**SMART PROJECT MANAGEMENT SYSTEM**

Osanda Malimabada Gamage

(IT18140330)

BSc (Hons) in Information Technology

Specializing in Software Engineering

Department of Software Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

October 2021

**SMART PROJECT MANAGEMENT SYSTEM**

Osanda Malimabada Gamage

(IT18140330)

Dissertation submitted in partial fulfillment of the requirements for the Bachelor of Science

Department of Software Engineering

Sri Lanka Institute of Information Technology

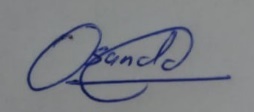
Sri Lanka

October 2021

# DECLARATION

I declare that this is my own work and this dissertation does not incorporate without acknowledgement any material previously submitted for Degree or Diploma in any other University or institute of higher learning and to the best my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to Sri Lanka Institute of Information Technology, the non-exclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part of future works (such as articles or books).

 2021/10/08

Signature: Date:

The above candidate has carried out research for the bachelor’s degree Dissertation under my supervision.

Signature of the supervisor: Date:

# ABSTRACT

Project Management Systems are the most important systems in the university time period. Most of the universities use different project managing the platform or systems which includes various functionalities. But when we consider about international industrial project management systems, their requirements don’t come with a common solution for managing projects. Because international project management systems are different by one to another. Therefor we decide to develop a common project management system which includes all facilities in existing systems and some special parts that are not include in existing systems.

The Smart Project Management System include Predict future contribution percentage possibility using the GitHub contribution part which is a one of the special part that we are going to include to our system. Using GitHub REST API [7], we generate a progress report using student group project repositories [5] and give them a grade. So the student can get an idea about their marks before getting viva or presentation. That is how are we going to predict future contribution possibility using GitHub. Notify students for incoming events and due date notifications, Performance report which represents individual contribution to the project by each team member and Issue tracking are the other sections of this research part.

The end of the solution, we provide a web based application to manage projects with highly customized functionalities. Application will be an open source solution for the universities.

**Keywords:** GitHub Prediction, GitHub API, Smart PMS, Algorithm

# DEDICATION

This research is dedicated to team members, supervisors and my parents. I will always appreciate all they have done. Project Management is quite simple with current technology and most of the time Project Manager has the technology to manage most of the management process automatically using existing solutions. This research is focusing on campus students that manage to undergraduate project management.

Predict future contributions with percentages and possibilities using GitHub contribution is one of the important part for universities. Because using a version control system is mandatory to doing current group projects. As such, it is expected to introduce the new Project Management System with many capabilities for undergraduate project management.

# ACKNOWLEDGEMENT

I would like to present my deep gratitude for the endless support and guidance helped by our supervisor Ms. Uthpala Samarakoon (Senior Lecturer - Faculty of Computing SLIIT) for her valuable and constructive suggestions during the planning and development of this research work. Her guidance and support helped us the way to get into competitions and show to our talents and complete this SMART Project Management System. Also I would like to give second gratitude to our co-supervisor Ms. Kugathasan Archchana (Lecturer - Faculty of Computing SLIIT) for her encouragement and much needed motivation. Without our supervisors’ major contribution, I would have never obtained this accomplishment.

Also I would like to thank Mr. Gayantha Udara (Project Manager - Jadon Software Solutions) for all the guidance given during my research.

Not forgetting all the contributors who participated by answering our questionnaires, filling google forms and testing the solution. Furthermore, we would like to thank our parents for their patience, time and providing resources to acquire the needful and for other expenses. Special thanks to my team mates who worked as a team to make this research more effective and success. Thank you for your time, encouragement, and recommendations.

**TABLE OF CONTENTS**

[DECLARATION iii](#_Toc84539946)

[ABSTRACT iv](#_Toc84539947)

[DEDICATION v](#_Toc84539948)

[ACKNOWLEDGEMENT vi](#_Toc84539949)

[LIST OF FIGURES viii](#_Toc84539950)

[LIST OF TABLES ix](#_Toc84539951)

[LIST OF APPENDICES ix](#_Toc84539952)

[LIST OF ABBREVIATIONS x](#_Toc84539953)

[1 INTRODUCTION 1](#_Toc84539954)

[1.1 Background & Literature survey 1](#_Toc84539955)

[1.2 Research Gap 6](#_Toc84539956)

[1.3 Research Problem 9](#_Toc84539957)

[1.4 Research Objectives 11](#_Toc84539958)

[1.4.1 Main Objectives 11](#_Toc84539959)

[1.4.2 Specific Objectives 11](#_Toc84539960)

[2 METHODOLOGY 13](#_Toc84539961)

[2.1 Architecture 13](#_Toc84539962)

[2.2 Dataset 19](#_Toc84539963)

[2.3 Research Area 19](#_Toc84539964)

[2.4 Requirement Gathering and Analysis 20](#_Toc84539965)

[2.5 Design 21](#_Toc84539966)

[2.6 Tools and Technologies 21](#_Toc84539967)

[Tools 21](#_Toc84539968)

[Server Side 21](#_Toc84539969)

[Client Side 21](#_Toc84539970)

[Database 21](#_Toc84539971)

[2.7 Commercialization aspects of the product 22](#_Toc84539972)

[2.8 Testing & Implementation 23](#_Toc84539973)

[2.8.1 Testing 23](#_Toc84539974)

[2.8.2 Implementation 26](#_Toc84539975)

[2.8.2.1 Web Application Development 26](#_Toc84539976)

[2.8.2.2 API Development 30](#_Toc84539977)

[2.8.2.3 Supervised Learning for Prediction Process 31](#_Toc84539978)

[2.9 Deployment 35](#_Toc84539979)

[3 RESULTS AND DISCUSSIONS 36](#_Toc84539980)

[3.1 Results 36](#_Toc84539981)

[3.2 Research Findings 37](#_Toc84539982)

[3.3 Discussion 37](#_Toc84539983)

[4. SUMMARY OF THIS RESEACH CONTRIBUTION 38](#_Toc84539984)

[5 CONCLUSION 38](#_Toc84539985)

[6 DESCRIPTION OF PERSONAL AND FACILITIES 40](#_Toc84539986)

[7 Reference List 41](#_Toc84539987)

[8 APPENDIX 42](#_Toc84539988)

# LIST OF FIGURES

[Figure 1: Percent increase in open source project creation per active user compared to previous year [8] 2](#_Toc84539159)

[Figure 2: Geographical distribution of active users 2020 [8] 3](#_Toc84539160)

[Figure 3: Top languages over the years [8] 4](#_Toc84539161)

[Figure 4: Architecture of the GHTorrent data-on-demand service [1] 8](#_Toc84539162)

[Figure 5: Feature Comparison with the existing solutions and proposed system 10](#_Toc84539163)

[Figure 6: Interaction among Stakeholders 14](#_Toc84539164)

[Figure 7: System overview 15](#_Toc84539165)

[Figure 8: Simple workflow diagram 17](#_Toc84539166)

[Figure 9: Prediction Process 18](#_Toc84539167)

[Figure 10: Project Overview by commits progress 18](#_Toc84539168)

[Figure 11: Create new Project Environment 26](#_Toc84539169)

[Figure 12: Instructions to create Personal Access Token (PAT) 27](#_Toc84539170)

[Figure 13: User's all projects 28](#_Toc84539171)

[Figure 14: Overview for single project 28](#_Toc84539172)

[Figure 15: Prediction for a single user 29](#_Toc84539173)

[Figure 16: save GitHub data endpoint 30](#_Toc84539174)

[Figure 17: Describe of data and a chart with previous vs this month 32](#_Toc84539175)

[Figure 18: sample data set 32](#_Toc84539176)

[Figure 19: Data cleaning 33](#_Toc84539177)

[Figure 20: Select a model and train it 33](#_Toc84539178)

[Figure 21: Fine Tuning the Model 34](#_Toc84539179)

[Figure 22: final result 34](#_Toc84539180)

[Figure 23: ngrok logo 35](#_Toc84539181)

# LIST OF TABLES

[Table 1: Test case 01 23](#_Toc84539548)

[Table 2: Test Case 02 23](#_Toc84539549)

[Table 3: Test Case 03 24](#_Toc84539550)

[Table 4: Test Case 04 24](#_Toc84539551)

[Table 5: Test Case 05 25](#_Toc84539552)

[Table 6: Test Case 06 25](#_Toc84539553)

[Table 7: Test Case 07 25](#_Toc84539554)

[Table 8: Prediction Overview for User 36](#_Toc84539555)

[Table 9: Description of personal and facilities 40](#_Toc84539556)

# LIST OF APPENDICES

[Appendix 1:Sample questionnaire – Collect the data 42](#_Toc84544014)

[Appendix 2: Sample questionnaire response 43](#_Toc84544015)

[Appendix 3: GitLab Repository details 44](#_Toc84544016)

[Appendix 4: Branches in the Repository 44](#_Toc84544017)

[Appendix 5: Programming languages used in this repository 45](#_Toc84544018)

[Appendix 6:First Contribution from my branch 45](#_Toc84544019)

[Appendix 7: Last Contribution from my branch 46](#_Toc84544020)

[Appendix 8: Commits from me 46](#_Toc84544021)

[Appendix 9:Graph chart in GitLab 47](#_Toc84544022)

[Appendix 10: Plagiarism Score 48](#_Toc84544023)

[Appendix 11: Plagiarism Score 02 49](#_Toc84544024)

[Appendix 12: Plagiarism Score 03 50](#_Toc84544025)

# LIST OF ABBREVIATIONS

**Abbreviation** **Description**

VCS Version Control System

VC Version Control

PMS Project Management System

API Application Programming Interface

REST API RESTful Application Programming Interface

PAT Personal access token

GRA GitHub REST API

URL Uniform Resource Locator

# 1 INTRODUCTION

## **1.1 Background & Literature survey**

Project Management is quite simple with current technology and most of the time Project Manager has the technology to manage most of the management process automatically using existing solutions. For the industry, existing project management tools are powerful enough to reduce the workload for the manager. However, when it comes to the Undergraduate Project Management, most of the management tools are more advanced and also some of the required features are not available with common management solutions. Therefore, we’ve decided to develop a Project Management System where we define our own features and also included with common features.

Version control systems known as VCSs are used to load and rebuild previous models of programming source code. In industry, almost all IT companies use version control systems to update their systems and carry out their projects and source codes. It is imperative to maintain secure source codes and manage individual contributions.

GitHub is a provider of Internet hosting for software development and version control using Git. This is an open source version control software platform that lets multiple people make separate changes to the same pages at the same time. Because this platform provides the real time collaboration, also allows multiple developers to work on a single project at the same time and reduces the risk of conflicting codes, and can help decrease production time. Because this platform reduces the Integration process.

When we come to Universities, students usually do not use any version control system in their undergraduate studies, and it is a bad practice to avoid version controlling since the project is a group project. The problem is students are not familiar with version control. The use of version control systems can solve another problem of underestimating project duration, which can cause incomplete projects within the specified time. So, incorporating version controlling facility in Undergraduate project management is another important aspect of project management.

As I mentioned above, GitHub is an open-source project that enables developers to collaborate and share their expertise. It is also a place where social coding features and pull requests were born. Writing code can be easy. But as a team, you have to integrate with others’ code. In order to become a gentry programmer, you need version control support and also the support of others who code. That is why you use a version control system.

GitHub is a great place to conduct research. Its robust repository data can be used to study various facets of the platform. GitHub offers a tremendous research potential. For instance, it is a flagship for current open source development, a place for developers to showcase their expertise to peers or potential recruiters, and the platform where social coding features or pull requests emerged. However, GitHub data is, to date, largely un-derexplored. To facilitate studies of GitHub [1]

GitHub is the world's largest code host. Over 50M developers collaborate on the platform. Numerous popular open-source projects have migrated their code base to GitHub. There are 56M total developers on GitHub. There 60M+ new repositories created in the 2020 and there are 1.9B contributions added in the last year. [8]

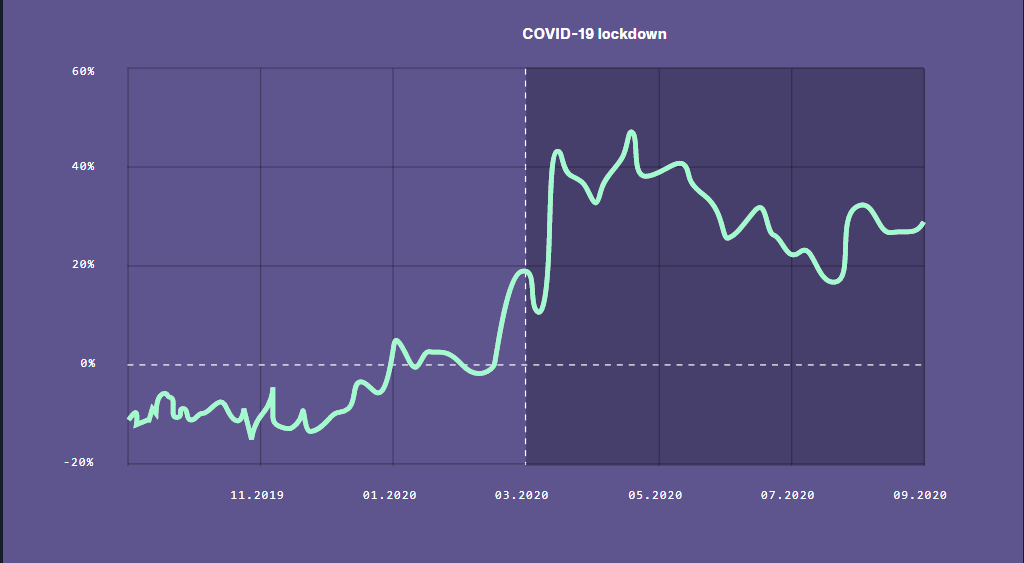


Figure 1: Percent increase in open source project creation per active user compared to previous year [8]

Figure 1 represents the Percent increase in open source project creation per active user compared to the previous year 2020. As you can see, when starting the COVID-19 pandemic, open source project creation per active user became high. That’s because Governmental and Private companies’ responses to COVID-19 had a clear effect on working from home across all regions and while developers have been putting in longer working hours, they’ve also spent more time on open source projects [8].

