**SMART PROJECT MANAGEMENT SYSTEM**

**2021-234**

Project Proposal Report

Amarasekara T.N.E.

B.Sc. (Hons) Degree in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

February 2021

**SMART PROJECT MANAGEMENT SYSTEM**

**2021-234**

Project Proposal Report

B.Sc. (Hons) Degree in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

February 2021

# Declaration

We declare that this is our own work and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our 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.

|  |  |  |
| --- | --- | --- |
| Name | Student ID | Signature |
| Amarasekara T.N.E. | IT18009446 |  |
|  |  |  |
|  |  |  |
|  |  |  |

The supervisor/s should certify the proposal report with the following declaration.

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor: Date :

# Abstract

Worldwide universities conduct various types of Information Technology Projects. Most of the universities uses common project management solutions which includes industrial quality functionalities. However, there are requirements that doesn’t come with these common project management systems. Automatic Group Generation is one of the major requirement in universities. Project practical teaching has an important significance to enhance students’ practical ability and is also an important part in teaching reform in colleges and universities, however, there is lacking of scientific strategies to grouping students in project practical teaching [5]. Universities doesn’t use any logical way for grouping process. Usually groups will randomly generate or according to a registration list. However, the efficiency and effectiveness of a such group is unpredictable. Therefore, a Computer Based Group Generation system becomes a major requirement for universities.

The Smart Project Management System is proposed to overcome above mentioned requirements as well as most of the common features. System will generate project groups using student Skills, GPA and previous project solutions. With the GitHub integration to the system, grouping process can reach maximum accuracy. However, to calculate the accuracy system need to access various types of skills of the students. Therefore, System will provide facility to manage student profiles by their own, and students can define their very own skills on Programming knowledge. Using this strategy system can use the data that required for the grouping process. Once the process is finished assigned lecturers can check the groups and publish groups among students.

The end result of the solution is a project management system with highly customizable web application. Since the system will an open source solution, Universities can expand the capabilities of the system by their own. Smart project management can be used to improve efficiency and effectiveness in group projects.

# TABLE OF CONTENTS

[Declaration………………………………………………………………………........ i](#_Declaration)

[Abstract…………………………………………………………………………….... ii](#_Abstract)

[Table of Contents..………………………………………………………………….. iii](#_TABLE_OF_CONTENTS)

[List of Figures…………………………………………………………………...…... iv](#_LIST_OF_FIGURES)

[List of abbreviations………………………………………………………………… v](#_LIST_OF_ABBREVIATION)

[List of Appendices………………………………………………………………….. vi](#_LIST_OF_APPENDICES)

[1 Introduction……………………………………………………………………... 1](#_1._INTRODUCTION)

* 1. [Background & Literature Survey……………………………………………. 2](#_1.1_Background_&)

1.2 [Research Gap………………………………………………………………… 6](#_1.2_Research_Gap)

1.3 [Research Problem……………………………………………………………. 7](#_1.3_Research_Problem)

[2 Objectives………………………………………………………………………. 9](#_2._OBJECTIVES)

[2.1 Main Objectives……………………………………………………………... 9](#_2.1_Main_Objectives)

[2.2 Specific Objectives…………………………………………………………. 10](#_2.2_Specific_Objectives)

[3 Methodology………………………………………………………………….. 11](#_3._METHODOLOGY)

[3.1 System Architecture....……………………………………………………... 12](#_3.1_System_Architecture)

[3.1.1 Software Solution……………………………………………………… 13](#_3.1.1_Software_Solution)

[4 Description of Personal and Facilities………………………………………… 19](#_4.__)

[4.1 Data Collection GitHub and Google Forms………………………………… 20](#_4.1__)

[4.2 Usage of gathered Data Collection…………………………………………. 21](#_4.2__)

[4.3 Other Functions…………………………………………………………….. 22](#_4.3__)

[5 References…………………………………………………………………....... 23](#_5._REFERENCES)

[6 Appendices……………………………………………………………………... 25](#_6._APPENDICES)

# LIST OF FIGURES

[Figure 1.1 - factors that can be considered for group formations……………………. 4](#_Figure_1.1:_factors)

[Figure 1.3 - Feature Comparison with the existing solutions and proposed system… 8](#_Figure_1.3:_Feature)

[Figure 2.1 - Grouping Algorithm flow chart…………………………………………. 9](#_Figure_2.1:_Grouping)

[Figure 3.1 - System Architecture Diagram………………………………………..... 12](#_Figure_3.1_–)

[Figure 3.1.1 - SDLC Life Cycle…………………………………………………...... 13](#_Figure_3.1.1:_SDLC)

[Figure 4 - Overall Grouping Process……………………………………………….. 19](#_Figure_4:_Overall)

# LIST OF ABBREVIATION

Abbreviation Description

GPA Grade Point Average

SLIIT Sri Lanka Institute of Information Technology

API Application Programming Interface

SDLC Software Development Life Cycle

XML Extensible Markup Language

ITP Information Technology Project

JS JavaScript

UI User Interface

UX User Experience

JDBC Java Database Connectivity

AWS Amazon Web Services

CSV Comma Separated Values

# LIST OF APPENDICES

Appendix Description Page

[Appendix – A Sample questionnaire 25](#_Appendix_–_A:)

[Appendix – B Sample questionnaire response 26](#_Appendix_–_B:)

[Appendix – C Gantt Chart 27](#_Appendix_–_C:)

Appendix – D Plagiarism Report 28

# 1. INTRODUCTION

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.

