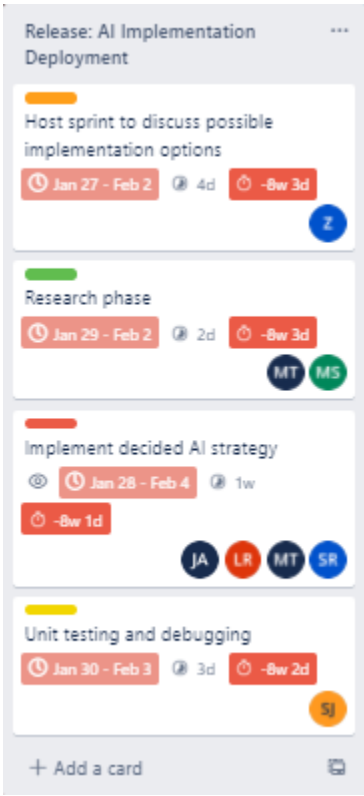
The first sprint's main focus was to create the foundation for the web application and the project. One of the more important events in this sprint ws to meet with the team's supervisor, Dr. James Andrew Smith. With him, the team discussed our view of the project, the features we planned for it, and how we would go about developing our project in an Agile manner. By the end of this sprint, we had a checklist of features we planned to implement for the MVP, got every team member's desktop set up with the tools needed for the project such as Node JS, React JS, a Github repository, MySQL, and a local webapp set up in each desktop.

The main focus of the second sprint was the entry page of the web application. Since the usage of the web application was based on whether the user was a student or teacher, the team figured that working on the login and sign-up page to create these two types of account types early in project development would make our progress smoother down the line. Alongside this, the team also designed a database schema that would be used by both the teacher and student, as well as a rough draft of elements and features that would be included in the teacher and student home pages. By the end of the sprint, we had a functional sign-up and login page that loaded two different pages based on whether the account was created to be a student or a teacher, and an approach for the next sprint based on our UML diagram for the database and the sketch of the home pages.

The main focus of the third sprint was attempting to create a functional teacher's page. This meant creating the database based on the UML design discussed in the previous sprint, following the page flow for teachers, and incorporating the elements and features for the teacher page according to the rough draft we generated in the previous sprint. During this sprint, we also planned on the MVP requirements and what features would be necessary to implement in time for the MVP release. The work flow for this sprint required the back-end team to create a very rough skeleton for the teacher's page with buttons, display panels, and text boxes that would be used for I/O connected to the database. The back-end team then

hands this off to the front-end team to improve its UI and unify its design across other pages such as the login/sign-up and the soon-to-be-developed student page. During this, the back-end team would work on creating the database and ensuring the GET and POST methods functioned properly with the teacher page.

4.4.2 AI Implementation Deployment Release

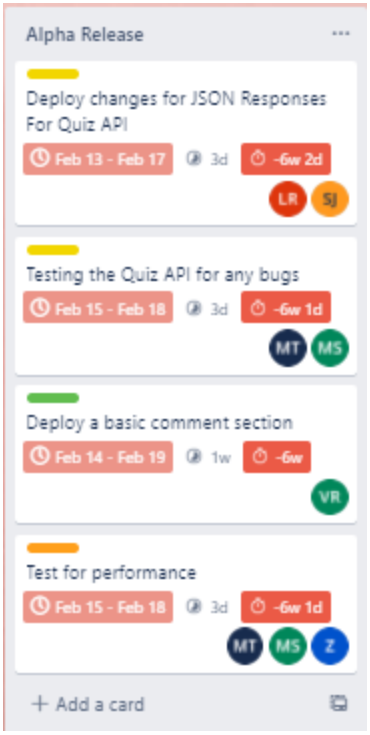


The second release had sprint 4 and 5 leading up to it. The goal of this release was to implement some of the features that utilized AI/Machine-Learning.

The main focus of the fourth sprint was the creation of the student page. The process for this was similar to sprint 3's for the teacher's page. In the scrum meeting, the teams discussed the UML diagram for the student's database, how that would interact with the teacher's database, and the page flow for students. The back-end team created the skeleton of the page, the front-end team applied the design and layout over it while the back-end team worked on making the proper API calls and database connections to the interactable page elements.

The main focus of the fifth sprint was implementing more features into the teacher page. New features added to the teacher's page included a file upload system for teachers to upload files to their courses. We also made the course page display the lessons hosted under it, fully integrating lessons with courses rather than keeping them as separate entities. Lastly, we started testing with sentiment analysis and text-to-speech, both of which utilize AI methods. These tests were done locally, ensuring we understood its functionality before attempting to implement it to the web application. By the time the release was to be deployed, we implemented sentiment analysis and text-to-speech into the web application after testing and understanding its functionality.

4.4.3 Alpha Release

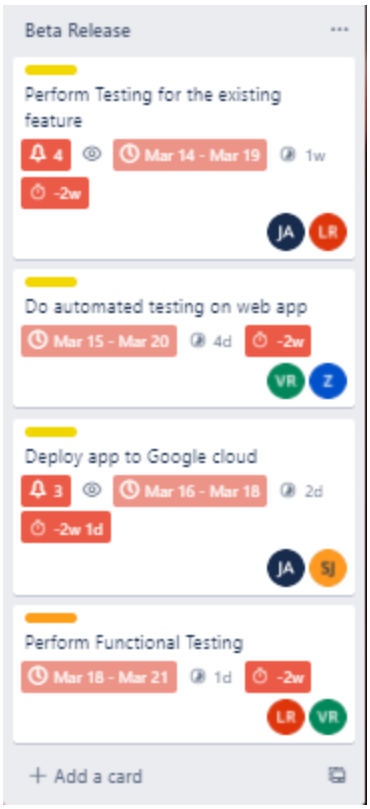


The alpha release only had one sprint to work on it, sprint 6. The goal of this release was to deploy more features that made the web application more interactive. Thus, we added the quiz API,a comment section to allow more ways for students and teachers to interact, and implemented the course recommendation feature that utilized machine-learning.

The main focus of the sixth sprint was implementing AI-based features and a quiz API on the student's side. During the meeting, we decided to implement the quiz and comment section feature from the previous sprint's reflection. The quiz was to be used not only to ensure students understood the lesson content, but to also give students some extrinsic motivation to study, as discussed with Dr. James regarding students with intrinsic motivation versus those that require extrinsic motivation. The comment section will be used for the sentiment analysis that teachers will be able to view and reflect on.

We also added a course recommendation feature using a frequency-based tag search that learns and uses the student's previous search inputs to provide better course recommendations. How this algorithm works is explained in the *As-Build Release Description* talking about the course recommendation feature.

4.4.4 Beta Release



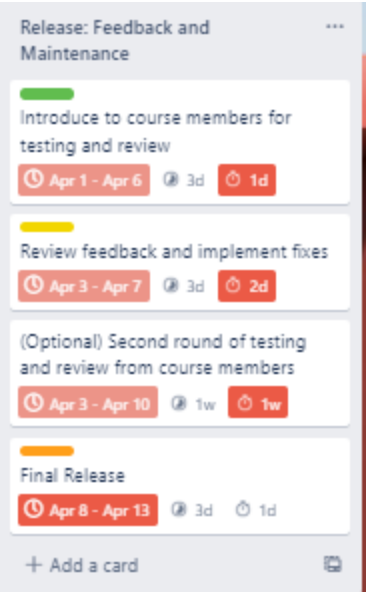
The beta release had sprints 7 and 8 focused on its features. The goal of the beta release was to have a product closer to the original vision of the project. Thus, a lot more focus was placed on improving features that promoted learning, such as the quiz API, as well as deploying the web application to the cloud. Another focus for the beta release was ensuring web application stability once deployed to the cloud. This meant performing tests such as automated unit tests, functional tests, a web application walkthrough for any bugs or errors, and coverage tests. These were also performed and recorded for the beta release document. These results can be seen under *Performance* in the *As-Built Design* section.