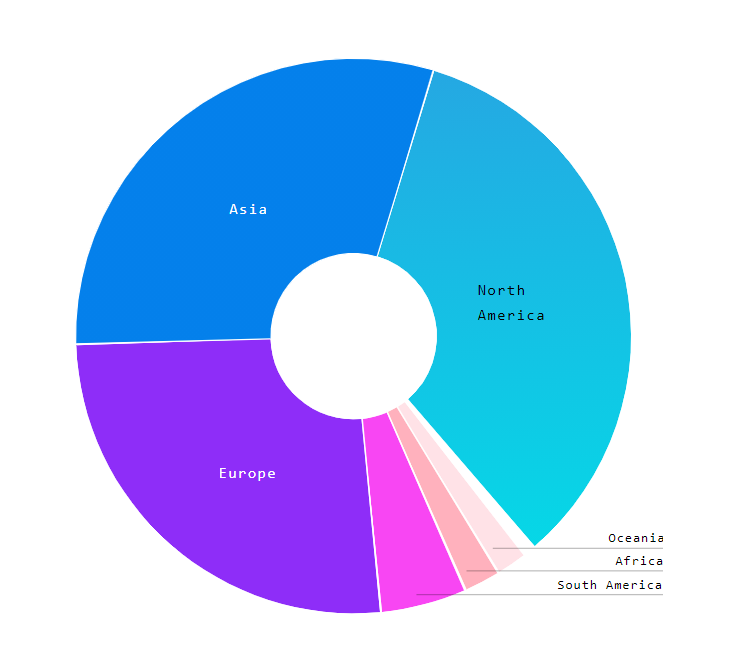


Figure 2: Geographical distribution of active users 2020 [8]

Figure 2 represents the Geographical distribution of active users in 2020. This is the regions that scaled in 2020 due to COVID-19 pandemic. In a time when everyone is forced to stay home, developers around the world have found connection and community through open source. While many caution that technology can be tiring, these patterns show that open source may provide a creative outlet that’s noticeably different from the workplace [8]

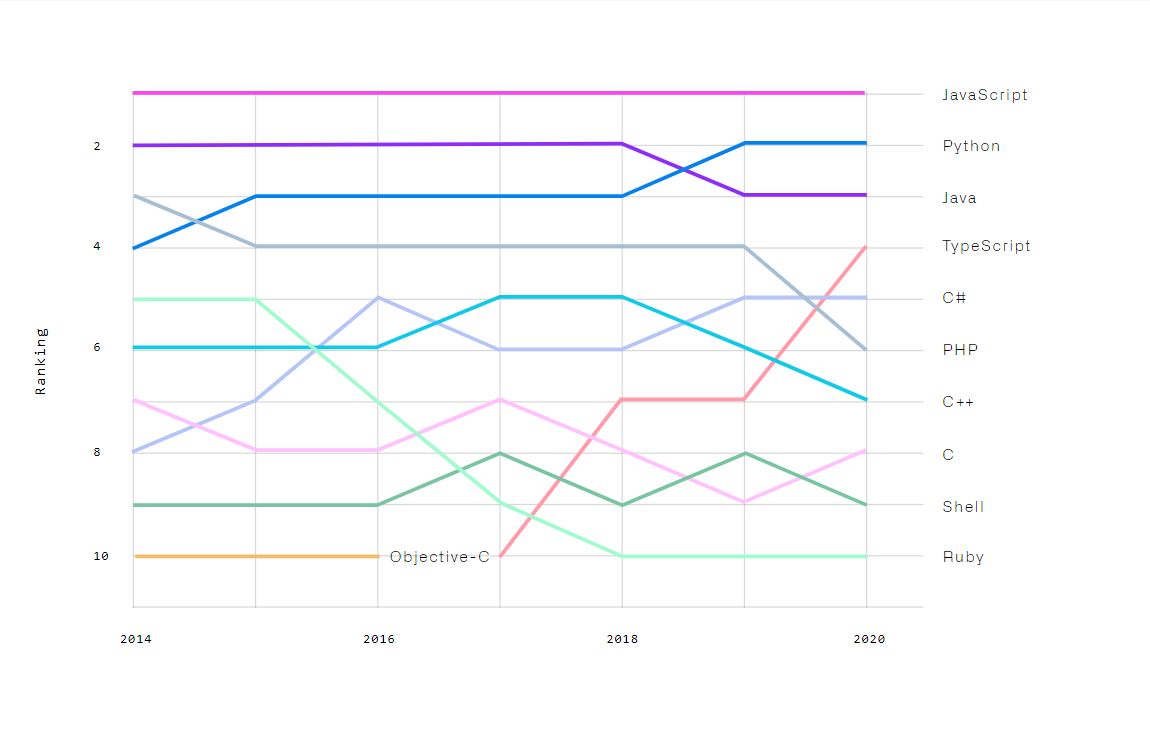


Figure 3: Top languages over the years [8]

When it comes to this research, we are focusing on undergraduate students to practice VC. Because in their group projects, they need to integrate their work as a team. That is why we called as a group project.

Predict future contributions with percentages and possibilities using GitHub contribution is one of the important part for universities. Because using a version control system is mandatory to doing current group projects. However, there are several version control systems. But GitHub is the most popular version control system in project management part. Most of the students in universities are using GitHub.

Our goal is to develop a system, to predict future contribution with percentage and possibilities using GitHub contribution. There are some systems to predict popularity of repositories [2], predict ecosystem health in GitHub [3], predict issues lifetime in GitHub projects [4]. But there isn’t a system to predict future contribution. Our goal is to predict student’s future contribution in a group projects.

To achieve this goal, we planned to use GitHub REST API [11] to manage student’s group projects. GitHub REST API is the API that you can use to interact with GitHub data. They allow you to create and manage repositories, branches, issues, pull requests, and many other options. We can collect data and analyze the GitHub repositories [5] and Users [6] from the GRA. You can access GitHub data using only GitHub PAT. As you can see, this is a personal access token and you can access All your GitHub data and if give your PAT to someone, they can also have access to your GitHub data.

We are going to track the code, commits, user contribution, task completion using GRA and we planned to contribute progress tracking using volume of code, overall commits, user contribution, quality of the code and task completion. There will be charts for the commits overview and other progress evaluation sections.

However, there is a research found related to GRA and they created a system called GHTorrent, a scalable, queriable, offline mirror of the data offered through the GitHub REST API. They present a novel feature of GHTorrent designed to offer customizable data dumps on demand. The new GHTorrent data-on-demand service offers users the possibility to request via a web form up-to-date GHTorrent data dumps for any collection of GitHub repositories [1]

When we consider about the process to predict future contribution in this research, Students adding their project GitHub URL to the system, supervisors and team members can get a rough idea about the project progress and speed of their project work. We are introducing this feature to identify each student do their work by themselves. Not the same person doing the entire group project. That will track using commit progress. There is a doubt. Yes, we have GitHub and we can use that to predict this problem. But in our system, we create a group first. Then assign a lecturer or a supervisor. After that supervisor will assign task to complete the project or give them a Topic and the team leader has to create a repository in GitHub and add the GitHub repository link to the system. So the system will predict the progress and generate the progress evaluation.

## **1.2 Research Gap**

There are several version control systems. But GitHub is the most popular version control system in the project management section. Most of the students in universities are using GitHub. In existing research to predict the Popularity of GitHub Repositories. In the GitHub platform, users can show appreciation to projects by adding stars to them. Therefore, the number of stars of a repository is a direct measure of its popularity. In that research, they have used multiple linear regressions to predict the number of stars of GitHub repositories. These predictions are useful to both repository owners and clients who usually want to know how their projects are performing in a competitive open source development market [2]. In that research, they show that the proposed models start to provide accurate predictions after being trained with the number of stars received in the last six months.

Another research found on Using Dynamic and Contextual Features to Predict Issue Lifetime in GitHub Projects, the lifetime of an issue depends not only on characteristics of the issue itself, but also on the state of the project as a whole. Hence, issue lifetime prediction may benefit from taking into account features capturing the issue's context [4]. In this research, they analyze issues from more than 4000 GitHub projects and build models to predict, at different points in an issue’s lifetime, whether or not the issue will close within a given calendric period, by combining static, dynamic and contextual features.

Another research represents “Healthy or Not: A Way to Predict Ecosystem Health in GitHub”. They found the development of the open source community, through the interaction of developers, the collaborative development of software, and the sharing of software tools, the formation of the open source software ecosystem has matured. Development of open source community, through the interaction of developers, the collaborative development of software, and the sharing of software tools, the formation of the open source software ecosystem has matured. They take GitHub as an example to analyze the health condition of open source ecosystem and, also, it is a research area in Symmetry. Firstly, the paper presents the healthy definition of GitHub open source ecosystem health and, then, according to the main components of natural ecosystem health, the paper proposes health indicators and health indicators evaluation method. Based on the above, the GitHub ecosystem health prediction method is proposed [3].

According to the researches mentioned above, we realize that we need a system component to Predict Future GitHub Contribution, which can be used to improve Project Quality. Therefore, predict future contributions with percentages and possibilities using GitHub contribution is one of the essential parts for universities. Because using a version control system is mandatory to doing current group projects.

Regarding to above mentioned research solutions, we found a way to develop “Predict Future Contribution on GitHub” part using GitHub REST API. The problem is to configure the volume of code, quality of code and user contribution to the project. For those first two parts, we have to develop another part to calculate the volume and the quality. There are some existing systems to measure and analysis both volume and the quality of the code [9]. So we don’t need to add new features or modify current options. We can directly use that technology. We need this part to generate a grade to the final result.

Using from the GRA, we can track overall commits for each student, the target is to use GRA and configure future prediction. Since all above mentioned existing systems are made for a single task.Some of the solutions are complex to be develop, and some them are not enough to configure and satisfy our requirements [2][3][4]. According to the existing system, there are two loosely coupled parts. A web server that handles data requests from users and the GHTorrent server that performs the data extraction. The two servers communicate via messaging queues [1].

In GHTorrent research, they clearly introduce how to use GRA and how they use GRA to their system. The architecture of the new GHTorrent data-on-demand service consists of two loosely coupled parts: a web server that handles data requests from users and the GHTorrent server that performs the data extraction. The two servers communicate via messaging queues. In the below figure 4 represents their Architecture of the GHTorrent data-on-demand service.

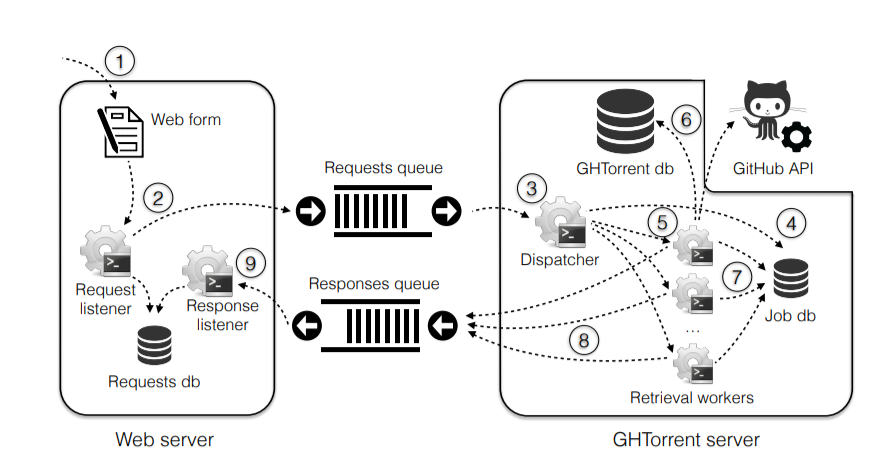


Figure 4: Architecture of the GHTorrent data-on-demand service [1]

As you can see their Architecture. They use GitHub REST API to data dumps for any collection of GitHub repositories. In our system, we planned to develop our system to predict future contribution, that means, we will get the repository data from the GitHub REST API. And then we will give students about progress regarding to overall commit progress, the user contribution to the project, the quality of code progress and a grade regarding to overall the project version control contribution to get an idea about their project overall evaluation before getting the actual marks. It means from the above mention commit progress, user contribution, quality of the code used to generate a grade. So the student can get an idea about the future contribution related to the group project. It means, they did the group project well or not before get the final result from the lecturer. That is the research gap from existing systems and the what I am going to develop.

# 

## **1.3 Research Problem**

Undergraduate project management is challenging when it comes to group projects. Even though there are many existing project management systems such as the Microsoft project, Jira, and Redmine, most of them were developed for general purpose. Hence, some important specific features which are useful when managing student projects such as automatic group formation, project tracking and notification generation on project progress are not available in those systems. Even though there are many project management systems available for general purpose, many of them are not very useful when it comes to group projects. This is because many of them were built for specific purpose only.

One of the major problem was tracking project progress. There is no project tracking system and it leads to project failures. In the other hand, students usually do not use any version controlling and it is a bad practice to avoid version controlling since the project is a group project. As I mentioned in the Background Study, VCSs are used to store and reconstruct past versions of program source code. In industry almost all the IT companies use version controlling systems to update their systems and also to carry out their projects and source codes. It is very important to maintain secure source codes and manage individual contributions. When we come to the progress tracking we have to use GRA to take the GitHub data from the student GitHub repository.