Automate Group generation is one of the required feature for universities. However, there are existing systems used to automate group generation. But our goal is to develop a system with fair group generation. This system uses specific data from the students and using machine learning approach, we’re developing a system where group generation can be done fairly. Most of the time students suffering and complaining about conflicts among members. Our goal is to reduce those conflict by grouping student with similar skill structure. Existing project management systems doesn’t come with this feature.

To achieve this goal, we need to retrieve student data to process. But the problem is how we gather all these data. Students grow up with different skills and goals. Therefore, system need to gather data about student’s skills and goals to come up with a conclusion and continue the grouping process. Student grouping is depending on the skills and goals that student define in the system. Therefore, the overall success rate is depending on the student data which defined by the student.

# 1.1 Background & Literature survey

Project Management Systems doesn’t come with facility to automatically generate project groups. Instead user must create a group and assign team members to the group. It is a time consume process when there are thousands of students. But from onward there are lot of facilities provided by the project management system and most of the time they’re not even use most of the facilities. Problem with all these system is the main focus is Industrial projects. They’re not exactly made for university use. By creating an open source project management system, users can create solutions for their requirements and also they can expand the system by themselves. Most of the project management systems are not open source solutions.

According to the existing grouping strategies in project practical teaching, teachers often grouping students according to their willingness, or by their enrollment numbers, or dividing them into groups at random [1]. By dividing students by these strategies not very accurate and effective. Students have different skills, different knowledge on programming and also their theoretical and practical skills are also different. By using non logical strategies for grouping effect on group management and also the success rate of the project. Therefore, we need a solution to integrate this grouping facility with the project management system. But the problem is student attributes that a university may consider is may be different than other universities when considering project groupings. By creating an open source project management system solves this issue. So the final solution will be a configurable and extendable project management system.

Since there are no automatic grouping facilities in any project management solutions, we need to integrate the grouping facility with the project management system. However, we need to implement the grouping facility first. In order to create better study groups, automatic computer-supported methods are proposed. This leads to several important advantages. Especially, it is possible to consider a large amount of information even from very different sources. Group creation can be performed very fast and anytime on demand by students or a learning system itself. Computer support also allows creation of anonymous groups in which members do not know their identity [16]. Therefore, we decided to explore some of the research projects and papers that researchers proposed. There are existing grouping solutions created by varies researchers. There is a grouping solution using fuzzy clustering. By grouping students considering scores of students’ previous professional curriculum as clustering characteristic quantity index, and bases on the index, divides students into different classes, then grouping students by heterogeneity grouping method reasonably and accurately [5]. In this research they divide students into 3 different classes. That is “very important”, “important” and “common”. After that they apply each class values to a weight comparing matrix table and based on weight scores they generate a project team structure based on class values. Issue in this grouping method is it only considers professional curriculum. In this research they used Object-Oriented Programming, Database Foundation and Application, Data Structures and Operating systems. There are multiple Object-Oriented Programming languages and likewise there are multiple sub sections to be consider when we need a fair grouping strategy. But this existing solution is also fair enough to increase the success rate and efficiency. But this solution can be extended and improved by using multiple algorithms to sort end results and increase efficiency.

GroupEng is another solution created by researchers to effectively assign student groups by applying multiple User-prioritized Academic and Demographic Factors. This solution is far more complex than the other one. Using GroupEng users can group students using different types of rules. This fairly removes the above mention issue of lack of options. GroupEng solution uses heuristic guided stochastic greedy algorithm. Greedy algorithm simply iterates through the rules in priority order fixing “breaks” by swapping students between groups. The following criteria are used to determine if groups “break” a rule:

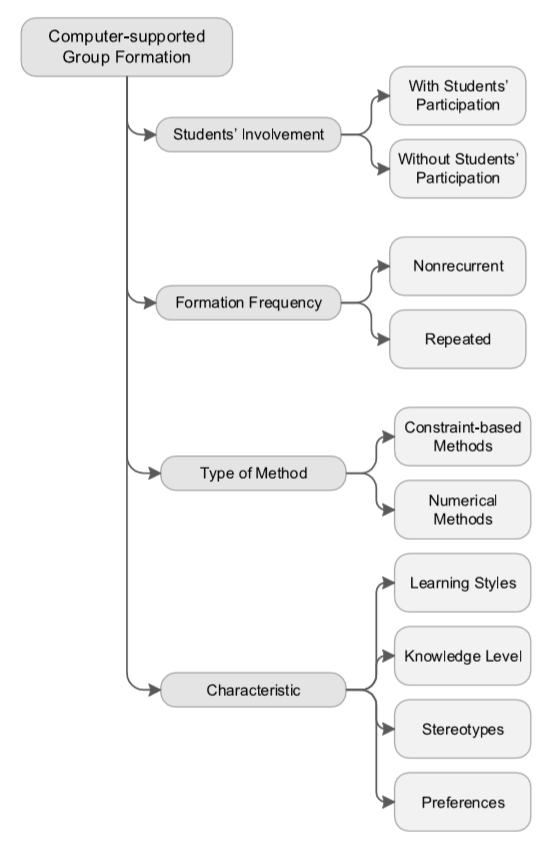
Cluster: broken by groups which contain exactly 1 student with the given value.

Aggregate: broken by groups which mix values for the given attribute. Usually1 group will break this rule for each value under consideration because the number of students is not evenly divisible by the group size. This is considered normal operation.

Distribute: broken by groups which have more or less than their “share” of members with any attribute value under consideration. This “share” is (#students with attribute value) / (# groups), or the two integers on either side of this value for fractions.

Balance: broken by groups for which the standard deviation of group member’s values for the attribute is larger than a tolerance times the class’s standard deviation for that attribute: StDev(group) > tol \* StDev(students).