The main focus of the seventh sprint was improving the quiz functionality and implementing the Feedback API using the sentiment score AI analysis. Testing the sentiment score from Google's API started in sprint 6 and is now implemented into the Feedback API for the teacher's use in this sprint. The Feedback API gives the teacher's course a sentiment score based on the average sentiment score of the reviews and comments under each lesson. This allows the teacher to easily analyze their performance and course material in a visual quantifiable and visual manner. Features were added to the quiz API, including the previously discussed mobile version using a QR scanner, an image-to-text feature to allow students to upload an image and compare the converted image-to-text result to the desired answer, and a speech-to-text feature allowing students to vocalize their answer instead of typing it.

The main focus of the eighth sprint was deploying the web application to the cloud and further improving the quiz API features. Using the provided project budget by the school, the team rented a temporary domain from the Google Cloud to host the web application and make it accessible to other students to use and provide reviews for, which will be used in the future. The features for the quiz API developed in the past releases were previously separate pages meant for testing and ensuring functionality. In this sprint, after confirming they all worked properly, we integrated them all into the quiz API, making the quiz API more interactive than simply its base function of the user clicking their answer on the quiz.

We also added other features to the quiz API such as using an AI-based string similarity algorithm to grade open-ended long answers. This expanded the question range of the quiz beyond multiple choice and true-or-false.

4.4.5 Final Release



The final release had only one sprint, sprint 9, to work on its deliverables. The focus of the final release was to add any feature improvements in the backlog, ensure all features were working properly, web application stability, and rounds of student reviews to get opinions and possible improvements on the web application.

The main focus of the ninth and last release is improving any feature doable within the week's span. The Feedback API was improved to include a graph display to make it easier for teachers to review the sentiment score of their courses. The lesson view page was improved to allow youtube streaming so that teachers can provide youtube video links of their lectures or resources and students can watch it without having to leave the web application. Features that targeted extrinsically motivated users were also added to the main page of the student page. This included a weekly score system that tracked their progress and improvement in courses based on their quiz scores.

The last thing to be done was releasing the web application to the course members and receiving their reviews, reading through the received reviews from the students and implementing any improvements and bug fixes. If time permitted, a second round of testing and review could be performed to further improve the web application. Each crowd test was expected to last 5 days before we would work on the received reviews. Reviews mainly centered around adding more features that could improve the student experience, and unifying the design across teacher and student pages. The team decided that adding new features could be risky as this might cause the new version to not work properly before the final release, and unifying the design between student and teacher pages would be time consuming as there were multiple pages to work on on the teacher's side. As such, no improvements were made based on the reviews.

4.5 As-Build Resource Allocation Matrix

The following table is a resource allocation matrix which displays the total number of hours spent by each team member per sprint. Tasks were assigned and had its completion status updated using the group's Trello board (Link: https://trello.com/b/xDzbgmTe/web-application-eng-4k).

Name	September	October		November		December
	Introductions and Brainstorming	Sprint 1 (4th -14th)	Sprint 2 (15th - 30th)	Sprint 3 (1st - 13th)	Sprint 4 (16th - 29th)	
Jihal P.	8	6	5	4	6	0
Joshua A.	8	6	6	6	5	0
Lukas R.	8	7	6	7	7	0
Mike S.	8	6	6	5	6	0
Phuong T.	8	6	5	5	4	0
Samyak J.	8	6	7	7	6	0
Sharujan R.	8	5	6	7	6	0
Varuhn R.	8	5	7	5	4	0

Name	January	February		March		April
	Sprint 5 (10th - 25th)	Sprint 6 (4th -13th)	Sprint 7 (16th - 27th)	Sprint 8 (1st - 13th)	Sprint 9 (16th - 28th)	Final Release Document
Jihal P.	7	6	5	7	7	6
Joshua A.	8	7	6	6	5	6
Lukas R.	7	6	6	5	6	6
Mike S.	7	7	6	6	6	6
Phuong T.	7	9	6	6	7	6
Samyak J.	6	7	6	7	6	6
Sharujan R.	6	5	6	5	6	6
Varuhn R.	6	5	6	5	4	6

Figure 40: Resource allocation matrix.

Each sprint begins with a 1 to 2-hour scrum meeting. Team members discuss the backlog of uncompleted deliverables from the previous sprint, the goals and deliverables from the next sprint, and allocation of tasks between subgroups. For example, improving the UI and CSS of the student page would be assigned to the more frontend-oriented members while managing the backend database connections of the teacher page would be assigned to the more backend-oriented members. Sprint durations would run between 1 to 2 weeks depending on the perceived difficulty of the deliverables. Members were expected to work a minimum of 5 hours per sprint.

4.5.1 Changes to Initial Release Plans

February (Sprints 6, 7)

With the final release date nearing, many of the decorative and secondary goals were put aside to focus on more primary goals. Secondary goals such as improving the UI of the teacher and student pages, including an email-based sign-in and sign-up function, and making some features take input from the user were no longer assigned to members in the Trello board. Hardcoding inputs to features and ensuring they were properly processed and displayed in the UI and implementing the remaining features that were meant to be in the final

product became the main priorities for February and onwards. We also focused more on improving the quiz API used by the students from sprint 7 going forward as we believed this feature is what would set our web app apart from other educational platforms.

4.6 As-Build Project Schedule

Working with the Agile methodology, the team had a list of features we planned to implement into the web application. Over the course of 9 sprints, we attempted to gradually implement each one and also discussed additional features we could add as we saw fit. Due to the flexible nature of the Agile methodology, we used a Trello (**Link**: https://trello.com/invite/b/xDzbgmTe/051f679f1057166ebb2a05c3e4177701/web-application-eng-4k) board to keep track of our progress, what deliverables were completed, what deliverables were in the backlog, and keep track of the number of sprints remaining before the final release.

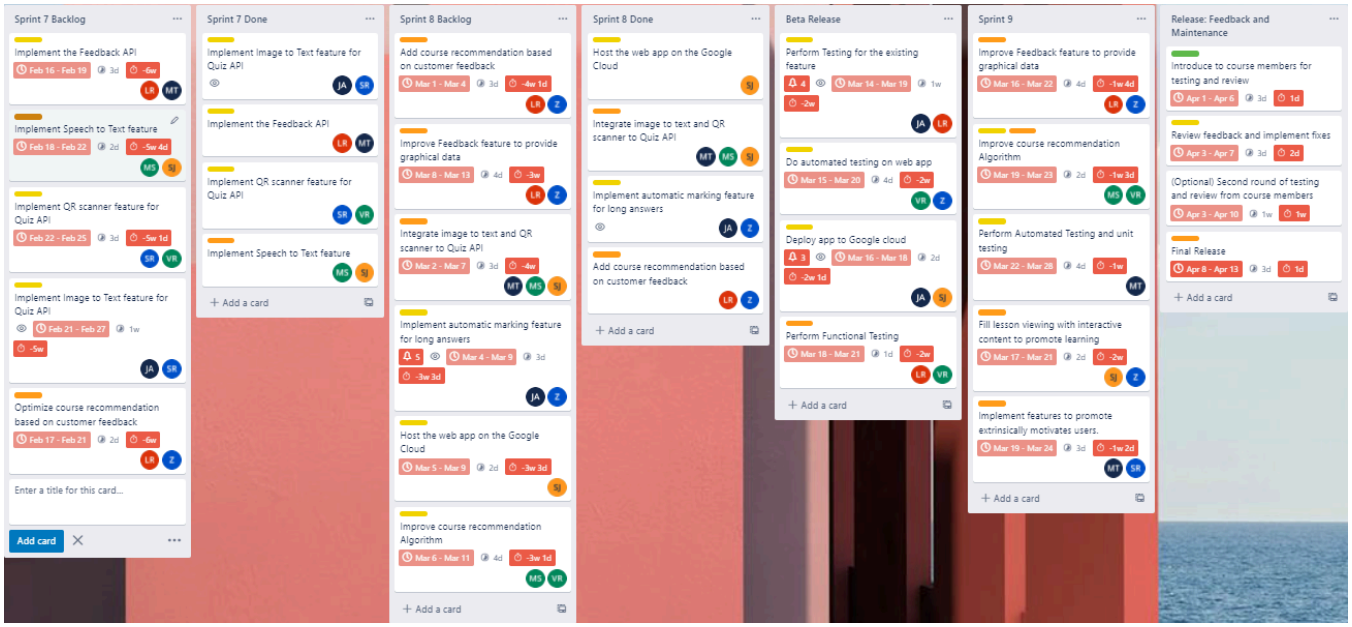


Figure 41: A snippet of the cards used in Trello to keep track of sprint deliverables and backlogs.