Problem is students are not familiar with version controlling. As I mentioned in the Background study, use of version controlling systems can solve another problem of underestimating project duration which can cause incomplete projects within the specified time. So, incorporating version controlling facility in project management is another important aspect in project management. As a team, students must need to know version controlling. When they get into the industry, the industry accepts the knowledge of version controlling. At least the students need to know the GitHub platform. So our goal is to practice GitHub using this research.

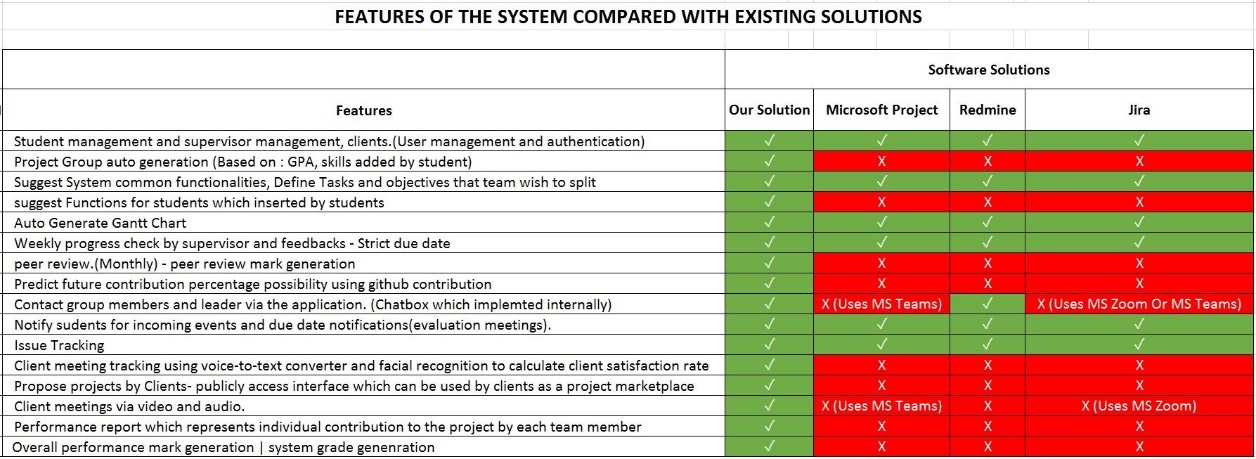
****

Figure 5: Feature Comparison with the existing solutions and proposed system

Figure 5 shows a table that we created by finding functionalities in the existing project management systems. The red colored cells show the functionalities are not implemented in the system and green colored cells show the functionalities are implemented in the system. As you can see the Predict future contribution percentage possibility using GitHub contribution and progress tracking are not implemented in neither software solutions, and it is a major requirement for any university to improve effectiveness and efficiency of university projects.

In this research, we introduce progress tracking module to track the student group project progress by fetching GitHub data and we predict the future contribution as a percentage possibility using GitHub commits contributions.

## **1.4 Research Objectives**

### **1.4.1 Main Objectives**

The main objective of this research is to Predict Future Contribution percentage possibility using GitHub contribution. Using GRA, can track the user code from the repository and generate progress related to key frames (Overall commits, user engagements, the quality of the code, etc.). Application is ready to get GitHub repository URL and PAT to identify each student group project. A supervised Learning Algorithm can use to predict any kind of dataset.

### **1.4.2 Specific Objectives**

Track the project progress using GitHub REST API.

* The students can track their work and the lecturer can check each of the students are working on the project. Using GitHub contribution percentage, the system will predict the future contribution to the project. This process will be applied to each student. This uses commits, project tasks, and timelines created at the start to track the project progress, and using extracted data, the system will generate a prediction about future contributions by each student. Using this facility system can identify if there is a risk of leading to an incomplete project or not.

Issue Tracking similar as stack overflow

* Different groups have different skilled students. Some of the students may have not known enough to solve an issue that might risk of stuck in workload. This solution is created to get inside help in the project. The team member can post issues that he came up with and team members can help to solve the issue. This solution is implemented to improve team working skills.

Notify students for incoming events and due date notifications.

* This facility is a notification system to inform due dates of current tasks and events. Such as evaluations and group assignments that interact with the project. Most university students having this issue. Most of the time they do not check course web and they miss some of the important due date. This system is implemented to reduce that risk.

Performance report which represents individual contribution to the project by each team member

* This facility is to generate an individual mark for their overall performance. This will be helpful to show as evidence in interviews. Experience is the key to everything, and most of the time students fail to show that because there are no visible evidences. Only repositories of their projects. This marking system also can be used to evaluate individual student performance in final evaluation meetings.

# 2 METHODOLOGY

## **2.1 Architecture**

Our overall research is a PMS. There are four users in this system. Student and the lecturer are the main users and others are admin and clients. In this part there are two users. Students and Lecturer.

There is an auto group generation process working with this research and after creating groups (this is the main part of the overall research) Lecturer will be assign to the relevant groups. Each group has a group environment. Group Environment means, this likes student, group project portal.

In this research, after generating the student groups, the leader can add the supervisor and the co-supervisor to their portal (group). Then supervisor assign a topic to the team and the leader can divide their work as tasks. After that leader has to create a GitHub repository and the PAT. Then leader need to create a project environment in our system. After that system will automatically track the project progress. So the supervisors can track students work.

This is a web application and the components of the application are handle by the system. Specially, the algorithm and models.

Mainly we can identify 2 sections in the system architecture

* Interaction Among Stakeholders
* System Overview

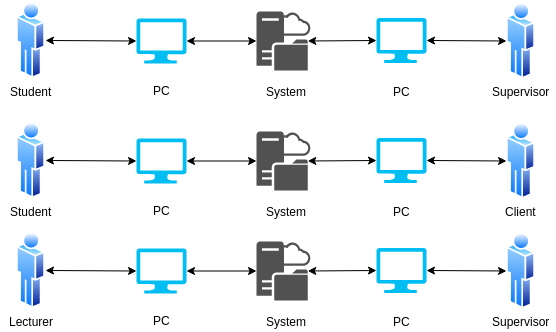


Figure 6: Interaction among Stakeholders

Figure 6 represents Interaction among the stockholders in our system. As I mentioned above, Student and the lecturer is main users. So they can interact any time through the system. Lecture can tract student progress, student can request comments for their tasks, Lecture can see student predicting for future contribution and lecturer can send notification when a student, weak and slow at their contribution are the interaction process in this system.

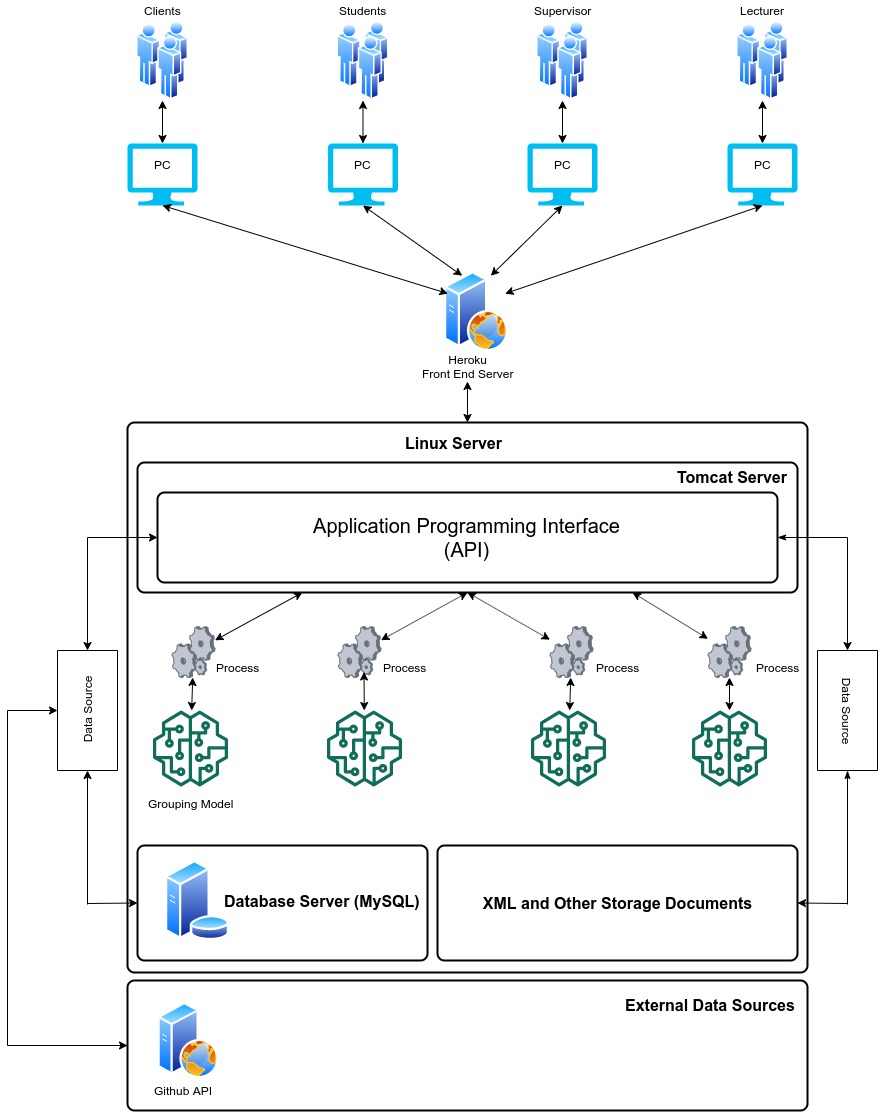


Figure 7: System overview

According to the system diagram as shown in Figure 7, We’re hosting the backend server in a Linux environment and inside the Linux server there are 4 components.

* API running on Apache Tomcat Server
* MySQL Server
* Trained Models for decision Makings
* XML and Other Storage Documents

Heroku Server used to deploy the Frontend solution to interact with users. The Frontend solution will be communicating with the users and the API.

As I mentioned above, the team leader needs to create a Repository in GitHub platform and create a project environment in this proposed system with the GitHub repository details. such as project name, repository owner name, repository URL, and the PAT. Only the team leader needs to create the access token, and other team members can view project integration details. Then a lecturer or a supervisor can be assigned and that person will assign tasks to complete the project.

In the below figure 8 represents the simple workflow diagram that has related to this research. There are two logins. One is for students and another is for the lecture/supervisor. Lecturer or supervisor can assign topic to the student group. After that students can continue their works by creating group project environment and keep pushing changes to their GitHub repository. The lecturer or supervisor can track students’ work using our system. When meet the deadline or finished student’s group project, supervisor can see the progress.

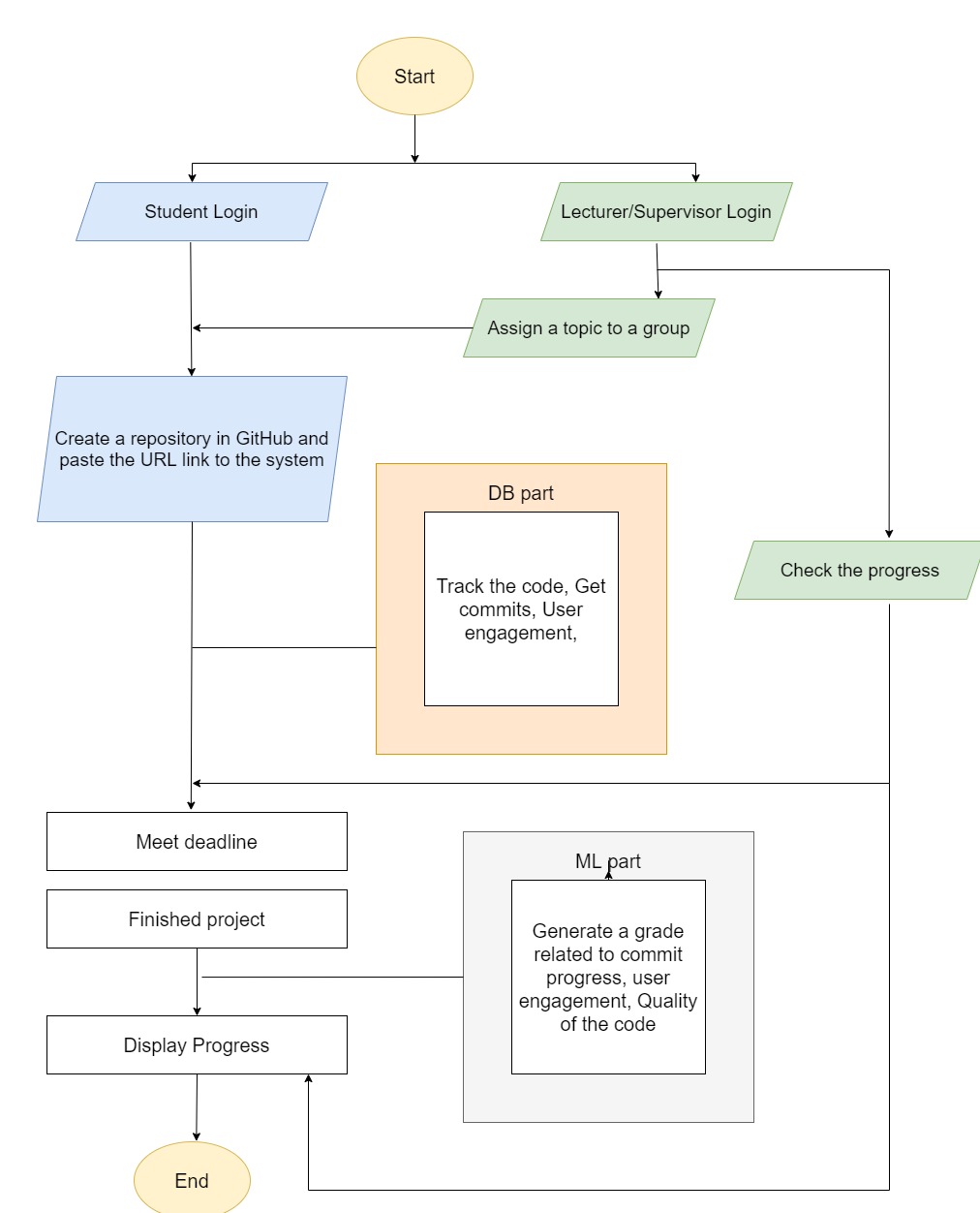


Figure 8: Simple workflow diagram

The system will predict future contributions with percentages and possibilities using GitHub contributions. This research use GRA to collect data and analyze GitHub Repositories. The user’s API can track the code, commits, User contribution, task completion for the relevant project. Next week progress will be predicted by previous two weeks’ commit progress using Supervised Learning method.