Once a high priority rule is finished, swaps which would break that rule are not allowed while processing lower priority rules. Uneven group sizes are handled by filling out the smaller groups with placeholder students which are considered to have the lowest value in the class when computing deviations for balance rules and otherwise ignored [2]. GroupEng Solution achieved higher accurate results than other research solutions that we’ve studied.



## Figure 1.1: factors that can be considered for group formations

When we consider grouping mechanisms there’re wide range of options can be selected. But the properties that include in those mechanisms are differ from another. Figure 1.1 shows the different factors that can be used for group formations. Since there are several choices to have, we need to select which factors can be used to achieve maximum efficiency and fairness.

Varies Researchers have worked on this topic and they’ve introduced solution as a part but not as a whole system. All these solutions are standalone solutions which not included in any project management systems that are currently in use. Our goal is to implement this Automatic Group Generation functionality to the Project management system and by doing that create a Smart project management system which satisfies major requirements that universities require.

# 1.2 Research Gap

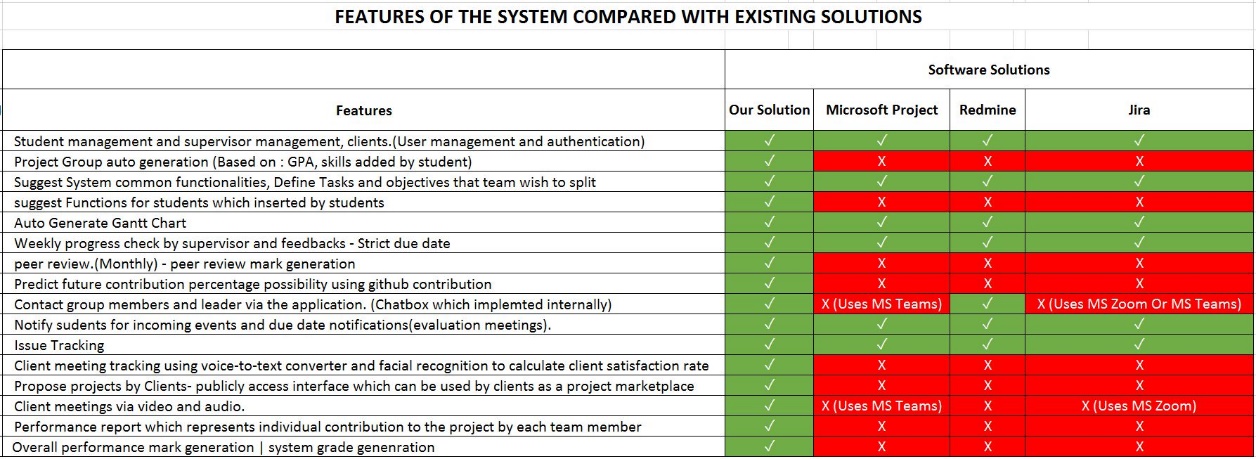
Above mentioned research solutions are targeted for a specific requirement that only satisfy specific dataset. The problem is, is it configurable for our own requirements and is it possible to use these solutions for our own system. Since all above mentioned systems are made for a single task it is bit complex to integrate it to the proposed project management system. The target is to use existing solutions and reconfigure them to satisfy our own requirements. Some of the solutions are complex to be configured and some of them are not enough sufficient to satisfy our requirement. So to implement an efficient and accurate system, we need to study these existing systems to configure and extend capabilities of the existing solutions.

According to the existing systems [5] [2] they specify what type of data need to be supply to the system and the end results that they expect from the implemented system. When we consider the Fuzzy Clustering algorithm [5], It considers student GPA and the field that student interested in. By using only these two components they managed to implement the grouping solution which is effective enough to make difference in the success rate. But the problem is variables that they used for the calculation is not sufficient to expect highest accurate end results. However, the GroupEng open source solution [2] is way ahead than the first research solution using Fuzzy Clustering Algorithm. But the problem GroupEng is a far more complex solution is for the expected solution. But the grouping process is mostly similar to the system that we’re expecting to build. So this solution is also a valuable resource for the development process.

# 1.3 Research Problem

Most Universities uses a standard grouping mechanism. One of the researches addressed this problem. According to the existing grouping strategies in project practical teaching, teachers often grouping students according to their willingness, or by their enrollment numbers, or dividing them into groups at random [1]. Using this strategy, the grouping problem is solved. But the problem is grouping that way creates conflicts within groups and students start to complain about the situations. But if the university create groups for them. They complain about different things. when the students, grouping for the project groups, there is a problem with that. Sometimes a group can have included all students as higher GPA students. Some project group can have all students as weak student. If university take the responsibility for grouping, usually the process is randomly grouping using a student list. There are some online tools which can be used for grouping. Those tools also using random group generation using a created student list [3]. But that is not effective and not fair. There are some qualified tools to do the process but it doesn’t come with project management tools. Those tools are available in standalone and also in web based applications [4]. Which leads to confusion. Because for the whole process university have to use different tools and it is bit confusing for management.

The best solution is to develop a system where project grouping can be done fairly and effectively. In a research paper which were published in 2010, It contains a solution for the grouping. It uses student’s previous compulsory professional curriculums scores to do the weighted statistical calculation according to their importance degree in the software project practical teaching, thus the professional mastery degree of student can be determined. Then, using the fuzzy clustering strategy to classified the students into different classes, based on this, doing the grouping action [5]. In this paper the scenario is different but the strategy we’re planning to do is the same. There is a research paper which they developed an open-source program, GroupEng, which assigns groups according to guidelines from education research. Guidelines include avoiding isolating women or minorities and assigning multi-disciplinary groups of mixed abilities. The program operates on a set of simple, flexible, faculty defined rules, keeps data local, and ensures “fairness” of group strengths [2]. Same procedure will to be used in this research where students will be grouped based on their skills and using their academic progress. And the major problem is how to determine the accuracy of the group formation. So we need to find an easy way to determine success rate of the implemented solution or otherwise we need to use the implemented solution for the test run for a whole project and see the end results by the end of the project. Easiest way to do is check the accuracy of expected result with the generated results of the balance between the team composition. Such as GPA and matching skills.