4.6.1 Burndown Graph

An efficient way to keep track of our progress in the Agile methodology is through a burndown graph, visually representing how quick the team works through our initial deliverables, and how much was completed by the end of the project. The burndown chart was made by comparing the number of tasks in the backlog of each sprint versus how much was completed in each sprint. Since deliverables were discussed and created dynamically during meetings, the “To Do” list is not a consistent number but varies according to what the team thinks is needed to implement in the project at that time. Therefore, the backlog count may change depending on whether the team adds new deliverables or removes old, unnecessary ones. This explains why

in the middle of the project, the To Do list hovers around the 6-7 range, since we came up with more features to add or improve in the project, but near the end, hovers around 4-5 since all that was left to do was maintenance and review inspections.

	Sprint	To Do	Completed	Remaining To Do	Completed Progress
	Sprint 1	5	6	59	58
	Sprint 2	6	4	53	54
	Sprint 3	6	7	47	47
	Sprint 4	7	6	40	41
	Sprint 5	6	5	34	36
	Sprint 6	6	6	28	30
	Alpha	4	4	24	26
	Sprint 7	5	4	19	22
	Sprint 8	6	4	13	18
	Beta	4	4	9	14
	Sprint 9	5	6	4	8
	Final	4	3	0	5
Total		64			

Figure 42: The numbers chart used to generate the burndown graph.

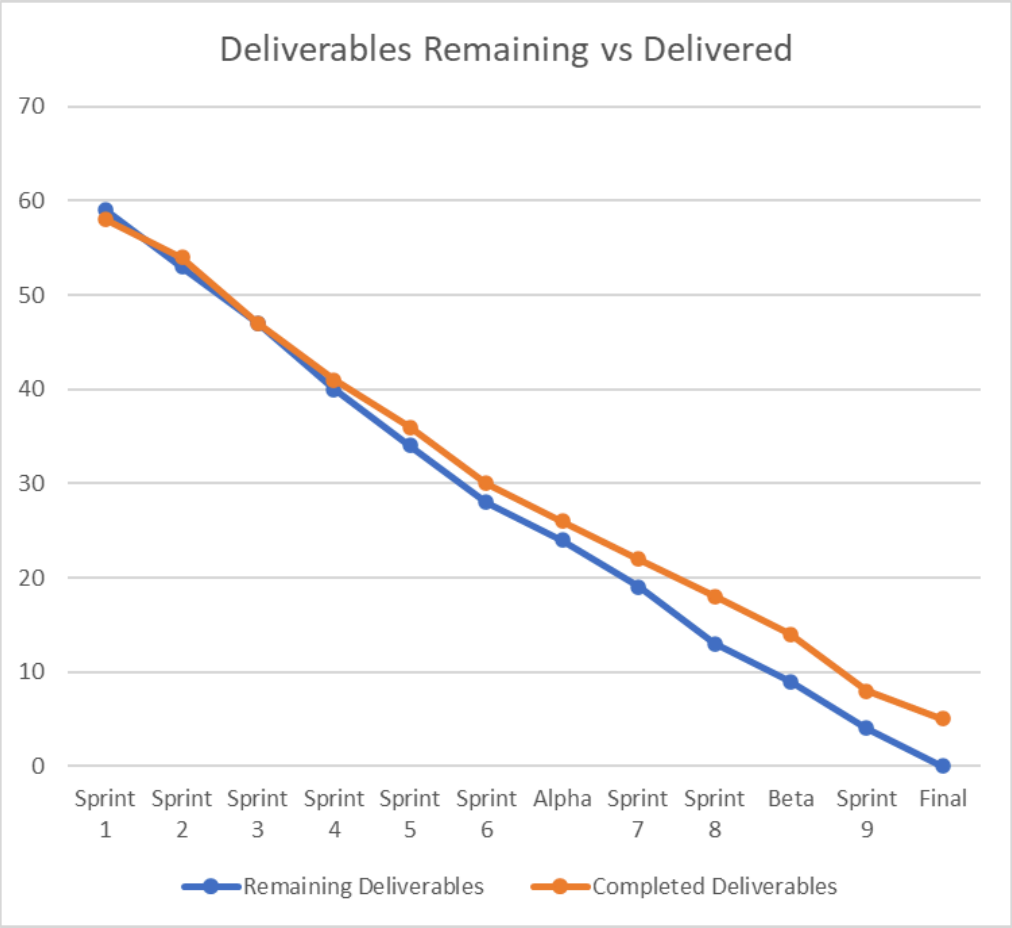


Figure 43: Burndown graph representing the team's progress through the planned deliverables versus how much was actually achieved per release.

Overall Work Division between Team Members based on Story Points:

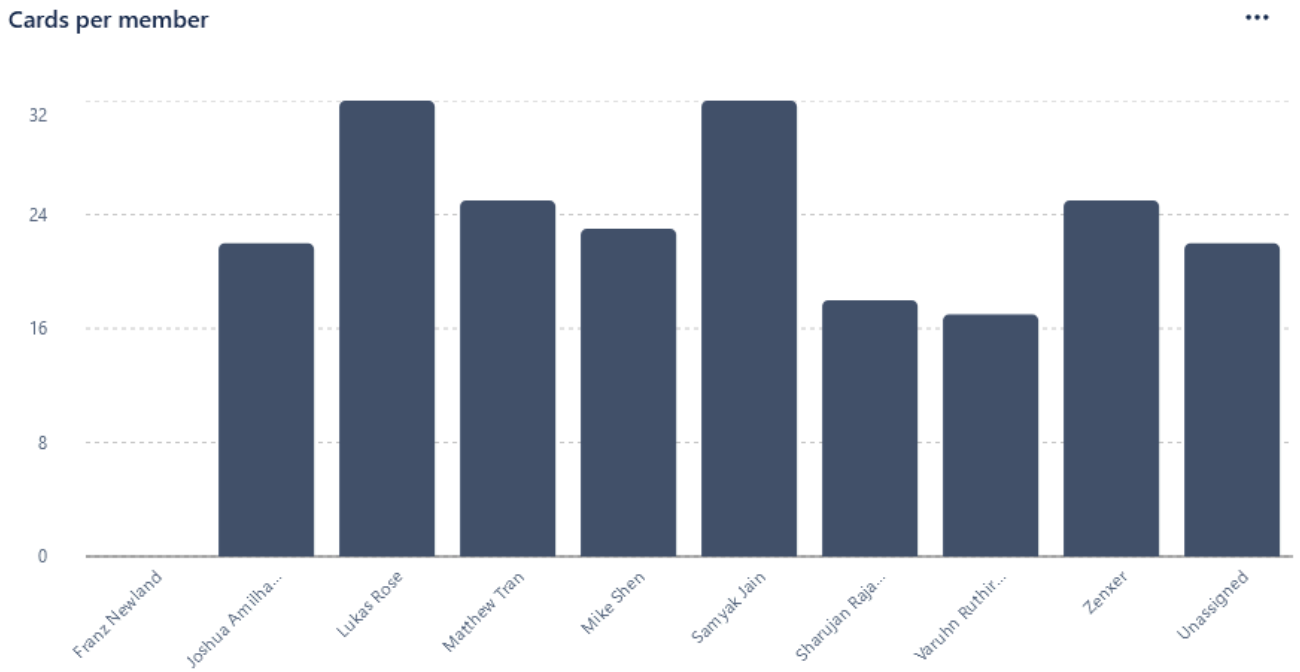


Figure 44: Work division between team members. (Note: Zenxer is Jihal Patel.)