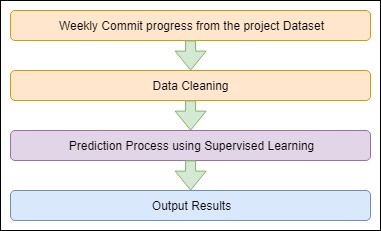


Figure 9: Prediction Process

Figure 9 represents the Prediction process of this research. We are collecting the commits every week and predict next week with the last two weeks. After collecting the commits, data will be cleaning. Then the Prediction process will start using supervised learning.

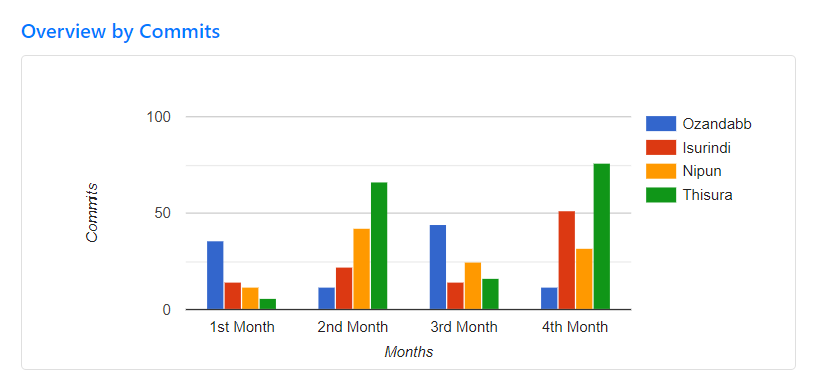


Figure 10: Project Overview by commits progress

Week by week, the system will track the commits progress for each student and will display the graphs and charts as a project overview. Supervisor and the team members can get a sketchy idea about the project progress and the speed of their project work. This feature helps to identify social loafers or free riders and ensure that a single person is not involving in the entire project work. Commit progress will be used to track this feature.

## **2.2 Dataset**

In this research, we use GitHub commits as the dataset. We don’t need an existing dataset to this research. Because we are going to predict future contribution using each member’s contribution to the project. We don’t need other projects’ commits, need only each project each member’s contribution. Using each member’s commits progress we predict the next week contribution for that member using last two weeks commits contribution.

## **2.3 Research Area**

Research part of this part is Predict Future Contribution of the student group project using GitHub REST API. Using that API, we track each student code, commits, user contribution, the quality of the code, enragements. Finally generate progress. So as to configure this research part, I have need to follow GitHub REST API documentation and the Supervised Learning algorithm in Machine Learning technology.

Supervised learning, also known as supervised machine learning, is the machine learning task (subcategory of machine learning and artificial intelligence) of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labeled training data consisting of a set of training examples.

The system will predict future contributions with percentages and possibilities using GitHub contribution. So we are going to use GitHub API to collect data and analysis GitHub Repositories.and user’sAPI can track the code, commits, User contribution, task completion for the relevant project. All we have to do is, pass the project Repository URL with the PAT. So supervisors and team members can get a rough idea about the project progress and the speed of their project work. We are introducing this feature to identify each student do their work by themselves. Not the same person doing the entire project. That will track using commit progress.

There is a doubt. Yes, we have GitHub and we can use that to predict this problem. But in our system, we create a group first. Then assign a lecturer or a supervisor. After that supervisor will assign task to complete the project and the team leader has to create a repository in GitHub and add the GitHub repository link to the system with the PAT. So the system will predict progress and generate the progress evaluation. Our system will do the entire solution. That is the solution proposed to track the project part. Finally, it will generate a mark using GitHub progress and it will get implemented on system grade for the project.

## **2.4 Requirement Gathering and Analysis**

This part is the most important part in my research part. Because I had to gather requirements and analysis them to develop my part. Especially I must concern about the gathered information because it should be important to my research part. Before starting to development, requirement gathering is most important to give the right solutions.

* Read research papers related to my research part.
* Referred GitHub Documentation to analysis how to configure GitHub REST API
* Referred Survey Results

## **2.5 Design**

After I gathered requirements, I analyzed those requirements to create a design. Design helps me to specify what are the hardware, software and system requirements to identify the architecture. So I can manage the tools and technologies that I am going to use to develop my research part. The architecture depends on my design and the software that I am going to use.

## **2.6 Tools and Technologies**

### **Tools**

* Visual Studio Code
* Apache Server
* Eclipse IDE
* Maven
* Postman
* Ngrok

### **Server Side**

* Java – JAX-RS Rest API
* Python
* Redux

### **Client Side**

* HTML, CSS, JS (jQuery, React JS)

### **Database**

* MySQL

## **2.7 Commercialization aspects of the product**

This research provides a wide variety of Project Management System with an easy and efficient way to access information and provide information. As I mentioned in the introduction, when we come to Universities, students usually do not use any version control system in their undergraduate studies, and it is a bad practice to avoid version controlling since the project is a group project. The problem is students are not familiar with version control. The use of version control systems can solve another problem of underestimating project duration,

When we finished this research, we can apply for the universities to use this with their group projects. So the students will practice for version controlling by using this system.

This application would involve a different kind of users in categories such as,

* Student
* Lecturer
* Client
* Supervisor

So we can move into the IT industry. Because we can handle clients using this research.

We can give this as free trail. So we can sell this product after trail if the user wants this. There is another way to commercialization of this product. We can provide limited function or feature for the user. Then the user wants more advanced functions, the user has to pay or upgrade their account.

## **2.8 Testing & Implementation**

### **2.8.1 Testing**

In the software life circle, Testing is a most important part. Because of according to Guru99 testing is a method check whether the actual software product matches expected requirements and to ensure that software product is defect free. There are two parts that we consider in testing part. That is Functional and non-functional.

Testing will be begun from the beginning of the system development and will be proceeded until the last endpoint of the system development. All these testing will be done so as to check whether the sections have been completed related to client requirements. For the functional testing we are going to use Sanity and User Acceptance testing. Testing will be proceeded until all the parts are tested completely. For the non-functional testing we are going to use all the non-functional testing types. That will be done to verify the development will have finished related to client requirements.

Table 1: Test case 01

|  |  |
| --- | --- |
| Test case ID | 1 |
| Test Case description | Verify if a user will be able to login with a valid username and valid password. |
| Pre- condition | Get the URL and connect the System |
| Test procedure | Get the URL and connect the System and Input the data |
| Test Input Data | Email and Password |
| Expected Output | User can logging the system successfully |
| Actual Output | User can logging the system successfully |
| Result | Pass |

Table 2: Test Case 02

|  |  |
| --- | --- |
| Test case ID | 2 |
| Test Case description | Verify if a user cannot login with an invalid username and an invalid password |
| Pre- condition | Get the URL and connect the System |
| Test procedure | Input the invalid data |
| Test Input Data | Invalid user name or Invalid password |
| Expected Output | System should display the error message Can’t logging the system |
| Actual Output | System should display the error message Can’t logging the system |
| Result | Pass |

Table 3: Test Case 03

|  |  |
| --- | --- |
| Test case ID | 3 |
| Test Case description | Verify if the user can create a new project environment. |
| Pre- condition | Go to add new project page using student profile sidebar |
| Test procedure | Input data to create new project |
| Test Input Data | Fill all the data |
| Expected Output | User create a project and system will display a toast message as a success message |
| Actual Output | System should display the success message |
| Result | Pass |

Table 4: Test Case 04

|  |  |
| --- | --- |
| Test case ID | 4 |
| Test Case description | Verify if the user tries to create a new project without required fields. |
| Pre- condition | Login into the system as a student |
| Test procedure | Miss GitHub filling fields. |
| Test Input Data | Fill only basic details. Not the GitHub data fields. |
| Expected Output | System should display the error message |
| Actual Output | System should display the error message |
| Result | Pass |

Table 5: Test Case 05

|  |  |
| --- | --- |
| Test case ID | 5 |
| Test Case description | Get all Project that related to logged in user. |
| Pre- condition | Login into the system as a student |
| Test procedure | Go to all projects page |
| Test Input Data | Pass the user ID |
| Expected Output | Get all Projects for the logged in user. |
| Actual Output | System will display all the projects inside the all project screen. |
| Result | Pass |

Table 6: Test Case 06

|  |  |
| --- | --- |
| Test case ID | 6 |
| Test Case description | Get a Single Project |
| Pre- condition | Login into the system as a student |
| Test procedure | Click one project in the all projects screen |
| Test Input Data | Pass the project ID |
| Expected Output | Get single project’s all data. |
| Actual Output | Single project screen will display with the project data. |
| Result | Pass |

Table 7: Test Case 07

|  |  |
| --- | --- |
| Test case ID | 7 |
| Test Case description | Display prediction process in the single project page |
| Pre- condition | Login into the system as a student |
| Test procedure | Click prediction tab that is inside the single project page. |
| Test Input Data | Pass commits per a collaboration |
| Expected Output | Prediction table will display with actual commits and the prediction commits for next week |
| Actual Output | Prediction table |
| Result | Pass |

### **2.8.2 Implementation**

This section start using my design, it will be converting the design into a working program. Implementation stage divides the part into several sections. Such as assigning Task or a Topic (That will be doing by the lecturer or supervisor. After created student groups, there is a lecturer will assign to the relevant group. Then the lecture will assign a task or a topic), Team leader has to publish the GitHub project URL to our system, get each contributions credentials with the PAT (API needs repository contributions’ credentials to configure), checking the code, commits and all other engagements related to each repository, analysis the evaluation of the repository (Commit charts, engagement charts, user engagements, some results related to code complexity and the quality of the code), Generating a mark.

In shortly, we can consider about following parts in the Implementation process.

* Web Application Development
* API Development
* Supervised Learning for Prediction Process

### **2.8.2.1 Web Application Development**

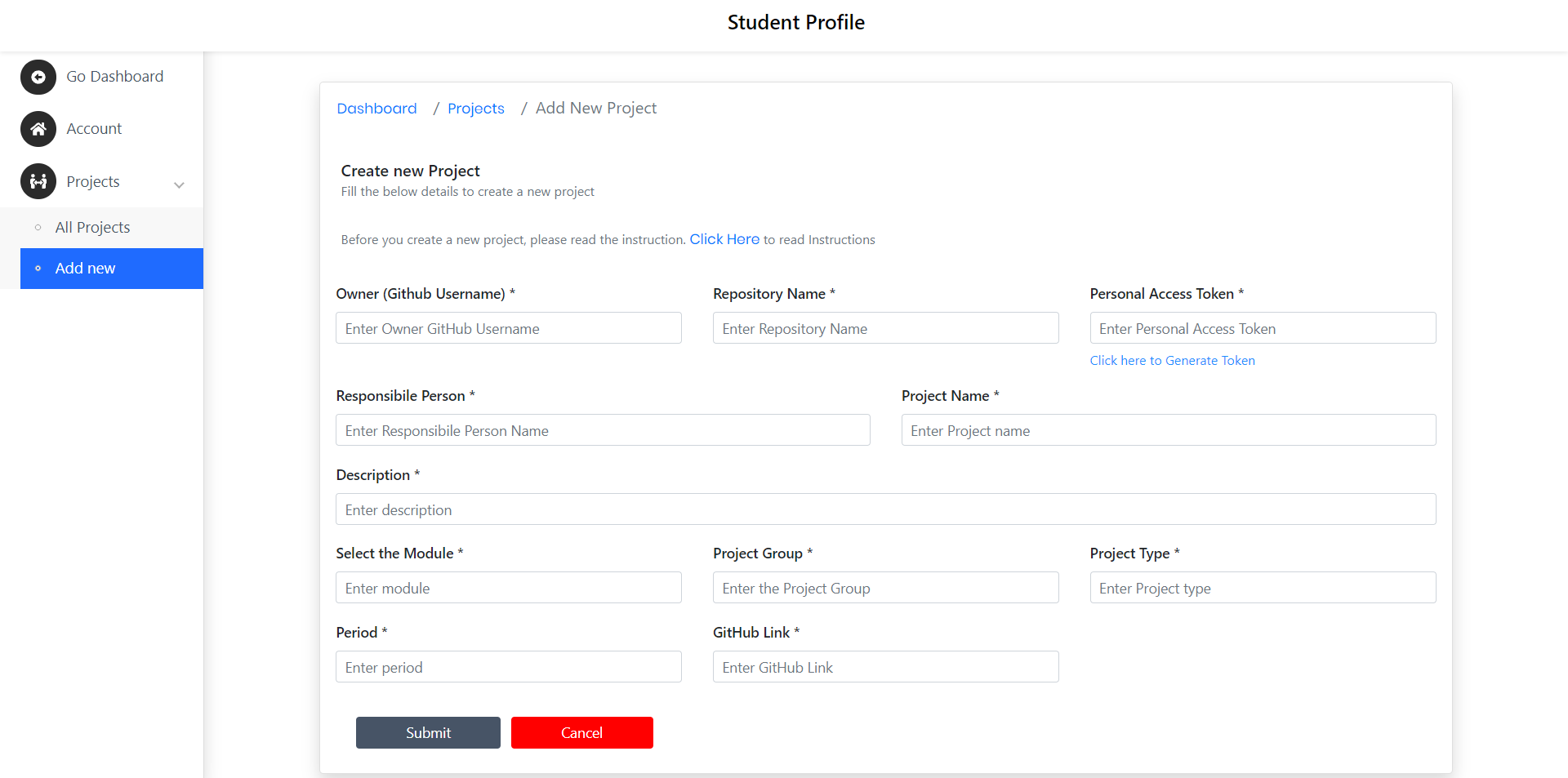


Figure 11: Create new Project Environment

Figure 11 represent create a new project environment interface implemented in the system. After creating GitHub repository, team leader need to create project environment in this system. As you can see, there are several fields to enter for create a new project. In here, we require owner name (owner GitHub name), Repository name (owner who created a repository in GitHub) and the Personal Access Token (PAT). Also we are looking for GitHub repository URL for further details.