## Figure 1.3: Feature Comparison with the existing solutions and proposed system

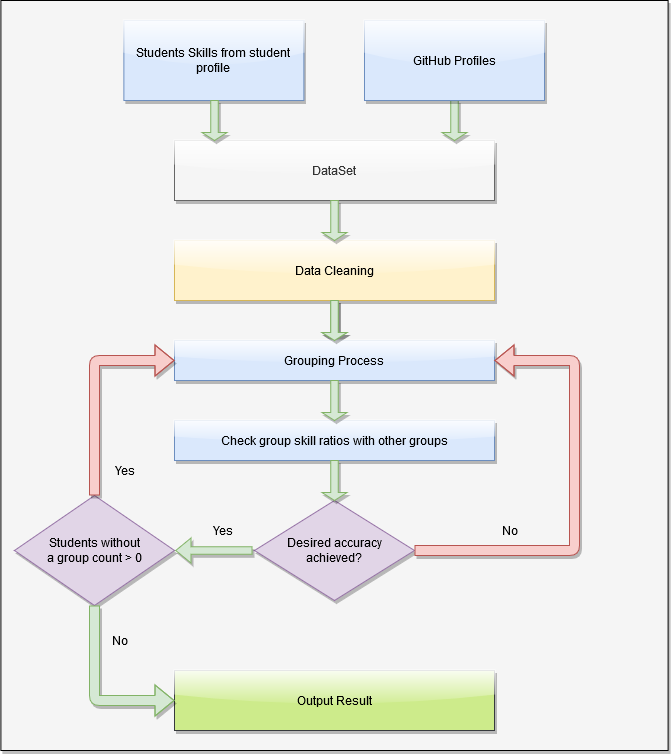
Figure 1.1 shows a table that we created by finding functionalities in the existing project management systems. The red colored cells show the functionality is not implemented in the system and green colored cells shows the functionality is implemented in the system. As you can see the grouping mechanism is not implemented in neither software solutions and it is a major requirement for any university to improve effectiveness and efficiency of university projects.

Since there is no grouping strategy implemented in current existing project management systems, Options are to use a standalone system to just for grouping requirement, or to implement a system where these required functions are integrated in to the same system so that system can automatically do the management easily.

# 2. OBJECTIVES

# 2.1 Main Objectives

The main objective of the research is to implement a system that can automate the student project group generation. Easiest way to do the grouping is the current system where grouping is depending on the student registration or random group formation. However, that strategy is efficient by not effective. Conflicts inside the team increases while doing so. So the option is to generate student groups in a well manner and fairly. So the objective is to create an Algorithm using machine learning to increase the efficiency in group generation and the effectiveness in group working environment by generate groups considering student skills.



## Figure 2.1: Grouping Algorithm flow chart

Figure 2.1 shows the grouping algorithm which to be used in our system. System will iterate the grouping process until the group composition rate of every group is at least fair. To create the system, we need to find a matching machine learning approach and to do so we need to complete some sub objective. Sub objectives are given below as specific objectives.

# 2.2 Specific Objectives

In order to achieve the main objective there are specific objective to be completed. Specific objective that need to be attained is as followed.

1. Data collection.

Data collection is a major role in the process. Data is a key resource that we need to have for the data analysis. Without sample student data, we cannot proceed to later steps. Using sample data, we can create the data cleaning processes and start the model training. For data collection we’re using google forms. By sharing google forms to the students in SLIIT we’ll be able to get enough data for training and testing data.

1. Analyze the data and find relationships between data.

After data collection we need to find relationships with data and analyze how to use them for the grouping process. Some of the fields will be only used for separate students for different sections. For an instance there are 3 types of developers. Desktop developers, Web developers, Mobile developers. By separating data set can reduce processing time.

1. Using data analysis find an optimal algorithm or algorithms to achieve the expected results.

Currently we’re planning to use fuzzy clustering algorithm for grouping process. But later we might be needing additional algorithms for additional processes. Before identifying algorithms, we need to analyze data and then we can get an idea which algorithm suitable, efficient and effective.

# 3. METHODOLOGY

This group division methodology considers 7 factors to increase the fairness in the group composition.

* Develop Technology
* Develop For
* Programming Languages
* Frontend Frameworks
* Backend Frameworks
* Database Management Systems
* Version Controlling