As we can see, division of the work between every member is fairly equal and this is due to equal division of tasks during the sprint planning and correctly analyzing the workload for every member.

The allocation of these tickets were based on story point and we make sure every member gets an equal amount of story points assigned during a sprint.

4.7 Project Procurement List

The cost of our product is based on monthly expenses as we are currently paying for a cloud service subscription and other web hosting website subscriptions. Costs of developing are not included within the list. These figures will be constantly changing as user count fluctuation will increase and decrease costs of running the website.

Expenses	Cost/Price in CAD	Why do we need this
Cloud services like Google Cloud	$10 per month	We need an EC2 web server which is provided by Amazon web services so that we can run our website on the cloud instead of local instances.
Domain name	$15 per month	To host our website to our users, preferably a domain name that resembles our product name.
Web Hosting	$10 per month	So that our website is accessible and searchable to users through our domain name.
	Total - $35 per month $420 per year	

Table 1: Project Expenses.

4.8 Preliminary Business Case

The following section will be covering the project going forward. It will discuss the new changes we aim to make on our existing product to make it even better as well as opportunities that our product has. These all will take into consideration what the team believes our stakeholders needs are and what changes are most suitable using the strength-weakness-opportunity-threat analysis.

4.8.1 Feedback Plugin

A major feature of our current product is the teacher AI feedback feature, as mentioned earlier in the report. Going forward the team has discussed that we should make this a separate plugin. Plugins will offer us more flexibility and as well as save time, not to forget to mention that it will be more reliable. This is one of the major changes our team has aimed to implement going forward.

4.8.2 User Interface

When the appropriate time comes we want to move on from the current visual design of the user interface and present something more sleek and simple. While the green-black-white color combination is eye-catching, it lacks professionalism and simplicity which are trademarks for platforms in the education sector. An overhaul of the user interface is definitely something our team is eyeing going forward with our product to attract users.

4.8.3 Mobile Friendliness

A key implementation of any major web platform is to make it accessible on any device, including the popular mobile smartphone. Currently, our product lacks in giving the mobile user the same experience as on desktop. This has been taken into account and thus is another area of improvement for the future of our product. The decision of making it accessible on a mobile browser app, developing an app, or both is still to be made by the team.

4.9 Risk Monitoring

This section is to display our updated risk monitoring which is going to be used for future changes and implementations to the existing product. The practice of tracking and evaluating the levels of risk in an organization is known as Risk Monitoring. The Risk in the project life cycle could occur in any of the 5 stages -- Planning, Requirements, Design, Build, Document, Test, Deploy, Maintain. We will be focusing on the Maintain stage risks.

Risk Involved	Where it could happen	Strategy to Mitigate the Risk
1) Database could get corrupted due to incorrect data storage	In the maintain phase of the project	We can fix the format of the input in which we will accept the user input. For example in simple text or in PDF format etc.
2) Data might get leaked to the hacker.	In the maintain phase	Different vulnerability tests for different cyber attacks like SQL injection should be performed.
3) Production and deployment issues could occur as amazon web services may not be compatible with our website.	In the maintain phase	Proper testing servers should be used so that at the time of deployment the risk of failure is minimized.
4) Accessibility issues could occur as our server capacity	In the maintaining phase	We should use big data centers such as amazon web

could be limited to serve a small number of users at a time.		services where we can host our website and it is accessible to a larger number of clients.
5) Data name duplication	In the maintain phase	Name duplication is minimal as it could possibly cause confusion to users but no deeper issues. Having username never duplicated is more important and needs to be safekept.
6) Web platform crash caused by server failure	In the maintain phase	This is very unlikely but if were to occur it would be caused strictly by the failure of Google servers.

Table 2: Risk Monitoring during Maintain Phase.

Link to access Risk Matrix Board:https://app.conceptboard.com/board/y7x9-eae0-h3ob-dtsf-4ik6

Figure 45: Risk Assessment Matrix.

5 Lessons Learned

5.1 Deviations From Plan

This section will be a summary of the more significant deviations from the design plan. It will cover major changes of the design as well from the initial planning stages as we found it was important to mention changes during that phase too.

5.1.1 Gamification

In the early stages of our project we underwent multiple meetings where our main objective was to make requirements and objectives that we want our final product to meet. During these meetings we came upon the idea of designing the product around gamification. The team researched the use of gamification and found positive benefits as can be seen from a diagram we viewed:

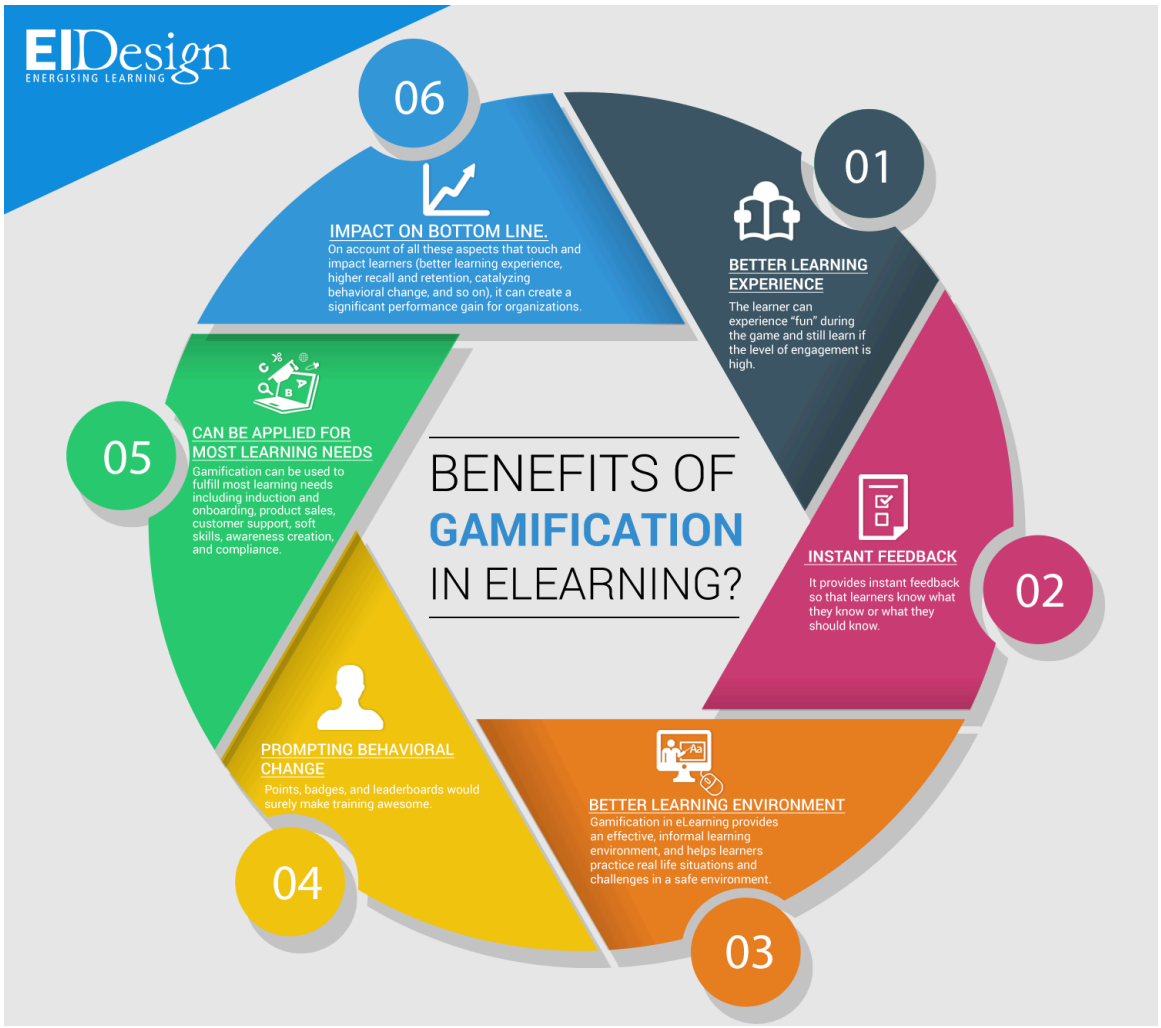


Figure 46: Diagram Depicting the Benefits of Gamification

The whole team saw great value in this idea so we pursued it further. Our reasoning behind gamification is that it would increase user engagement with our product because it is found that about 70% of online learners reported themselves not being engaged, according to an online poll on Gallup. We saw an opportunity, at that moment, to be innovative and implement gamification to our educational web platform. Here are some of the ideas we came upon during our brainstorming:

- User progression by score
    - Receive points to increase scores from:
        - Quizzes
        - Course Achievements
        - Course games

- Ranking system
    - Users could climb up the ranks
    - Comprised of course completion and scores

The team was very hopeful with this idea, but we soon realized our lack of experience would hinder the ability to gamify every aspect of the product. We brought our ideas and questions to our supervisor, Professor James Smith, during a meeting and he gave us some great feedback and things to think and discuss within the team. These included:

- Who is our customer?
    - What age range is our customer?
    - Will our customer need encouragement to learn?
    -
- Will gamification be non-generic?
    - Point systems/badges/leaderboards are not very thoughtful
    - Could overlook the importance of creating a fun environment while learning
    - If generic, customers will quickly lose interest
    -
- Will it be an option of gamification or forced on every user?
    - Not everyone needs the extra encouragement to learn
    - What if the user is self-deterministic?
    - Will teachers be required to gamify elements of their course?

After discussing our supervisors feedback and brainstorming questions we realized that, while it seemed so promising, we completely overlooked the amount of work that would go into designing our platform around gamified elements. And on top of this, we realized that our

gamification would have a very high chance of failing as it is a hard task to engage users on a regular basis and not wear off.

After consideration and identifying the cost vs. value of gamification, we decided it would not be a smart decision to undergo such an in depth and detail-filled task, given that it might not bring positive value to the platform in the long run. We learned alot from this deviation early on in the project. We knew that the customer niche we will be pursuing is a self-deterministic person, this way we will not need to add gamified elements. We decided to still aim at creating user engagement but more through our artificial intelligence elements in our platform.

This was the first major deviation that helped us set our direction for our AI education platform.

5.1.2 UI Overhaul

The deviation covered under this subsection occurred towards the end of the semester.

During the development leading up to the ‘Minimum Viable Product’ gate the frontend team members implemented a user interface that we were all happy with for the MVP product. While it was a good interface for the user at that stage of our product, we did agree that we could have a more appealing look and design to attract customers. The website user interface we were aiming for is sleek, modern, and minimalistic.

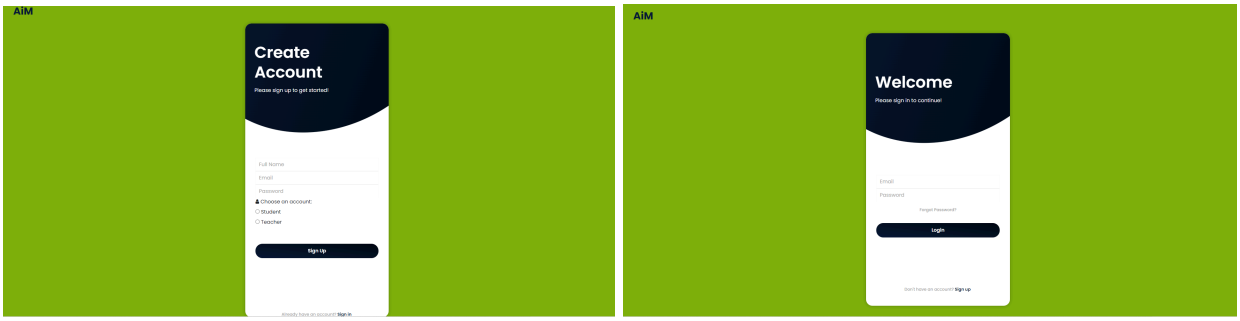


Figure 47: MVP Sign-up and Login Pages.

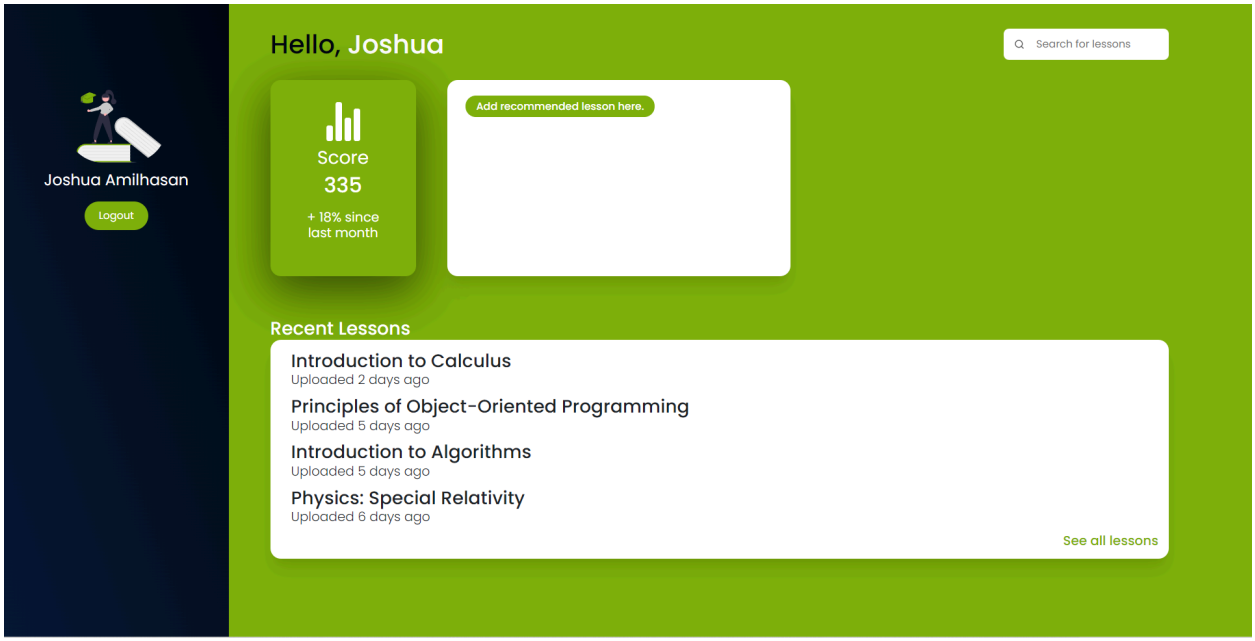


Figure 48: MVP Home Page

This was initially set to the side because it was not of highest priority. Instead the team focussed on implementing more main features of the product as well as improving existing ones. While this was the right decision to make, the team underestimated the amount of work some features would need and eventually we did not have the time to focus on smaller priority goals.

We are glad to say that our platform features were completed and that our initial user interface still was an overall good design, but nonetheless our user interface overhaul had to be deviated from and instead work on the features that our product would showcase because of time constraints.

5.2 Failure Report

This section will cover and summarize some of the bigger failures our team had during the project.

5.2.1 Email Authentication/Verification

The team decided after the minimum viable product submission that we should add email authentication within a new user's registration as it has become a staple in registration processes everywhere and would help with the security aspects of the product. Research found even further good reasons to implement the small feature.