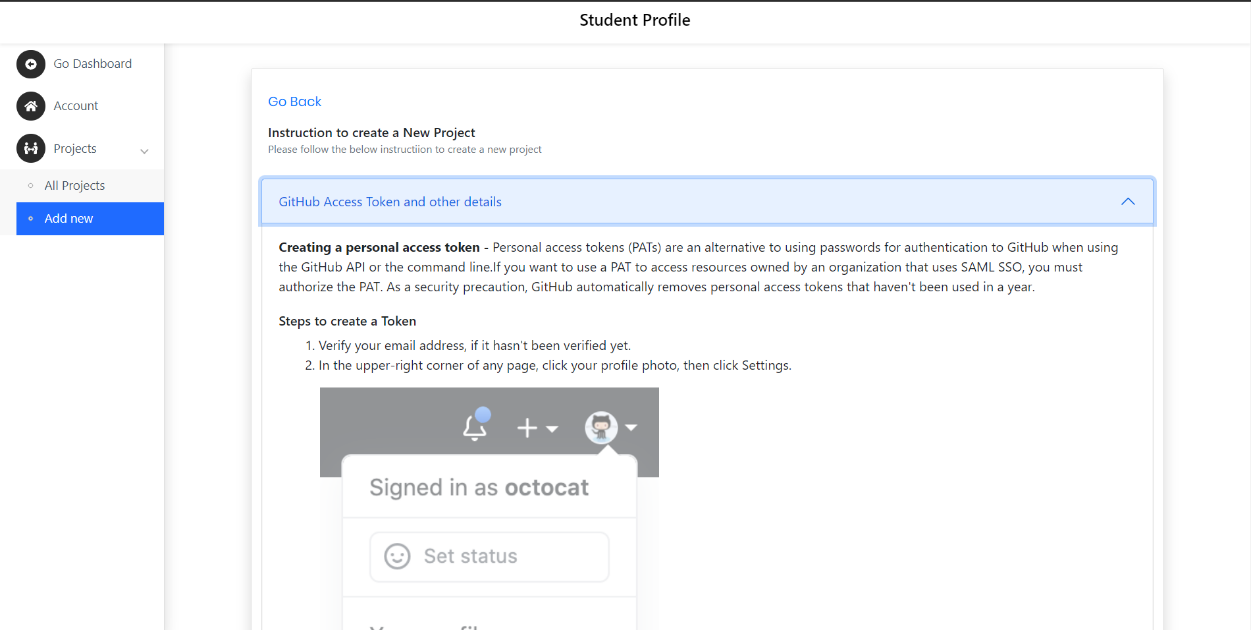


Figure 12: Instructions to create Personal Access Token (PAT)

Figure 12 represents an interface which includes how to create GitHub PAT as an instruction page. In this page user can learn how to create a PAT step by step. In figure 11, there is a text field to enter PAT and if the user doesn’t know how to create that, the user can click the toolip called “click here to generate a Token”. After that this interface will display.

Students can create/add a new project at any time. But in a group project, only the team leader needs to create/add a new project. We introduce this add a new project for all other students. Because the students can practice version controlling using their other projects. In this research, there is another user called the client and the students can take projects from clients. So students can use this add a new project feature for that also.

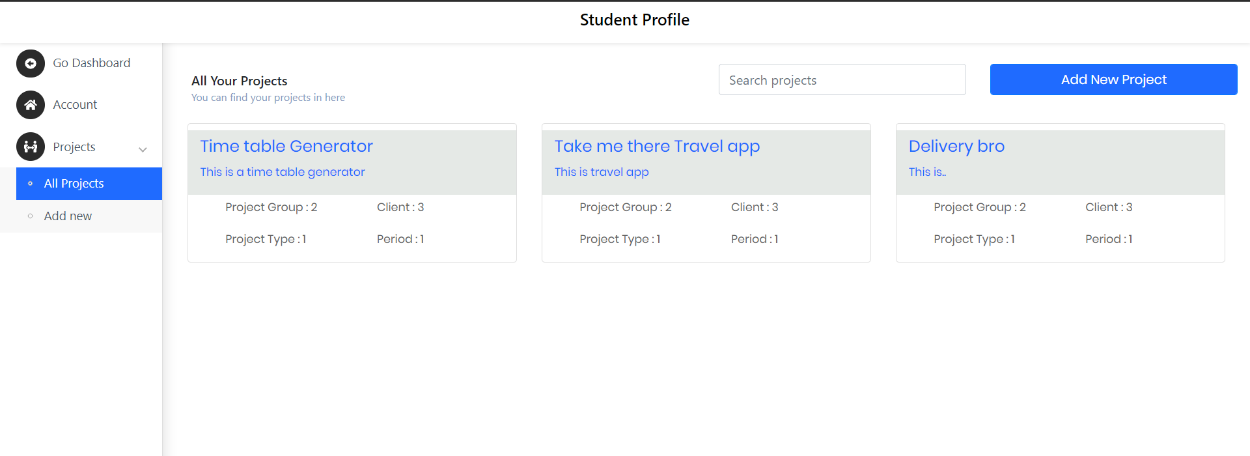


Figure 13: User's all projects

Figure 13 represents the user’s all projects interface. In this interface, you can see there are three projects that have been owned by the logged-in user. These projects are from different project groups. Also user can create/add a new project.

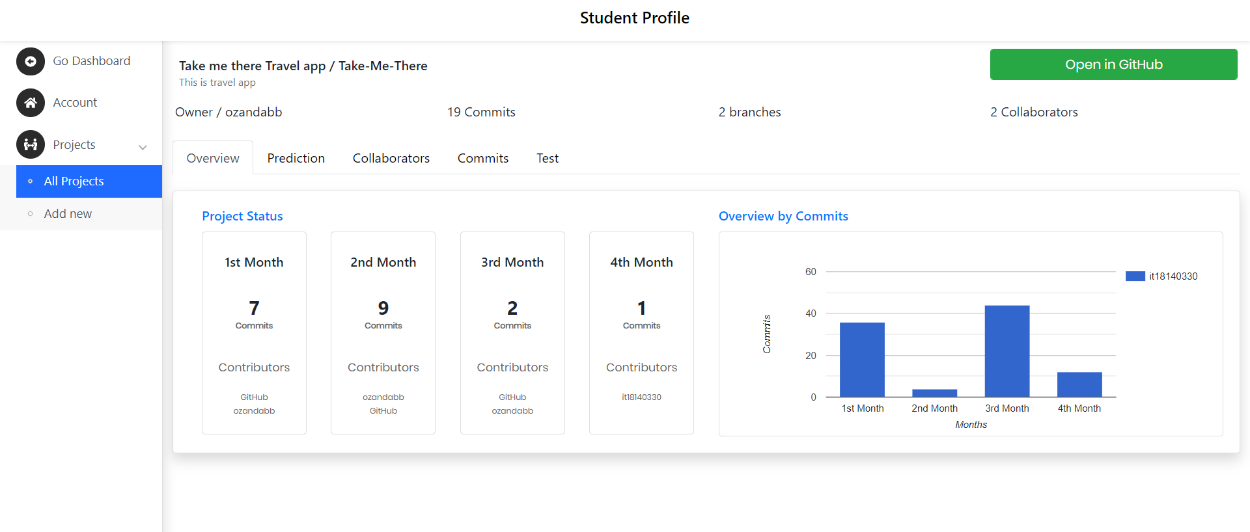


Figure 14: Overview for single project

Figure 14 represent the overview for a single project. As you can see common GitHub data also display in here. Such as owner, total commits, branches and the collaborators. In the overview tab represents the overall project status using total commits per each month and the commits chart for each month.

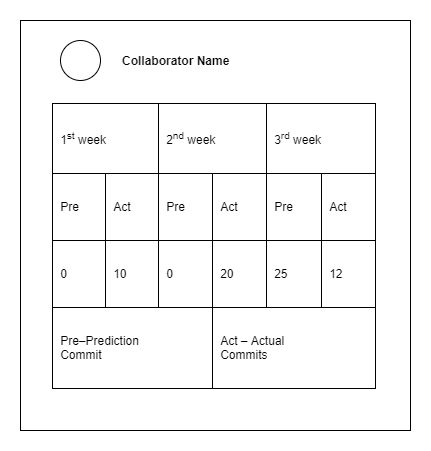


Figure 15: Prediction for a single user

In figure 15, is the prediction component for a single user. As you can see there are several weeks and each week represents the actual commits and the prediction commits. As I mentioned in the methodology, next week will predict with the last two weeks. So prediction for 4th week will predict with 2nd and 3rd weeks actual commits.

After each section is completed it transfer to testing to verify and validation. we used React JS technology to develop the frontend part and Redux to store token in localhost (When some user log into the system there is a token to identify each uses. Redux will helps to store the token) and for the backend part we used python and JAVA. MySQL is our database.Visual Studio Code, Eclipse IDE and Xampp are the tools for relevant technologies.

### **2.8.2.2 API Development**

For the backend part, we use both Java and Python. We use Java to create basic endpoints. Such as login, register a user, register a new project like wise.

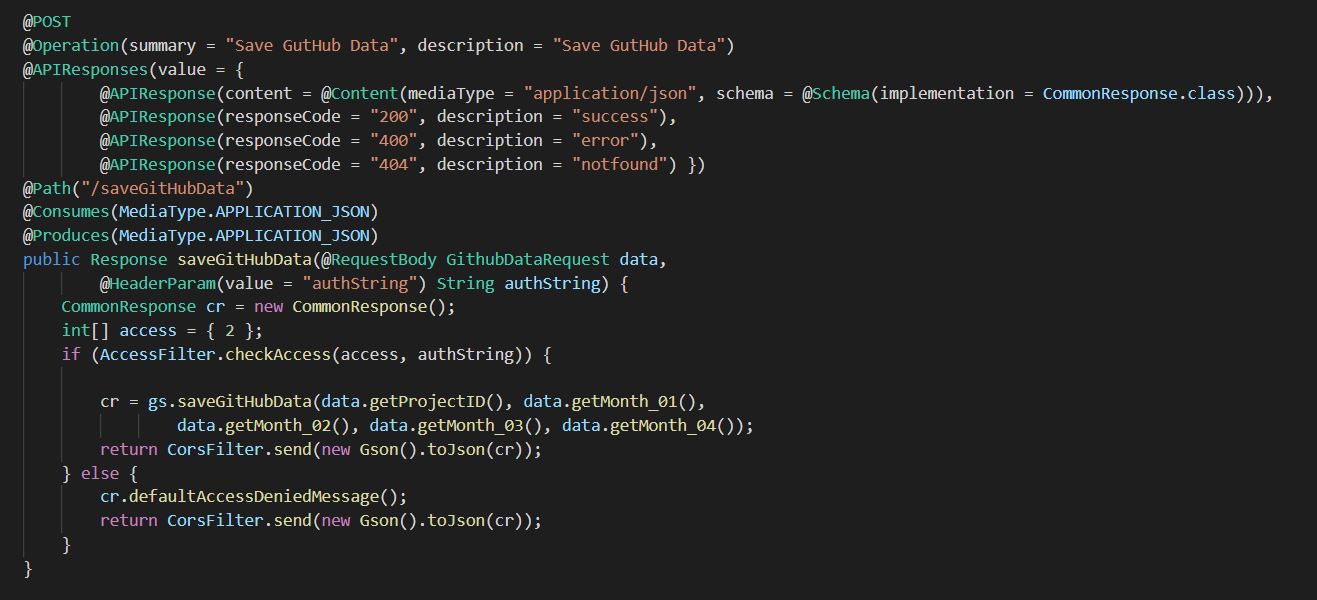


Figure 16: save GitHub data endpoint

To develop the algorithm, we use python and Machine Learning. We used pyrunner to run the python script that has developed in our system.

### **2.8.2.3 Supervised Learning for Prediction Process**

#### **2.8.2.3.1 What is machine learning algorithm?**

Machine learning algorithms are the systems that can learn from data and build on knowledge without human intervention. It is known as a part of artificial intelligence.

#### **2.8.2.3.2 Types of machine learning algorithm.**

There are four types of Machine Learning Algorithms.

* Supervised Learning Algorithm
* Unsupervised Learning Algorithm
* Semi-supervised Learning Algorithm
* Reinforcement Learning Algorithm

In this research, we are using Supervised Learning Algorithm. So we need to know what is Supervised Learning and why we are using that.

#### **2.8.2.3.3 What is Supervised Learning Algorithm?**

As I mentioned in the Introduction, Supervised learning, also known as supervised machine learning, is a subcategory of machine learning and artificial intelligence. This uses labelled training data to learn the mapping function. We know,

* Classes as well as the labels.
* What attributes used in the model.
* Historical application details.
* We have the dataset.