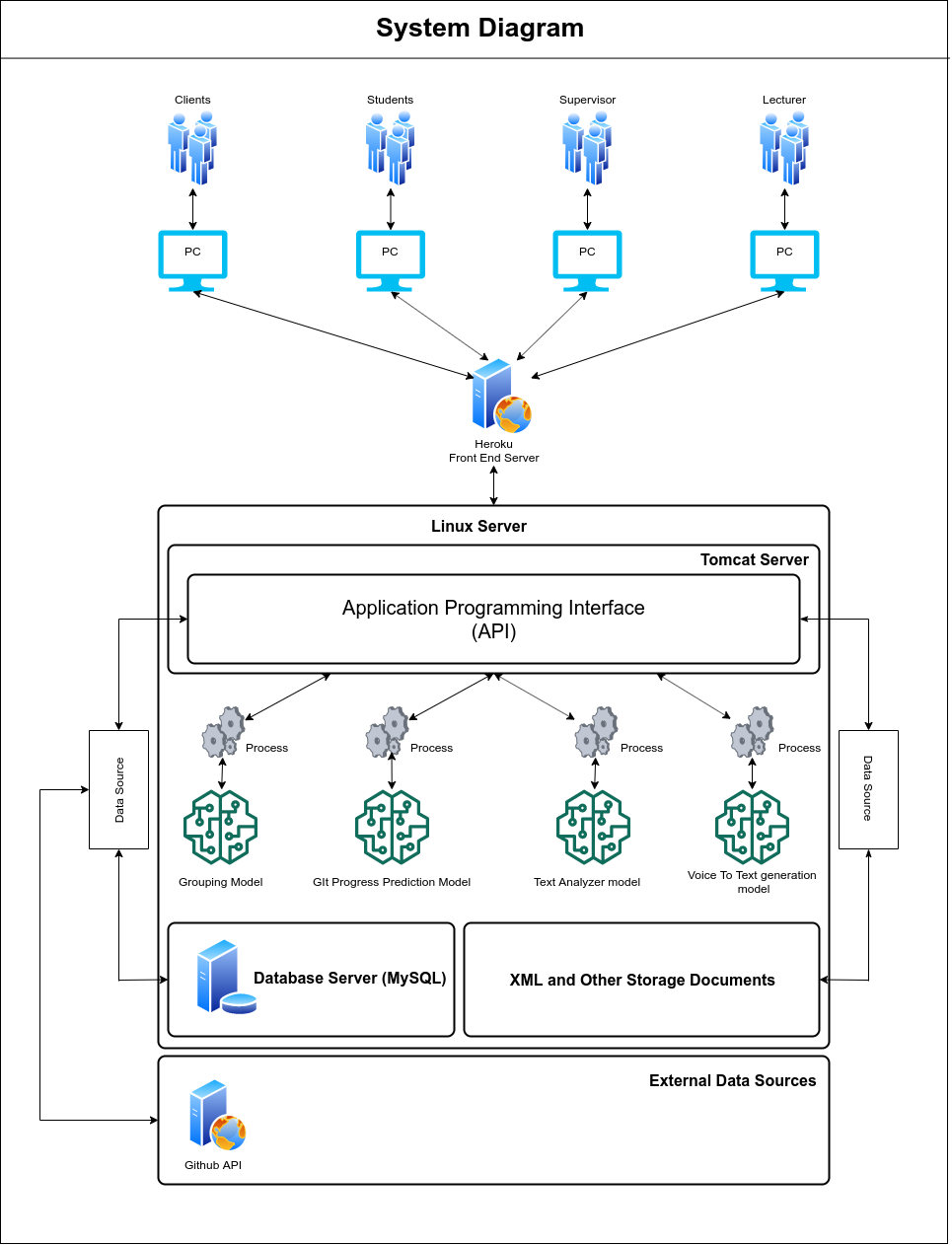
Using these factors, we can divide students into different sections. For an instance we can provide sections as Desktop Applications, Web Applications, Mobile Applications. But the theory is not to just group them separately, but to mix all sections and create a fair group composition where it is not just random but the group composition is same for all generated groups. By doing so the we can reach the maximum efficiency that we can expect from a group project. But the grouping mechanism is dependent on the student’s honesty. If a student supplies incorrect information about themselves, grouping success rate would be reduced. But it is the student’s responsibility to provide their true skills to the system. To reduce that effect, we’re using student GPA and previous project details from their GitHub repositories to gather details of programming languages that they used, so the system can match given programming skills against skills that they actually used from the GitHub repositories.

The grouping process will take time since the student count in a university is much larger. But the effective on the project success rate can be increase significantly by using a system generate group formations instead of using traditional grouping methods.

Students need to first registered to the project management systems and then students are prompted to a form where students can add their skills to their profile where it can be also modified in their skills. However, before starting grouping mechanism, students need to provide details about themselves. Such as their GPA, above mentioned skills and their existing GitHub repository or a newly created repository. Since the grouping mechanism is depending on the student details, without them the process cannot be started. If a student hasn’t supplied required details and when the grouping mechanism starts, System will generate an email and send to the student as a warning letter. If the student still didn’t supply their details the grouping process will be started but the student will be out of the process. Like mentioned before the whole grouping mechanism is depending on the student’s honesty and their dedication. Later the student can contact module lecture in charge or any other person who is appointed for the management process and request a group for them.

# 3.1 System Architecture

The system architecture is shown in the figure 3.1.



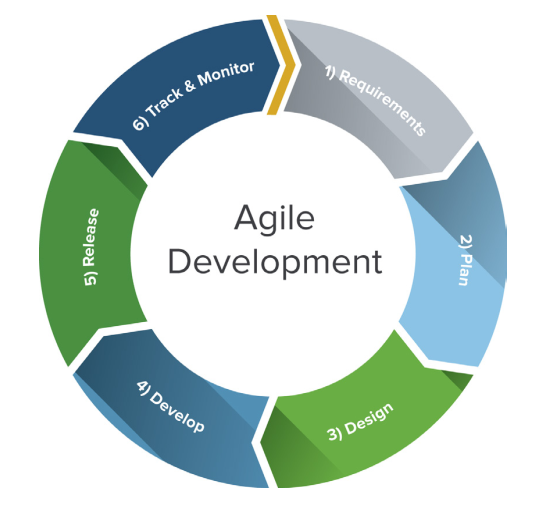
## Figure 3.1 – System Architecture Diagram

According to the system diagram as shown in Figure 3.1, 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. Frontend solution will be communicating with the users and the API.