The goal seemed attainable during the meeting we had discussed the addition of the feature. But because of other, higher priority, features falling behind on schedule we had to leave this email authentication feature for the moment.

Once we did get to developing the feature, we had a small amount of time remaining to implement the feature and when we encountered problems with the email sending and receiving input, we ultimately decided to cancel the feature altogether for the final release.

Figure 49: Benefits of Email Verification

5.2.2 Data Encryption

Another implementation we aimed to add to our product was the use of data encryption. We have a couple of students who have experience in the cybersecurity side of a web platform so we saw it as a perfect opportunity to use the strengths of the team members for the betterment of our product.

While this also was looking like a sure added feature, the combination of roadblocks and time constraints caused the team to have to abandon the implementation. The team members were needed in other fields of the product which the team deemed more vital to our final gate so the decision had to be made. The team will have to weighencr the cost and value of possibly implementing data encryption further down the road. As of now, our research has only shown positive outcomes that could come from data encrypting on our database uploads. The main advantages of implementing this can be seen on the following diagram:

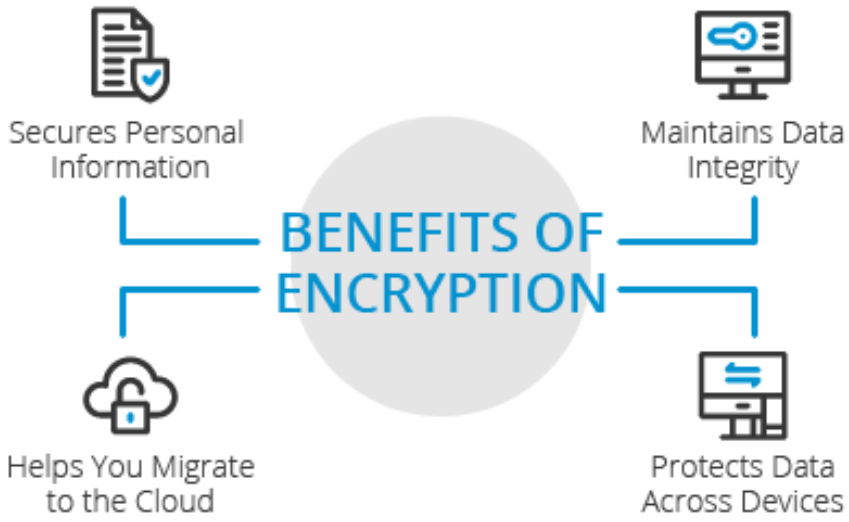


Figure 50: Benefits of Encryption

5.3 Lessons Learned

This section is a final team reflection on how the lessons we learned during the project would have helped us run the project phases differently if we had the hindsight available to us. We will also suggest how we should run our next project differently.

The section is split into subsections, each focusing on an area and the changes we would make and suggestions for that area.

5.3.1 Realistic Goal Setting

A lesson that our team learned was based around the goals we had set at the initial stage of planning. We were very optimistic with every idea that was mentioned in initial meetings so our goals quickly became to an unrealistic level for the time frame we were given.

An example that came from this was mentioned in Section 5.1 where we had initially aimed at creating gamified elements into the whole platform. This idea was fortunately put into perspective quickly with the amount of work needed and we deviated from that idea.

The team learned the valuable lesson that every aspect of the project needs to be taken into consideration when setting goals. From the amount of time given, to the strengths and weaknesses of the team, and to the resources at hand all need to be taken into consideration. If this is not done, then you get situations where costly adjustments will need to be made to the product. It's important to make goals using the SMART goals outline as it is a valuable tool that one can use.

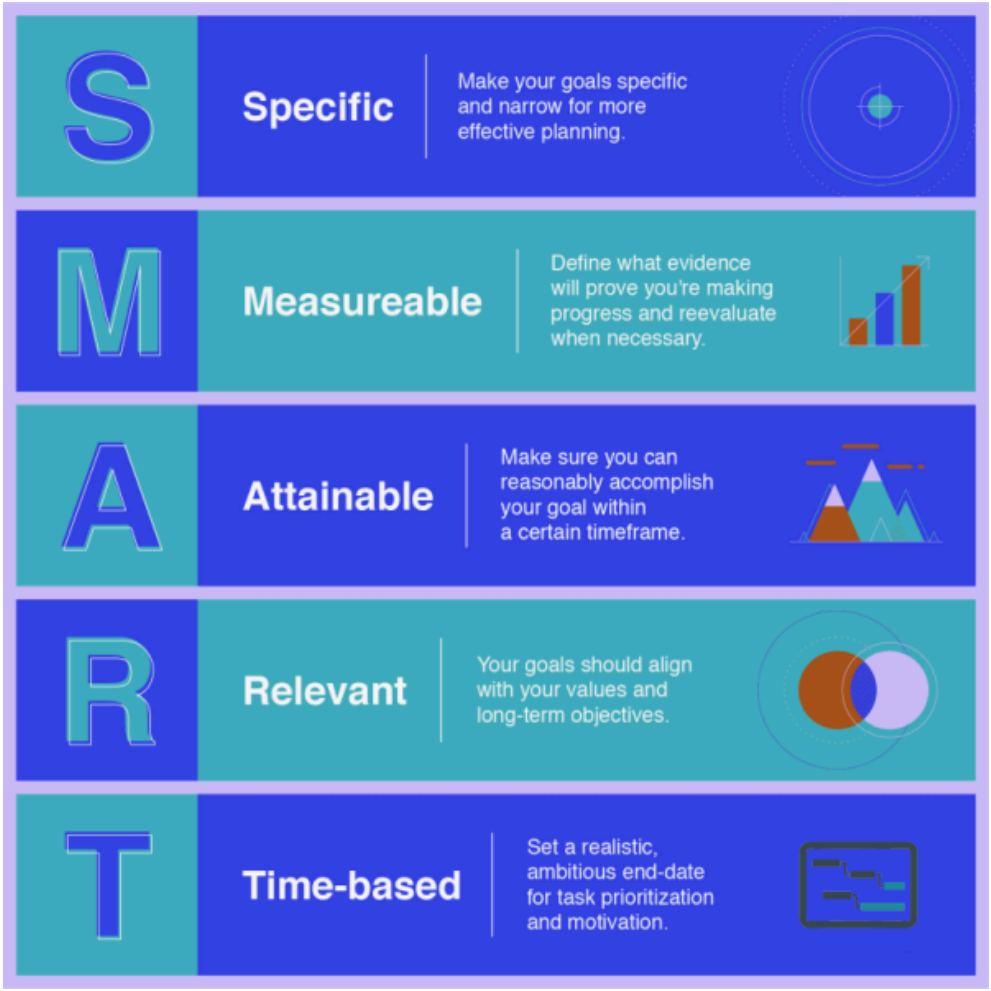


Figure 51: SMART Goals Planning Diagram

If our team had the hindsight available we would have put greater emphasis into making sure our goals are attainable by using tools like SMART. This is what our team would strongly suggest so that costly changes into phases further into the project do not need to be done.

5.3.2 Time Estimations

As a team we learned the lesson that we are all human. We make mistakes and plenty of them, and we needed to remember this was going to be a learning journey. For the majority of our team this project was our first experience of working under the Agile software development approach. This would in return mean that we need more time to finish tasks.

But because of our inexperience we often set sprint goals where we assume everything will go perfectly to plan, we all know that isnt always the case. We noticed that we wouldn't be able to finish some sprint goals and they would have to be extended, which understandably caused us to think we were behind as we already had next sprint goals on schedule.

If our team had the hindsight available to apply new work methods to our project we would have spent more time on our sprint meetings reflecting on our sprint goals and keeping an open mind to possible obstacles that could be encountered for each goal. This would allow us to have made more attainable time frames for each goal as we would have taken into consideration the possibilities of hiccups during the sprint. We would also have used the tools at our disposal that help with time management such as The Eisenhower Matrix that helps with decision making on tasks depending on their priority.