The main reason to use Supervised Learning Algorithm is we have the data set. We collect the dataset as commits from previous weeks. As I mentioned above we need at least last two weeks commits to predict. Also we don’t need previous projects’ commits. We need only each project each commits per user. Because we predict each member’s future contribution. So we need only each project each member’s contribution as commits.

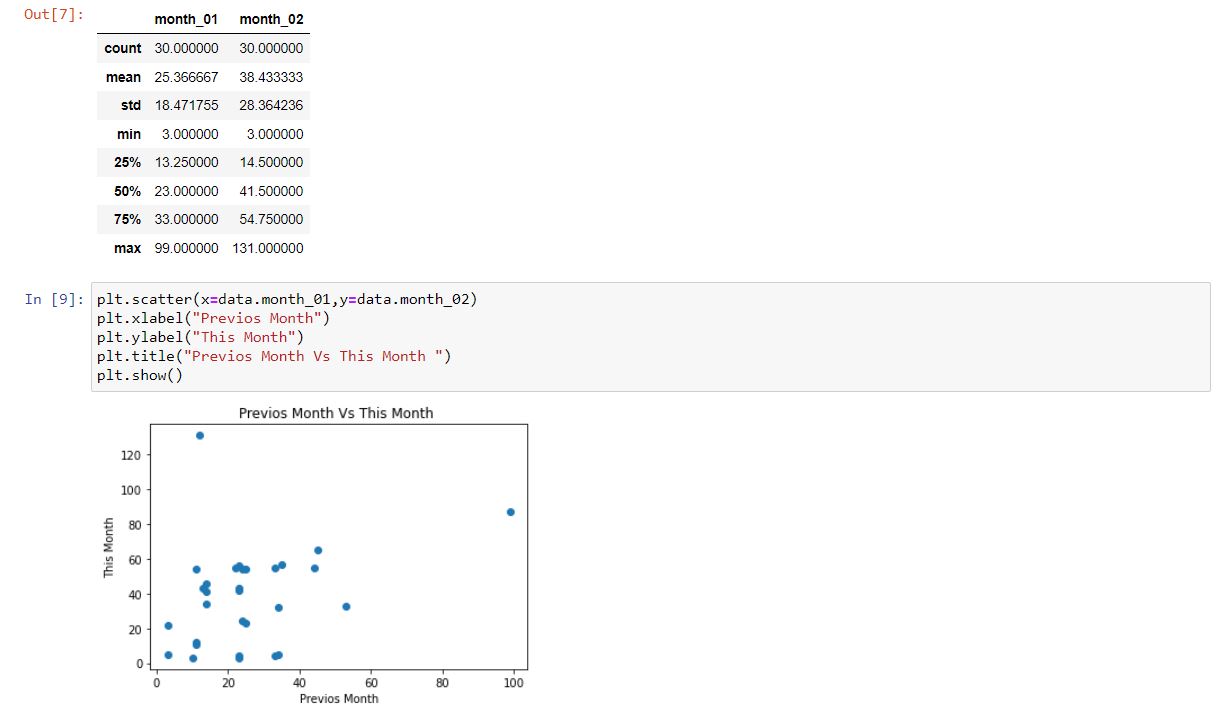


Figure 17: Describe of data and a chart with previous vs this month

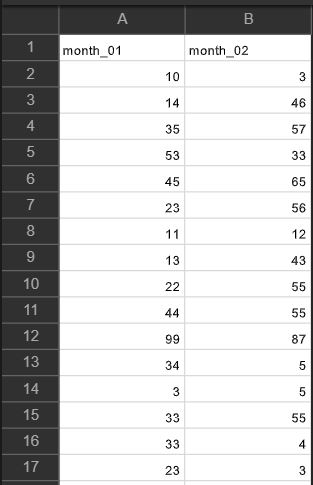


Figure 18: sample data set



Figure 19: Data cleaning



Figure 20: Select a model and train it

Figure 2.8.2.3.4 represent a sample code for prediction process. In here we predict next month with the last two months. We import LinearRegression from sklearn.linear\_model and then we assign LinearRegression as lr=LinearRegression().

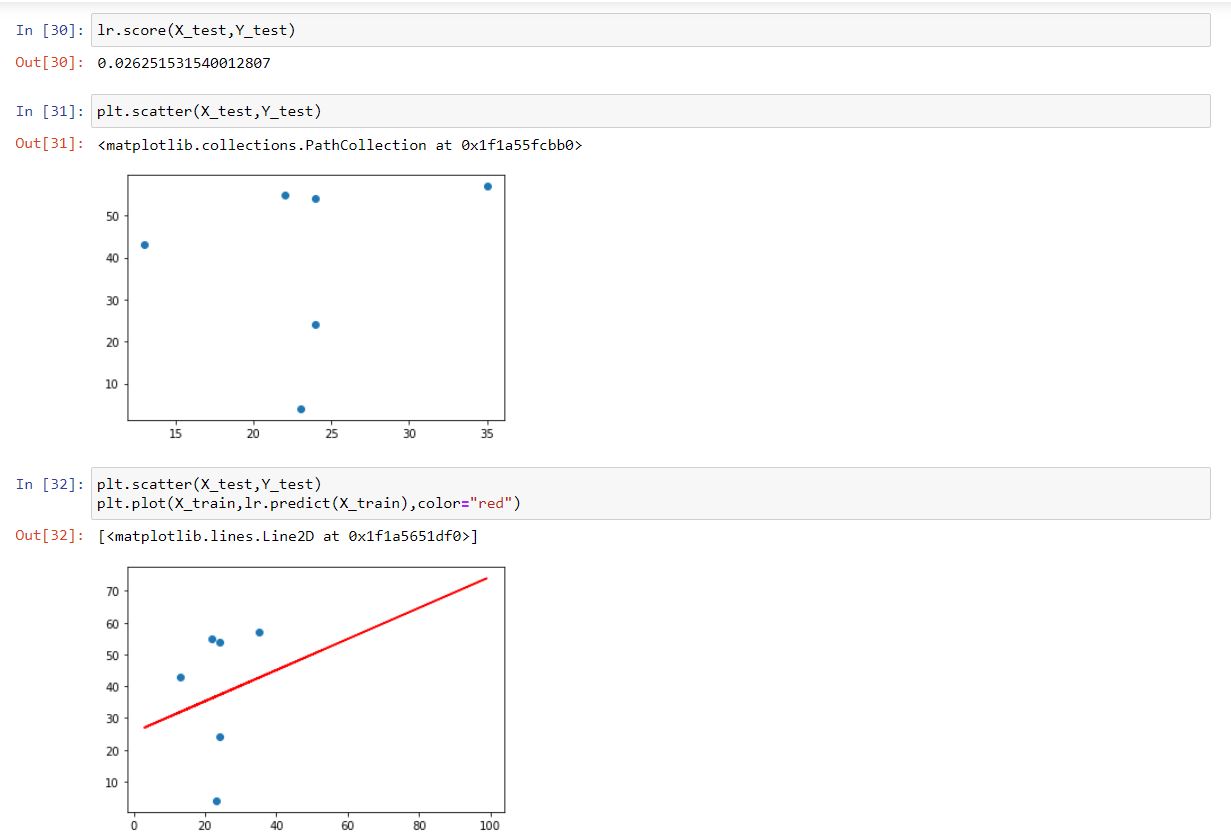


Figure 21: Fine Tuning the Model



Figure 22: final result

# 2.9 Deployment

* AWS

AWS has significantly more services, and more features within those services, than any other cloud provider–from infrastructure technologies like compute, storage, and databases–to emerging technologies, such as machine learning and artificial intelligence, data lakes and analytics, and Internet of Things. This makes it faster, and more cost effective [10].

* Ngrok

For the demonstration purposes we sometimes need to use other API implementations for the frontend implementations. To do that we can use Ngrok to temporary host the localhost and access the API anywhere. But the free version is only providing 2hr of online period. After that period ngrok needed to be restarted with a new URL. But for the temporary testing, ngrok is best solution for the temporary global hosting.



Figure : ngrok logo

# 3 RESULTS AND DISCUSSIONS

## **3.1 Results**

This research implementation worked collectively and interactively. GitHub prediction was able to predict the future contribution to the project relatively to each group member. First of all, Process should start with the grouping process. This is another research from my team. After the group generation process, the next step is to manage projects and project repositories. In order to do that, system uses GitHub integration using GitHub API. By retrieving repository data, System can generate GitHub contribution predictions.

Table 8: Prediction Overview for User

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1st week | | 2nd week | | 3rd week | |
| Pre | Act | Pre | Act | Pre | Act |
| 0 | 10 | 0 | 20 | 25 | 12 |
| Pre–Prediction Commit | | | Act – Actual Commits | | |

Table 2 represents the GitHub contribution prediction generated using the proposed system. Since the system considers the data from the previous weeks, next prediction should be according to the previous data and using these predictions, supervisors and the student himself can identify if project completion is possible. However, since there are no researches conducted by researchers for GitHub contribution prediction, this is the first research that GitHub contribution added to a project management system.

When we consider about the display all projects to a relevant logged in user, all the projects are display for each logged in user (Figure 13 in the implementation section). For the single project display project data page also displaying data successfully (Figure 14 in the implementation section).

## **3.2 Research Findings**

The key goal of this research was to predict future contribution per each student and progress tracking.

While researching this topic, we identified, the accuracy of this research, predict future contribution, can’t use in this research. Because this is just a prediction and the prediction process work with the last weeks as a prediction for the next week. So we display actual commits and the prediction commits for each student. We identified that was the way to predict each student contribution using commits contribution.

When we consider about progress tracking, we identified using GRA, can track project and the contribution. Only student has to pass the PAT, Repository name, Owner name and the GitHub URL. Using that data passing, system can track the project.

If the findings are compared with the existing research. Here, the algorithms used are very effective. While this analysis covered the newest developments in the application growth. Because this research is a new finding. The application is very good as it responds with less than a few seconds. This proposed system can also be trusted by consumers.

## **3.3 Discussion**

This research is a new finding and the existing researches are “Predicting the Popularity of GitHub Repositories [2]”, “Healthy or Not: A Way to predict ecosystem health in GitHub [3]”, “Using Dynamic and Contextual Features to predict Issue Lifetime in GitHub Projects [4]”. You can see all other researchers predict outside of the project. Such as the Popularity, Project Healthy, Issues like wise. No any other research found related to predict future contribution for each student.

When we evaluate the algorithm Supervised Learning, the data is first analyzed, and null values removed by in the cleaning process. Then get the Linear Regression model to evaluate a few algorithms. Identify key features and train models dependent on the value of the component.

# 4. SUMMARY OF THIS RESEACH CONTRIBUTION

As a summary of this research, the main key goal is to predict future contributions per each student in a group project. c for a student is to make a connection between the supervisor and the student. Because the supervisor can track the project by the student’s tasks and supervisor can notify students when they are in slow contribution.

Finally, our goal is to train the version controlling for undergraduate students before they getting into the industry. Because industry accepts version controlling.

# 5 CONCLUSION

With the proposed project management system was able to manage the workflow of a common project in a university environment. Combined with major requirements for a university projects, system was able successfully execute a complete project management process workflow. Starting with group generation, GitHub contribution predictor predicts future contributions from the students. Furthermore, with the meeting documentation process, students can check previous meeting details. And finally the peer review was able to contribute for the evaluation phase of a project management process. Since the proposed project management system was able to cover major steps of project management.

Even so the system is functional as proposed, system can be further improving with new technologies and features. Most of project management systems that currently in use are mostly for common purposes. Developing a configurable project management system is much more valuable for institutes since they have different variations of requirements. By developing such system is much more effective for an educational environment.

In university life, students do their project as individual and group. For the individual part, they do not use version control systems. But some of them are using version control systems. It is okay to have that. Because since it is an individual project and there is no integration process.

But when the project is a group project, students need to use version control systems. Because there is an integration process in there. So just use that. Because lectures can also track students’ contributions to the group project. But the lecturer or a supervisor can track the project, only in the viva sessions. So the supervisor has no idea about each student’s tasks and contribution to the project. The SMART PMS is a solution for that problem. In here supervisor can track the student contribution from the begging of the project. This is a platform to track the project progress and predict future contributions.

When we consider the future prediction part, the system can inform the students to do their tasks. As I mentioned above supervisor can track the student's contribution. If a student is weak with their task, it means if a student cannot complete their task by a given time period, the supervisor can send a reminder or a warning message through the system. So the student will learn time management with deadlines. To identify if a student is weak at their work, the system will display the prediction table for each student (Table 8). So the supervisor can get an idea about the student's contribution.

# 6 DESCRIPTION OF PERSONAL AND FACILITIES

Table 9: Description of personal and facilities

|  |  |  |
| --- | --- | --- |
| Member | Component | Task |
| Gamage O.M. | Predict future contribution percentage possibility using GitHub contribution. | * Assign task or a topic by a lecturer or supervisor part * Track the GitHub project progress * Generate a report related to project progress * Generate a grade related to project progress * Lecturer or a supervisor review or a comment related to the project progress * Issue tracking * Send Notification about incoming events and due dates |

After Google form is published, the next step is waiting until form gathered considerable amount of responses. We can check the responses that recorded by the google form after the data collection is performed. Google form records each response separately and outputs a summary of answers for each question. Google forms provide a feature to export responses to an excel or as a CSV file. Using this feature, we can directly use the csv file or the excel file for next steps which is training and testing phases in the model training.