# 3.1.1 Software Solution



## Figure 3.1.1: SDLC Life Cycle

For our solution, we’ll be using the agile methodology. The Internet economy has altered the rules of software engineering. Traditional development methodologies are too cumbersome to meet the rapidly changing requirements and short product cycles demanded by business. To meet these changing requirements, software developers have developed agile software development methodologies utilizing iterative development, prototyping, and templates [6]. Since our project’s requirements may change overtime the best software life cycle solution is the agile methodology.

* **Requirement Gathering and Analysis**

The first phase in the SDLC is the requirement gathering process. In the process we’ve gathered current issues with the project management process in the SLIIT ITP module. We’ve contacted the lecture in charge of that module to gather information about the module process and the current issues having within the management.

* + Survey Results

One of the major issue was the grouping mechanism. Usually there were conflicts between team members and the reason was the unbalanced group composition. Some groups having all the higher skill students and some of the groups having all the lower skilled students. So there is a need of a propper grouping mechanism for the project management.

* **Feasibility Study**
  + **Schedule feasibility**

Proposed system should finished in the given time period and according to the research scope, should be able to finish within the time period and from the time period we can arrange additional time period will be used for testing before demonstrations. According to the present timeline we’re few days ahead of the schedule.

* + **Economy feasibility**

Since the whole solution in deployed in a virtual environment, only for the server cost will be need to be taken care of. The solution can be also deployed in the university servers but for the security reasons, better to be deployed in a different server.

* + **Technical Feasibility**
    - Machine Learning knowledge

To implement grouping algorithms, we need to have a high level understanding on machine learning knowledge to identify algorithms and the logic resides within the source.

* + - Basic knowledge in APIs and MySQL Server

Need to have a basic knowledge in API implementations. Since the API will be developed using java, we need to have basic knowledge in java.

* + - UI and UX

Need need to have atleast minimum knowledge in Frontend Frameworks since we’re using React JS as the frontend development.

* **Design**

For the design phase, we’re focusing on 2 sides.

* + Frontend Design

We’ll be using mockflow or draw.io to desing wireframes for the system and so of the animations and transitions need to be decide using an online tool to test and explore new features for the system.

* + Backend Design

Since the as the backend we’re running an API, we need to design an path ways for the endpoints for the API. In needs a separate documentation to be maintained with the API source.

* **Implementation**

In the implementation phase, below steps will be completed to finish the implementations.

* + Implement an API to create a connection between the UI and the core system.
  + Analyze student skill data to classify student’s classes.
  + Create groups using student data and using clustering method the grouping and re-allocation students to the remaining groups.
  + Output the result to an UI, so the user can see the generated groups and the group composition rates.

After implementing the above mentioned functionalities, we’re providing an web application to interact with the system.

* + **Web Application Development**

The end result of the project will be a web application to manage projects and the integrated capability to automate the grouping mechanism using student skills. We’ll be using React JS for the web application development. React is a JavaScript library for building user interfaces [7]. It is maintained by Facebook and a community of individual developers and companies.[[8]](https://en.wikipedia.org/wiki/React_(JavaScript_library)#cite_note-4)[[9]](https://en.wikipedia.org/wiki/React_(JavaScript_library)#cite_note-5)[[10]](https://en.wikipedia.org/wiki/React_(JavaScript_library)#cite_note-6).

* + **API development**

For the API development, we’re using java with maven quarkus[11]. Maven’s primary goal is to allow a developer to comprehend the complete state of a development effort in the shortest period of time. In order to attain this goal, Maven deals with several areas of concern:

* + - Making the build process easy
    - Providing a uniform build system
    - Providing quality project information
    - Encouraging better development practices

Maven aims to gather current principles for best practices development and make it easy to guide a project in that direction[12].

* + **Database Handling**

Using java JDBC driver makes it easy to manage MySQL database with the API. JDBC Driver is a software component that enables java application to interact with the database[13]. For the database querying and configuring database structures, we’ll be using MySQL workbench. This database structure will contain Tables, Procedures and functions to reduce the code repetition.

* + **Model Training**

Model Training will be started after we collect enough data amount for the training dataset and for the testing data set. For the data gathering process we’re using google forms to collect data from the students. Once the data collection process is finished, we can start the training process to lock a model that can be use for the grouping process.

* **Testing**
  + **PostMan**

For the API testing we’re using Postman. Postman is a collaboration platform for API development. Postman's features simplify each step of building an API and streamline collaboration so the user can create better APIs faster[14].

* **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, easier, and more cost effective [15].

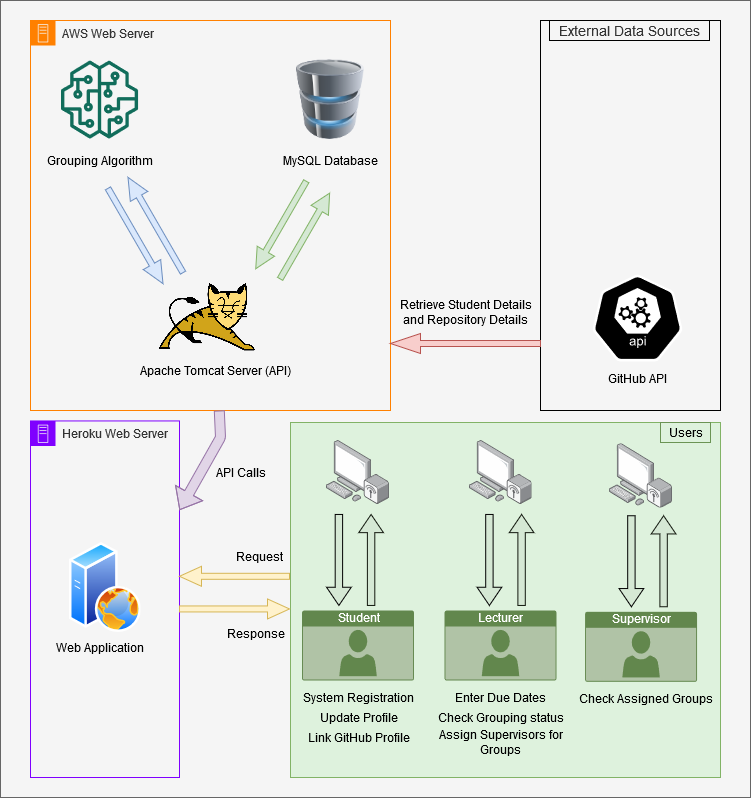
* + **Ngrok**

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.