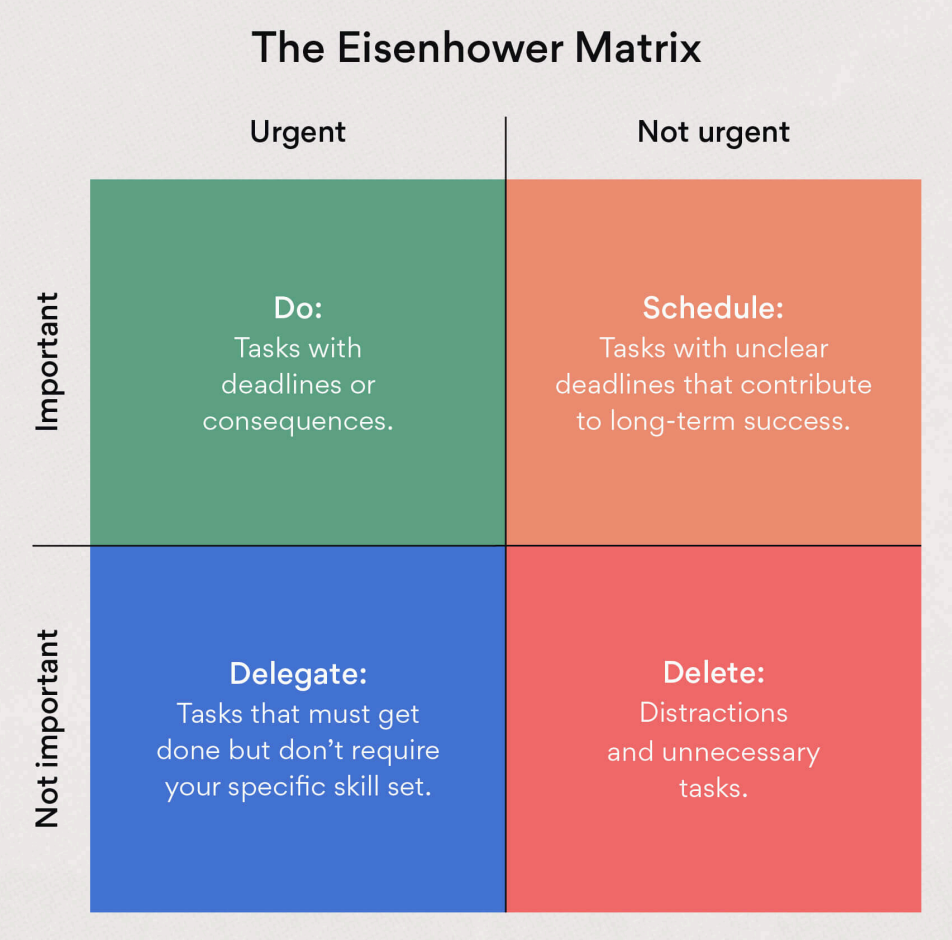


Figure 52: Eisenhower Matrix

These are the suggestions that our team would now strongly advise for any project.

5.4 Self-Evaluation

Criterion	Self-evaluation ranking	Justification
Contribute to a team in an appropriate and meaningful way	Exceeding	All of our members contributed equally and worked hard on the project
Apply an iterative process to refine or assign solutions for a given engineering design problem	Exceeding	We used a iterative process to refine our solution using agile methodology and and trade study analysis

Achieve a system design breakdown for management and implementation	Exceeding	We explained the system breakdown extensively in our baseline design section.
Justify the strength and limitations of the solution and make recommendation for possible improvements	Meeting	We justified the strengths of our product
Concise and coherent document that reflects critical analysis and synthesis	Exceeding	We make sure we are MLA formatting to keep our document professional and we provide the correct sequence for our headings.
Design has been reviewed for sustainability impact and potential for negative unintended consequences	Exceeding	We discussed how our product will meet the sustainability goals
Adjust project schedule based on project status	Exceeding	
Monitor Risks during the lifecycle of the Project	Meeting	This requirement was fulfilled as we evaluated our risks involved and how to mitigate them using the RISK matrix.
Applies all appropriate engineering concepts and fundamentals, theories and practices to solve the engineering problems	Exceeding	We applied many engineering concepts and tools to help us communicate ideas, such as UMLs, use cases diagrams, sequence and activity diagrams.

Table 3: Team self-evaluation.

5.5 Member Reflections

5.5.1 Joshua Amilhasan

5.5.1.1 Self-Reflection

The Capstone project was full of new experiences for me, which brought many new lessons and problems to overcome. I am glad our group decided to follow the Agile methodology for this year-long project as it has been something I wanted to dive deeper into and experience.

I believe that throughout the capstone project I was able to sharpen my behavioral skill set as well as the technical side by being engaged in both the development and communication side of things.

I realized the importance of planning throughout this course. There were situations at the beginning of the year where I felt pressured to start working on the product as soon as possible which came with its failures. There was the realization that we need a clear outline of our product so that a roadmap is laid out for the whole team to follow and not stray from.

5.5.1.2 Team-Reflection

Working in a large group as ours with a total of 8 members sure had its advantages and disadvantages. The great thing about a large group is that there are so many different skill sets that each person brings with them. Another benefit with our large group is that there were so many great ideas for the project that everyone shared in meetings.

This connects to what I believed was the biggest set back of the large group, is that it is difficult to have everyone on the same page and everyone understands exactly what the next steps and goals are in the project.

5.5.2 Jihal Patel

5.5.2.1 Self-Reflection

Reflecting upon myself as we stand at the end of the capstone, I can confidently say that, while I encountered multiple hiccups throughout the span of the project, I always put my best efforts into everything.

My peer feedback from ITP Metrics shows that the team recognized my efforts. I always valued the feedback on all my work as I myself set myself goals to improve my overall teamwork qualities.

I am studying as a Software Engineer and I was able to bring skills from my other classes to help the team in documenting and also implementing a few small features in the product.

5.5.2.2 Team-Reflection

Our group consisted of software and computer engineers. Even though our studies have been similar, everyone brought their own skills to the team and was able to help out in certain ways.

There were sprints where certain members of the team were leading the team as they had the most experience through those steps, while at other times I was able to lead a certain story point in a sprint.

The team dynamic was very positive and friendly between all the members. The team grew closer throughout the term as we got to know each other much more, especially when campus re-opened.

5.5.3 Lukas Rose

5.5.3.1 Self-Reflection

The capstone project was a complete rollercoaster for me. There were times of excitement from success that our team achieved, and just as many times stress from failures that I encountered. The journey was in every aspect a learning experience, and I am very glad I had the opportunity to undergo this year-long journey that resembles somewhat closely something I will encounter as an engineer in my career.

While my technical skills grew greatly from this project, it was not the area I improved on most. I strongly believe that my communicative skills were the greatest area of improvement. I entered the course as one that was always last to speak my ideas to the group, but with time I began to be more outstanding and at times even lead a conversation in a meeting, which I am proud of.

5.5.3.2 Team-Reflection

I believe the team had a great dynamic. We improved over the course length to be more open with another and more effectively communicated as a result. We got to know each other a lot better throughout the first term, and I tried to especially get to know members I did not know previous to this course. Our team has been able to handle problems fairly well whenever we come across them, this has improved the most from the beginning of this course.

I know that, as a group of 8, we worked well together. While issues were always bound to

happen, we were able to overcome them and learn from them. The team always had the vision of creating this education platform that would showcase our skills and hard work that went into it.

5.5.4 Mike Shen

5.5.4.1 Self-Reflection

When reflecting back on this capstone experience I can very appreciate all that this course has taught me. The whole year has been a journey as from the beginning I was keen to leave my mark on this project and be proud of my work.

I would have never imagined the amount of work it takes to keep up with all the sprints and not fall behind, but I can say now it was worth the hustle. I was able to help in different fields of the project but mostly in the front end department.

The peer feedback I received after every gate from my peers was very helpful in terms of improving my skills. I always set goals from the feedback received and it helped me improve as a team member and better collaborate with the team.