# 7 Reference List

[1] Georgios Gousios, Bogdan Vasilescu, Alexander Serebrnik and Andy Zaidman, “Lean GHTorrent: GitHub data on demand”, MSR2014, Available at <https://dl.acm.org/doi/abs/10.1145/2597073.2597126>

[2] Hudson Borges, Andre Hora and Macro Tulio Valente, “Predicting the Popularity of GitHub Repositories”, PROMISE 2016

[3] Zhuang Liao, Mengjie Yi, Yan Wang, Shengzong Liu, Hui Liu and Yan Zhang, “Healthy or Not: A Way to predict ecosystem health in GitHub”, 28 January 2019

[4] IEEE, “Using Dynamic and Contextual Features to Predict Issue Lifetime in GitHub Projects”, 2016 IEEE/ACM 13th working Conference on MSR.

[5] GitHub Documentation Guide, ”Repositories”, Available at <https://docs.github.com/en/rest/reference/repos>

[6] GitHub Documentation Guide, “ Users “ , Available at <https://docs.github.com/en/rest/reference/users>

[7] GitHub User insight using GitHub API – Data Collection and Analysis, Available at <https://towardsdatascience.com/github-user-insights-using-github-api-data-collection-and-analysis-5b7dca1ab214>

[8] The 2020 state of the OCTO-VERSE, Available at <https://octoverse.github.com>

[9] IEEE, “Comparison of static Code Analysis Tools”, 2009 third International Conference of Emergency Security Information, Systems and Technologies

[10] Amazon Web Services. (2021). Cloud computing with AWS. [online]. Available: <https://aws.amazon.com/what-is-aws/?nc1=f_cc>. [Accessed: 21-Feb-2021]

[11] GitHub Documentation Guide,” REST API”, Available at <https://docs.github.com/en/rest>

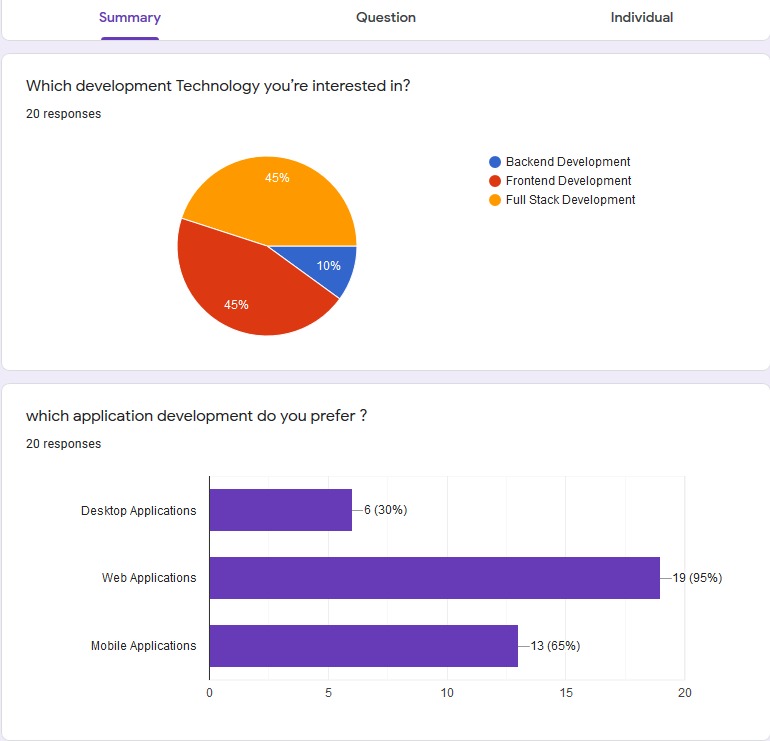
[12] GitHub Documentation Guide, “Creating a personal access token”, Available at <https://docs.github.com/en/github/authenticating-to-github/keeping-your-account-and-data-secure/creating-a-personal-access-token>

# 8 APPENDIX

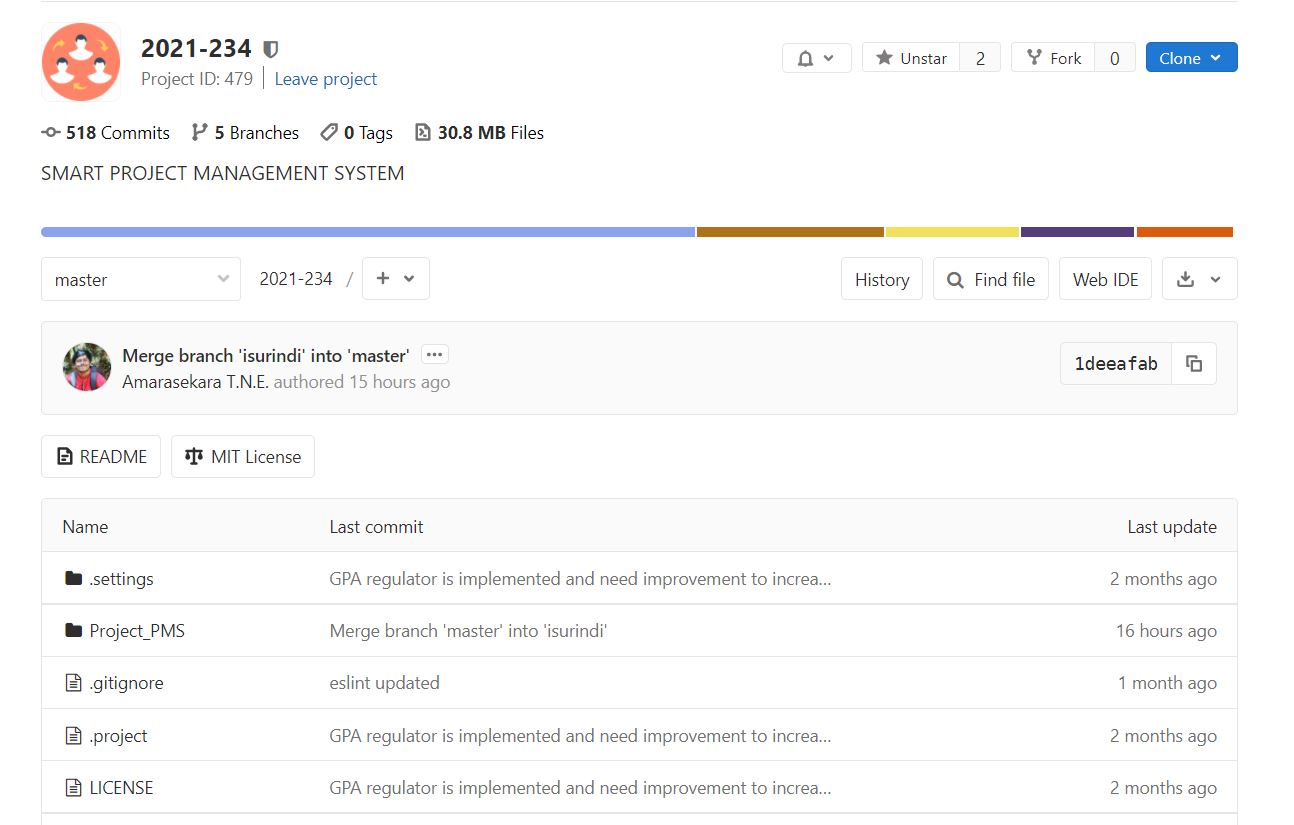
Appendix 1:Sample questionnaire – Collect the data

# 

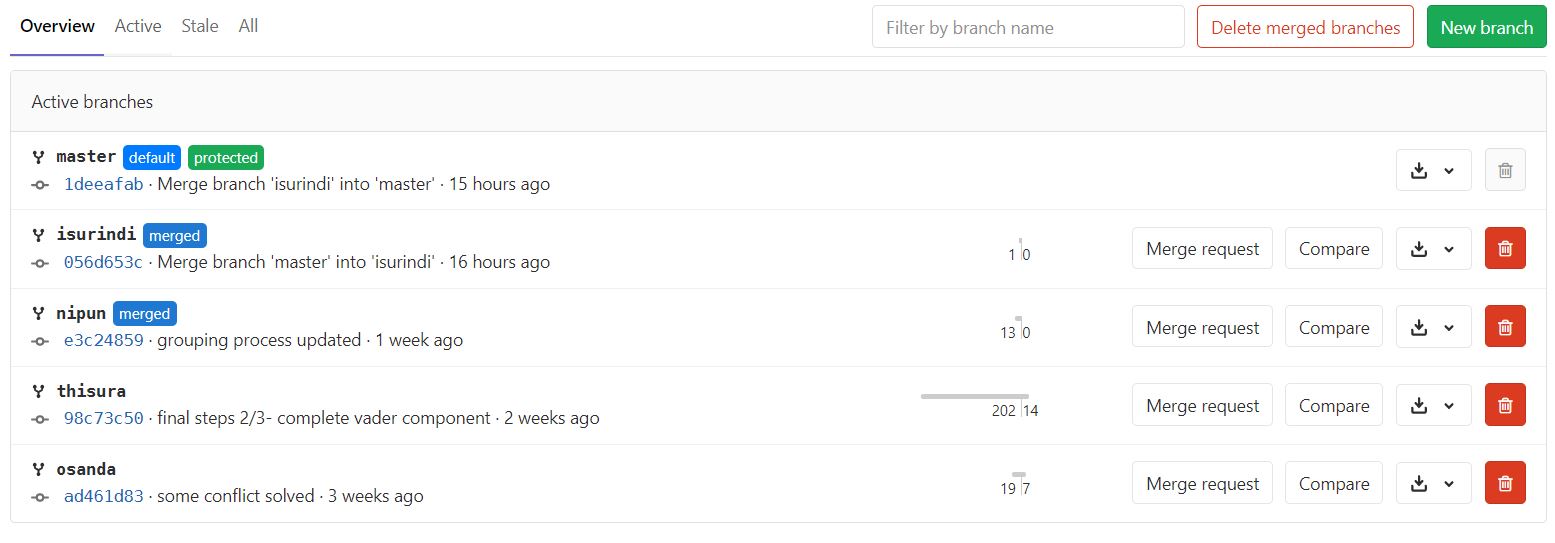
Appendix 2: Sample questionnaire response