# 4. DESCRIPTION OF PERSONAL AND FACILITIES

Grouping process will take a considerable amount of time, since the student count larger and to achieve best group composition system need to repeat the process until the groups are fairly organized. Before starting the grouping process there’re some constraints to be satisfied.

* All students should finish updating or registering before the due date passes. After the due date, system will automatically lock the profiles and until the grouping process completed students cannot update profile details.
* Lecturer in charge should add a due date for the final registration date. The date that lecturer specify, system will start grouping process.
* Before starting the process, system will gather student details and github profile details of each student to a staging environment. After aquiring all the required details, system will start the grouping process.



## Figure 4: Overall Grouping Process

For this process facilities that need to be implemented:

* Creating the model for the grouping process.
* UI components and API components for the data gathering and data recordings.

Figure 4 shows the overall process of the Grouping mechanism. In users section includes the basic operations that need to be done before the Grouping process and the next steps for the project groupings. System will use student defined skills and also existing knowledge about previous work using github integration. Using GitHub API, system can retrieve technical details about the previous projects.

# 4.1 Data Collection GitHub and Google Forms

GitHub integration will help the system to retrieve student details as well as student GitHub performance using GitHub API. Key resource for a computer based decision making system is Data. Using these data, system will generate various decisions that humans may cannot even compute. However, Without human involvement computer systems cannot proceed. Since this automatic grouping system is based on student skills, Without students, system cannot do any process. So the first step is to supply data to the system. In this instance, required data will be gathered using GitHub and user defined data. User defined data can be retrieved using google forms or while the system is running. Since the system hasn’t built yet, We need to a way to gather student data for the system.Without data system cannot run the grouping model which also generated using sample data. So the first step is a design a way to retrieve student data.

Before training the grouping model, We need large amount of data to run training program. We’ll be using google forms to retreive data from SLIIT students and then we can use retrieved data for the training model. We’ll Split dataset for 2 sections. 75% of the total records will be used for training and the other 25% will be used for testing. Using google forms, we can convert results to a excel format. So that we can directly use that dataset for data cleaning and other data analytic processes. Questionnaires that we’ve documented can be filled by any student from any university. However, Since we’re only providing this google form within the SLIIT students, the actual count of responses are considerably lower to run for model training. So we’re hoping to provide the google form so any university student can repond to it.

The questionnaire is focusing on retrieving student’s programming skills which will be used for grouping process according to student’s skills. Since this questionnaire doesn’t include specific details or questions refering SLIIT scope, We can use this form to gather data from any student who is in the same field of study.

This process only uses for the model training purpose. After a model lock, we can integrate it with the system and run an actual trail. Accuracy can be identified within the application, So the lecturer assign supervisors for each generated group.

# 4.2 Usage of gathered Data Collection

After 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.

# 4.3 Other Functions

1. User Management

This section will manage every user who interact with the system. Such as Student, Lecturer, Supervisor and Clients. This section will also manage permissions for the system users and user profiles.

1. Suggest Common Tasks for a Registered Project.

When a group register a project, System will automatically suggest common functions that can be included in to the project. However, this function improves with the system over time since the suggestion mechanism uses previous project details.

1. Auto Generate Gantt Chart

After groups define their project tasks and objective. This facility will automatically generate a Gantt chart within the given period. Period will be assigned by supervisor or lecturer in charge.

# 5. REFERENCES

[1] Chen Di and Jiang Bei-zhan, “Some Problems in Teaching ‘Educational Technology in Project Teaching’”, Journal of e-Education Research,pp.63-65 2007.

[2] Dimiduk, T. G., & Dimiduk, K. C. (2011, December). Effectively assign student groups by applying multiple user-prioritized academic and demographic factors using a new open source program, GroupEng. In 2011 WEPAN Nat. Conf. Advancing Women: Transforming Eng. Educ (pp. 1-12).

[3] Chirag Mehta, & Tamara Swedberg, Team Maker, 2000. [online]. Available: <https://chir.ag/projects/team-maker/>. [Accessed: 19-Feb- 2021]

[4] Brant Gurganus, Team Maker, 2000. [online]. Available: <https://sourceforge.net/projects/team-maker/>. [Accessed: 19-Feb- 2021]

[5] Z. Wen-qiang and W. Ben-ting, "Project practical teaching grouping strategy based on fuzzy clustering algorithm," 2010 5th International Conference on Computer Science & Education, Hefei, China, 2010, pp. 678-681, doi: 10.1109/ICCSE.2010.5593662.

[6] Livermore, J. A. (2007, March). Factors that impact implementing an agile software development methodology. In Proceedings 2007 IEEE SoutheastCon (pp. 82-86). IEEE.