5.5.4.2 Team-Reflection

The team as a whole always had my back, the positivity we showed towards each other helped the team succeed. Of course, there was always something our team could've done better, but overall we did very well to constantly improve and encourage each other.

I found it refreshing, having so many like minded people in my group, as it made meetings very interesting to hear what everyone's visions were for our upcoming sprint. We were able to showcase our skills and it was not anything negative for one to not have experience in a certain field, we worked together to get the job done.

5.5.5 Phuong Tran

5.5.5.1 Self-Reflection

I was able to help the team in designing the visuals of the user interface of the platform in the beginning of implementations. I think this was major as it gave the team a structure to work from. Most recently I have been able to use my skills in video editing to help the team complete the videos that will be shown on the capstone day. I am overall happy with the positive impact I had on the team and work to show for it.

5.5.5.2 Team-Reflection

We all contributed fairly equally which was something I valued. Each member of the team had

been assigned tasks every sprint depending on their strengths and weaknesses.

The team spirit, or dynamic as one may say, was always encouraging and even when we ran into issues we almost always found a way to find a solution without pushing everything on one person. Overall I very much enjoyed working with the team and will take the experience into the beginning of my career.

5.5.6 Samyak Jain

5.5.6.1 Self-Reflection

The 2 semester journey of the capstone project has given me great insights into team work and the Agile methodology. A lot of problem solving was showcased and polished for me as I took the lead role of the group for the majority of the time. I showed leadership in overlooking all the work and making sure we are staying on schedule. The feedback from my peers has been positive as they noticed my leadership impacts. Overall, reflecting on myself I made the very most of this project and it has made me grow as an engineer.

5.5.6.2 Team-Reflection

The team was mixed with software and computer engineers which helped us a lot in making decisions in the beginning. We decided on a project that would challenge us to produce an educational platform that used artificial technology to heighten the user experience.

If I could change anything from this experience, it would be to have the whole course in class. This was not the case because of covid restrictions. It made getting to know one another not as easy and overall I wish to have seen each other in person more than we had done. We did the best with what we were given and our spirits always stayed high throughout the project. I am happy with my experience with this group.

5.5.7 Sharujan Rajakumar

5.5.7.1 Self-Reflection

The experience I had in this capstone project was filled with many firsts for me. I enjoyed learning and putting to use the Agile methodology as I believe it is something I will be using fairly often in my software engineering career.

While I also ran into trials of my own during the journey, I was able to make the most out of them and use it all as a learning experience.

I was able to work on the baseline design and Software Architecture as well as the Sustainability and Unintended Consequences for the initial stage and throughout further into the course was able to use my skills in UI design and database knowledge to help teammates.

5.5.7.2) Team-Reflection

Our team dynamic improved on every report. My team has encountered various roadblocks throughout this project but has always managed to overcome them, and more importantly, take the lessons learned from the struggles to improve our team.

At the end of the project I reflect back on my time with the team and see an experience of working together, leading, and plenty of problem solving and adjusting. The team interactions is what I learned from most during this capstone and will cherish the time I had.

5.5.8) Varuhn Ruthirakuhan

5.5.8.1) Self-Reflection

I improved myself so greatly in many different areas. When I look back, I can confidently say that my technical knowledge and skills have grown this year more than any of the previous years in my degree. That just shows how much I had to learn and try new tools throughout the course to get the best results for my team.

5.5.8.2) Team-Reflection

The team was very keen to start working on the development. This seemed to have caused some roadblocks as the team needed to back track and iron out a few details we had initially missed in the planning process.

These problems we encountered did not bring us down but instead we grew stronger and improved with every challenge we faced. The team used each other's strengths to make up for other weaknesses which worked out very well. I am glad to have been part of this team and take with me the experience and new skills.

6 Acknowledgment and Appreciation

Our team would like to acknowledge everyone that was involved and helped us throughout the Capstone journey, we appreciate everyone who had a role in our success.

Course Directors: For all your support since the beginning and throughout the project, giving us valuable suggestions and advice. We appreciate the mentors you all have been to our team.

James Smith: For taking time to meet with us and push us to be innovative. Your mentorship as a team supervisor throughout the beginning of the course set the direction our team went in and helped us set up for success in this journey. Thank you!

Course TAs: We appreciate all the feedback, considerations and support you provided us throughout the term!

Other Capstone Teams: We are thankful and glad to have had so many other great teams and engineers give us advice and pointers throughout every peer meeting in the capstone year. All your feedback was used to help our decision making and progress in our product.

7 Conclusion

To conclude, creating a web application that incorporates AI interactions has proven to be a very complex and difficult task. This is because the AI needed to be incorporated in a way that would help course directors deliver relevant content towards their students. Many current web applications do not involve the use of modern AI advancements that help on the education platform. This is why our team looked to have AI involved in as many ways that could help transfer course content in the easiest way possible.

After discussing as a group, our team was able to derive what kind of AI implementations we would need and how our web application should be constructed. In terms of the web application, the team analyzed many different education platforms online and looked to provide a new application that would better serve the teaching purpose. Looking into how course directors would like to interact with their students as well as how students collect and study course material helped with the derivation and construction of application.

When constructing the application, it was also very important to involve a strong frontend and backend that would support our systems needs. That is why our team chose to use node js (backend) and react js (frontend). With our backend the team also decided on using the MySQL database so different users could access their accounts once logging in. MySQL would store user credentials which made it easier to provide safety and security to each user’s private accounts.

Our team also focused on separating ourselves from other current web applications involved in education by incorporating newer AI technologies. Some AI involvement would be more complex while the others would be rather simple. For example, when attempting to include a speech-to-text functionality, the team found that the Google ML API would help enable this feature fairly easily. More complex AI implementations would also be used to send course directors feedback on student’s progress in the course so they are able to learn and help students in different ways.

Thanks to the constant help from our advisors, supervisor, student clients, and the Capstone admin team, our group was able to make significant development in the project past our initially set goals. Receiving constant feedback upon each release helped the team to see certain areas of improvement we could look to achieve throughout the course. Some students were also able to use our website and provide our team with tips on improvement when moving forward with the project. Finally, our group would like to thank our supervisor Professor James Smith for meeting with our group and helping us find originality within our application. With the help of the team we were able to create a web application past our goals primarily set in the course.

8 Appendix

The following requirements that we came up with during the project scope are:

Functional Requirements	Non-Functional Requirements
The system should allow the student/teacher to select courses.	The system should have a load response time of around 2-5 seconds. (performance)
The system should allow the teacher to add lessons to their specified courses.	The system should provide security protocols when the user registers/logins.
The system should allow the student/teacher to register with a valid email and password.	The system should be able to provide good user experience.
The system should allow the student/teacher to login to the web application.	The system should be maintainable. For instance, the ease in which the web application can be modified or extended.
The system should incorporate AI technology to provide course recommendations based on student feedback.	The system should be reliable. This means that the web application is always running on the cloud.
The system should be able to scan the barcode and if successful allow the user to access the quiz.	The system should be scalable. This means that the web application is capable of growing and increasing its features and functionality without impacting its overall performance.
The system should provide quizzes to test the user's knowledge.	The system should be recoverable. This means that the web application has the ability to recover from a crash or a failure in the system and return to full operations.
The system should provide a collaborative session like a comment section where students can provide feedback and teachers can view them.	The system should be portable, which means that there is less effort to move the software to a different target platform.

Table 4: Appendix.

9 Links

Trello Board link :

https://trello.com/invite/b/xDzbgmTe/051f679f1057166ebb2a05c3e4177701/web-application-eng-4k

Github Management Software:

https://github.com/samyakjain5559/ENG_4K_web_app

Risk Matrix Board:

https://app.conceptboard.com/board/y7x9-eae0-h3ob-dtsf-4ik6

AiM Web Application Demo:

https://www.youtube.com/watch?v=SIv7ASgyzus