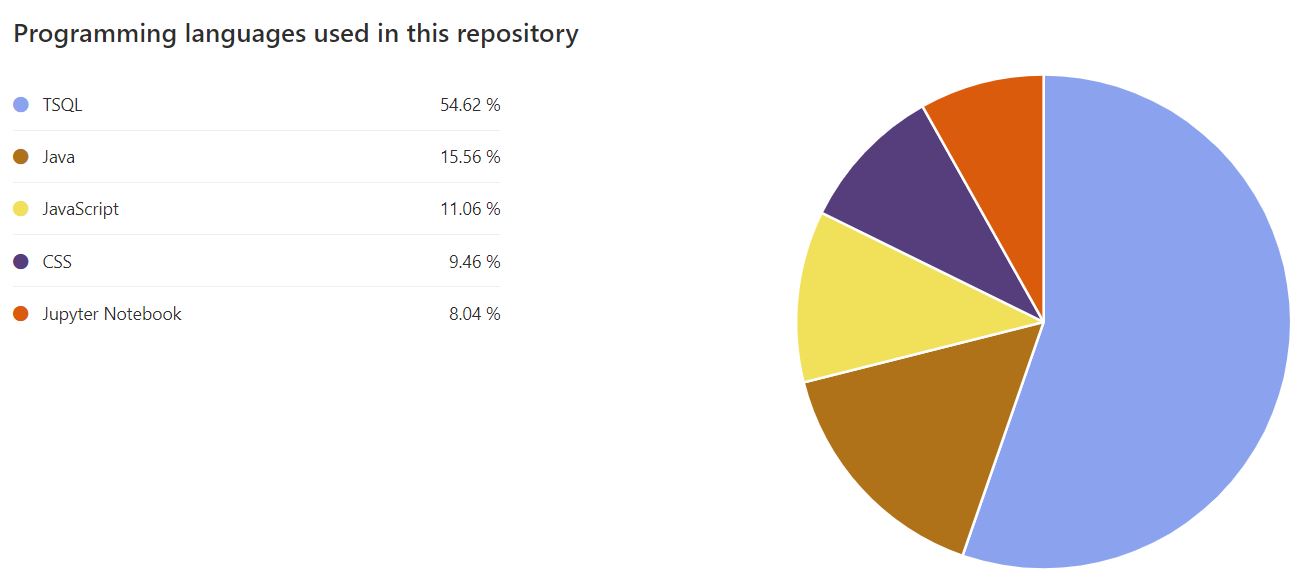
Appendix 3: GitLab Repository details



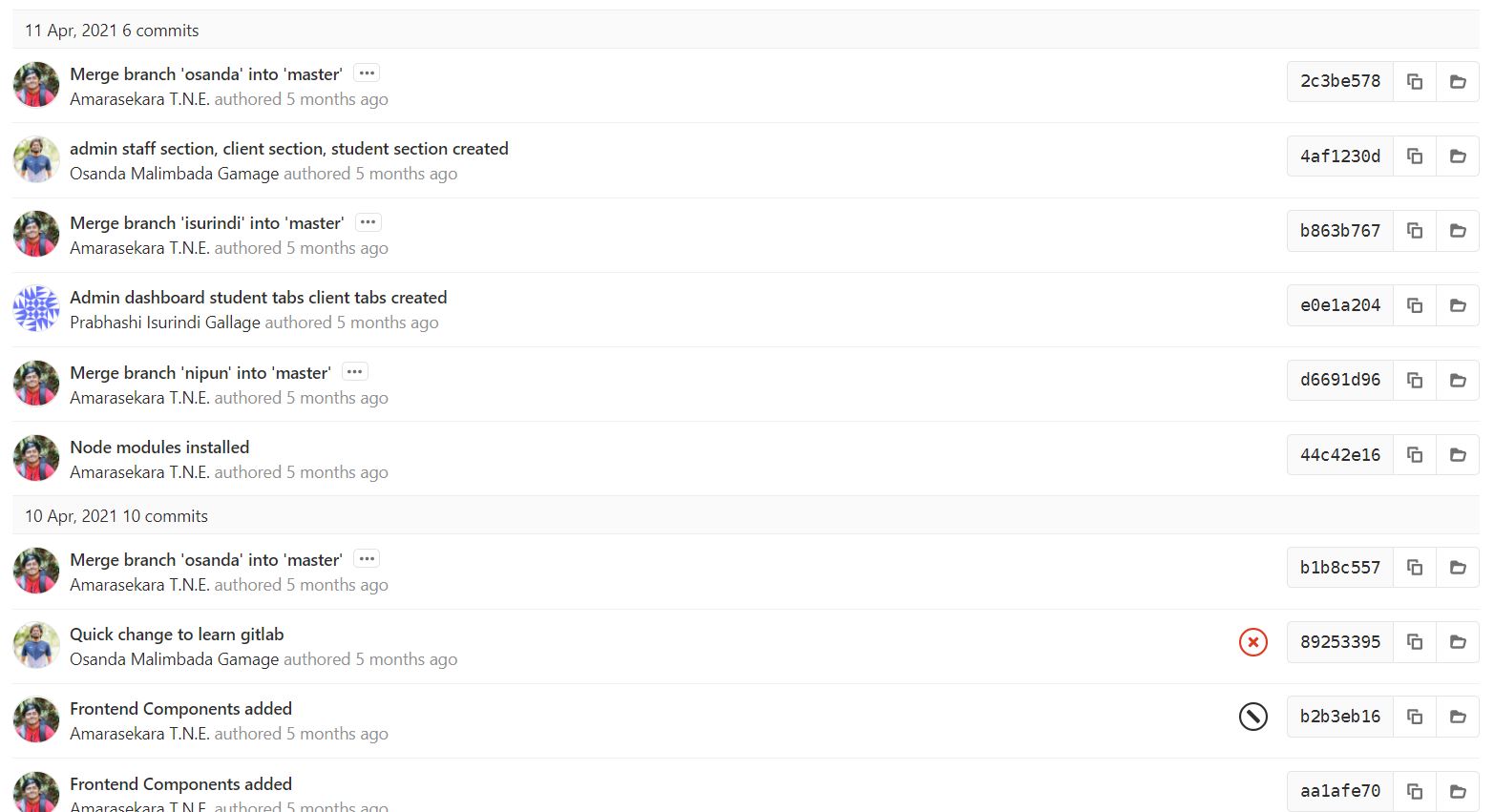
Appendix 4: Branches in the Repository



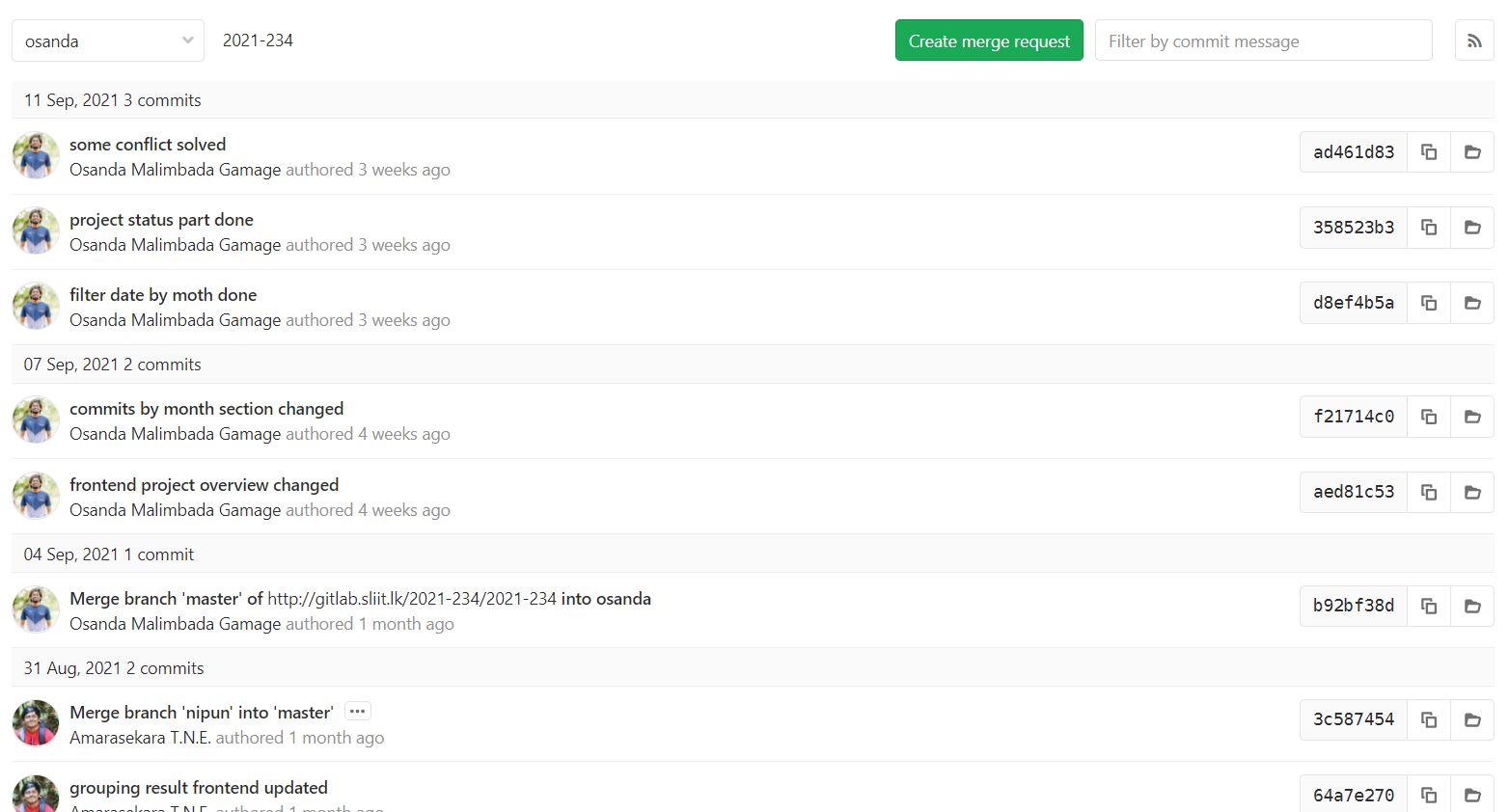
Appendix 5: Programming languages used in this repository



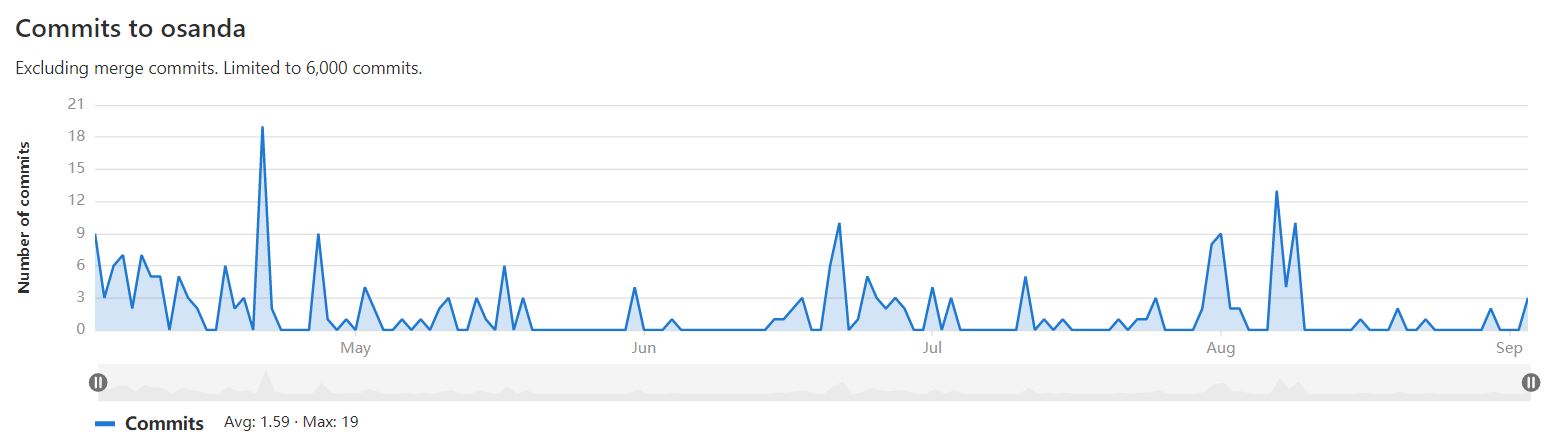
Appendix 6:First Contribution from my branch



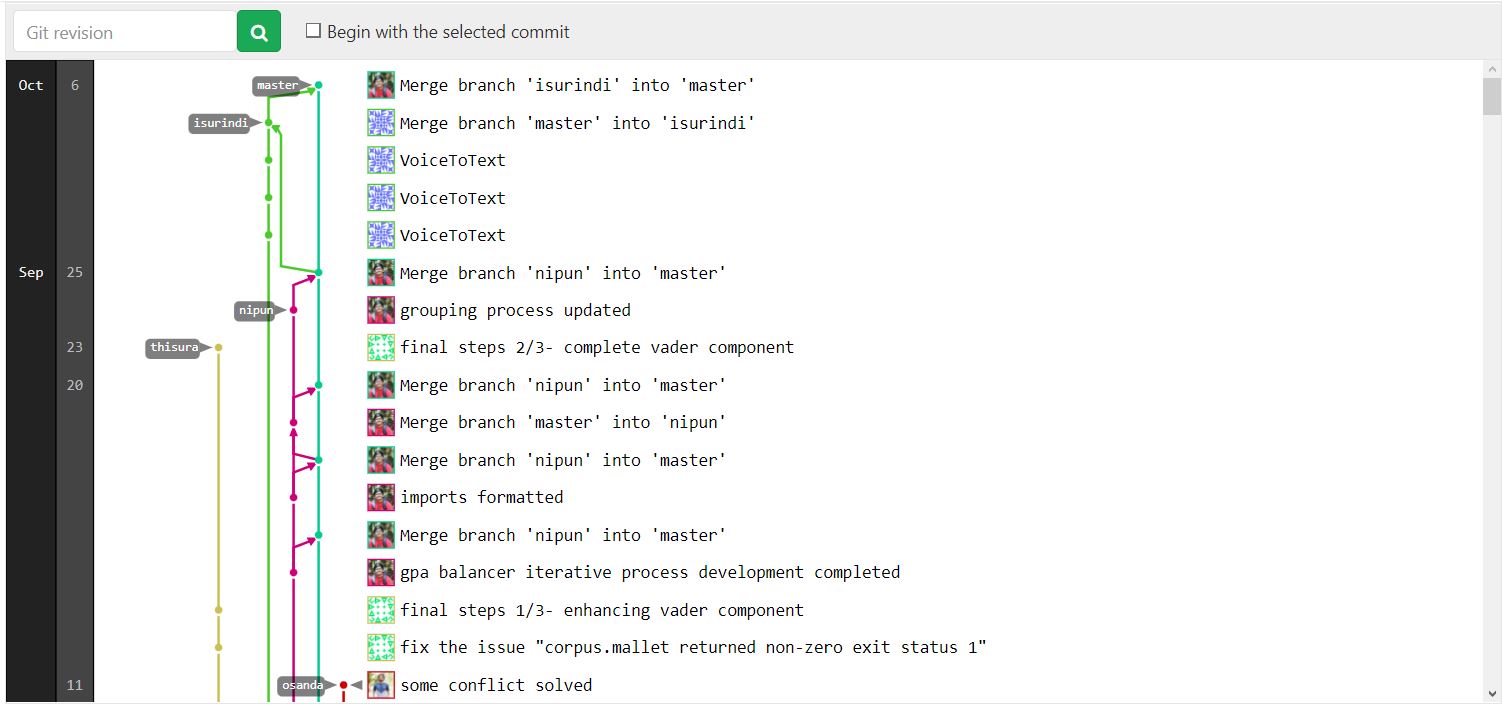
Appendix 7: Last Contribution from my branch



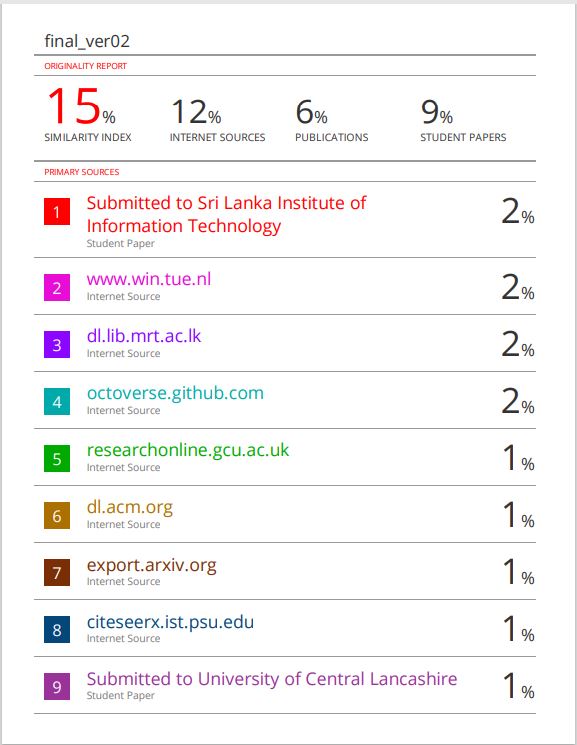
Appendix 8: Commits from me



Appendix 9:Graph chart in GitLab



Appendix 10: Plagiarism Score



Appendix 11: Plagiarism Score 02



Appendix 12: Plagiarism Score 03