[7] ReactJS. (2021). A JavaScript library for building user interfaces. [online]. Available: <https://reactjs.org/> [Accessed 21-Feb-2021]

[8] Paul Krill, “React: Making faster, smoother UIs for data-driven Web apps”, May 15, 2014. [online]. Available: <https://www.infoworld.com/article/2608181/react--making-faster--smoother-uis-for-data-driven-web-apps.html>. [Accessed: 20-Feb-2021]

[9] Hemel, Zef, "Facebook's React JavaScript User Interfaces Library Receives Mixed Reviews", Jun 03, 2013. [online]. Available: <https://www.infoq.com/news/2013/06/facebook-react/> .[Accessed: 20-Feb-2021]

[10] Dawson, Chris, “JavaScript’s History and How it Led To ReactJS”, July 25, 2014. [online]. Available: <https://thenewstack.io/javascripts-history-and-how-it-led-to-reactjs/> .[Accessed: 20-Feb-2021]

[11] Quarkus. (2021). A Kubernetes Native Java stack tailored for OpenJDK HotSpot and GraalVM, crafted from the best of breed Java libraries and standards. [online]. Available: <https://quarkus.io/>. [Accessed: 21-Feb-2021]

[12] The Apache Software Foundation. (2021). What is maven. [Online]. Available: <https://maven.apache.org/what-is-maven.html>. [Accessed: 21-Feb-2021]

[13] Javatpoint. (2018). JDBC Driver. [online]. Available: <https://www.javatpoint.com/jdbc-driver>. [Accessed: 21-Feb-2021]

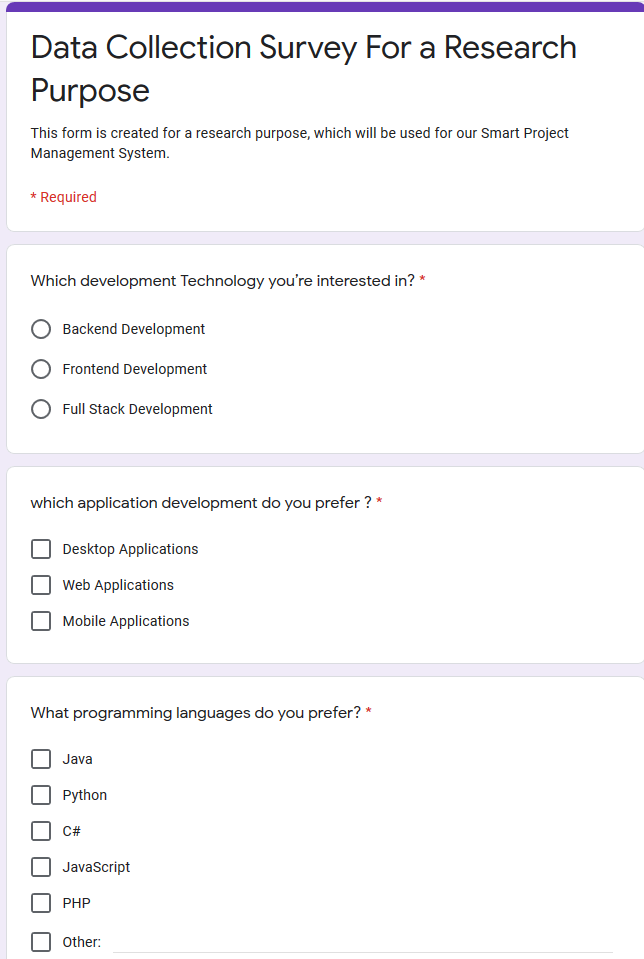
[14] Postman. (2021). The Collaboration Platform for API Development. [online]. Available: <https://www.postman.com/>. [Accessed: 21-Feb-2021]

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

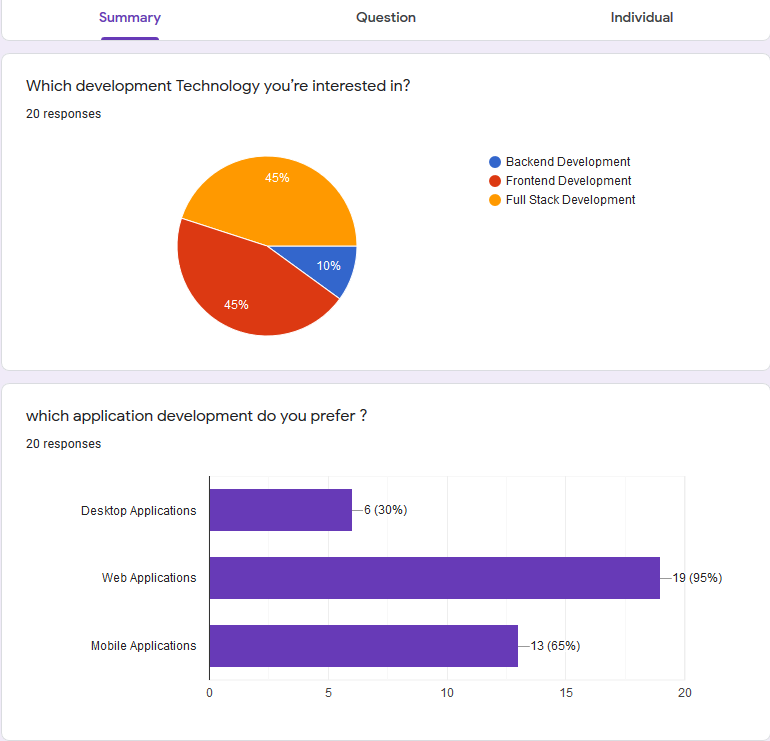
[16] Srba, I., & Bielikova, M. (2014). Dynamic group formation as an approach to collaborative learning support. IEEE transactions on learning technologies, 8(2), 173-186.

# 6. APPENDICES

# Appendix – A: Sample questionnaire



# Appendix – B: Sample questionnaire response



# Appendix – C: Gantt Chart



Appendix – D: Plagiarism Report

