Table of Contents

[Preface 6](#_Toc70223464)

[1. Introduction 7](#_Toc70223465)

[1.1. Project background 7](#_Toc70223466)

[1.2. Project aim and objective 7](#_Toc70223467)

[1.3. Project development methodology 8](#_Toc70223468)

[2. Requirements Engineering 9](#_Toc70223469)

[2.1. Elicitation activities 9](#_Toc70223470)

[2.1.1. Interview plans 11](#_Toc70223471)

[2.1.1. Other problem domain research 24](#_Toc70223472)

[2.1.1.1. Comparable software system review 24](#_Toc70223473)

[2.1.1.1.1. Edmodo 24](#_Toc70223474)

[2.1.1.1.2. Canvas 26](#_Toc70223475)

[2.1.1.1.3. Schoology 27](#_Toc70223476)

[2.1.1.2. Development relevant legislation 28](#_Toc70223477)

[2.1.1.2.1. The computer misuse act 1990 28](#_Toc70223478)

[2.1.1.2.2. The copyright, design and patent act 1988 28](#_Toc70223479)

[2.1.1.2.3. Data protection act 2018 29](#_Toc70223480)

[2.1.1.3. Any other relevant problem domain investigation data 29](#_Toc70223481)

[2.2. Requirements specification 29](#_Toc70223482)

[2.2.1. Problem domain description 29](#_Toc70223483)

[2.2.1.1. Existing business operations 30](#_Toc70223484)

[2.2.1.1.1. Enrolling a student to the university 30](#_Toc70223485)

[2.2.1.1.2. Taking attendances 31](#_Toc70223486)

[2.2.1.1.3. Submitting assignments and giving feedbacks 32](#_Toc70223487)

[2.2.1.1.4. Scheduling PAT day 33](#_Toc70223488)

[2.2.1.2. Summary of existing business operations 33](#_Toc70223489)

[2.2.2. Functional requirements 34](#_Toc70223490)

[2.2.2.1. Required features of the system 34](#_Toc70223491)

[2.2.2.1.1. Database and backup 34](#_Toc70223492)

[2.2.2.1.2. Students records 34](#_Toc70223493)

[2.2.2.1.3. Staff records 34](#_Toc70223494)

[2.2.2.1.4. Courses and modules records 35](#_Toc70223495)

[2.2.2.1.5. Management of module 35](#_Toc70223496)

[2.2.2.1.6. Management of assignments 35](#_Toc70223497)

[2.2.2.1.7. Management of attendance 35](#_Toc70223498)

[2.2.2.1.8. Management of personal tutor 35](#_Toc70223499)

[2.2.2.1.9. Management of the grades 35](#_Toc70223500)

[2.2.2.2. Access rights/Level of access 35](#_Toc70223501)

[2.2.2.2.1. Administrator, staff & student management 36](#_Toc70223502)

[2.2.2.3. Output formats for onscreen displays 46](#_Toc70223503)

[2.2.2.3.1. Email sent after student registration 46](#_Toc70223504)

[2.2.2.3.2. Email sent after publishing course content 47](#_Toc70223505)

[2.2.2.3.3. Email sent after publishing grades 47](#_Toc70223506)

[2.2.2.3.4. Email sent after publishing assignment 48](#_Toc70223507)

[2.2.2.3.5. Email sent after submitting assignment 48](#_Toc70223508)

[2.2.2.3.6. Email sent after PAT allocation 49](#_Toc70223509)

[2.2.3. Performance requirements 49](#_Toc70223510)

[2.2.3.1. Speed 49](#_Toc70223511)

[2.2.3.2. Capacity 50](#_Toc70223512)

[2.2.3.3. Reliability 50](#_Toc70223513)

[2.2.3.4. Usability 50](#_Toc70223514)

[2.2.3.5. Security 50](#_Toc70223515)

[2.2.4. Design constraints 50](#_Toc70223516)

[2.2.5. Commercial constraints 51](#_Toc70223517)

[3. System analysis and design 52](#_Toc70223518)

[3.1. Preliminary design stages 52](#_Toc70223519)

[3.1.1. Textual Analysis 52](#_Toc70223520)

[3.1.2. Significant event analysis 53](#_Toc70223521)

[3.1.3. Command queries and constraints 63](#_Toc70223522)

[3.2. Detailed Static / Dynamic System Designs 65](#_Toc70223523)

[3.2.1. First Draft BON System Architecture Diagram 67](#_Toc70223524)

[3.2.2. BON System Chart 69](#_Toc70223525)

[3.2.3. BON Cluster Charts 70](#_Toc70223526)

[3.2.4. BON Class Charts 70](#_Toc70223527)

[3.3. Data flow Diagram 71](#_Toc70223528)

[3.3.1. Context level 71](#_Toc70223529)

[3.3.2. First level data flow diagram 72](#_Toc70223530)

[3.3.3. Activity diagram 75](#_Toc70223531)

[3.3.3.1. Login activity diagram 76](#_Toc70223532)

[3.4. System database design 77](#_Toc70223533)

[3.4.1. E-R model 77](#_Toc70223534)

[3.4.2. Sequence diagram 77](#_Toc70223535)

[3.4.3. Attribute listing 80](#_Toc70223536)

[4. System interface design 81](#_Toc70223537)

[4.1. Draft Interface Designs 81](#_Toc70223538)

[4.1.1. Wireframes 81](#_Toc70223539)

[4.1.2. System Navigation Diagram 87](#_Toc70223540)

[4.1.3. System Screen Mock-ups 89](#_Toc70223541)

[4.1.4. System Activity Event Diagrams 97](#_Toc70223542)

[4.2. Design Revisions 99](#_Toc70223543)

[4.3. Heuristic Evaluation 100](#_Toc70223544)

[7. System evaluation 100](#_Toc70223545)

[7.1. Usability evaluation strategy 100](#_Toc70223546)

[7.2. System plot trails 101](#_Toc70223547)

[7.3 System Trial Results 105](#_Toc70223548)

[7.4 Project Work Log 106](#_Toc70223549)

[8. Project Conclusion 109](#_Toc70223550)

[9. Project Management 110](#_Toc70223551)

[9.1. Work Breakdown Structure (WBS) 110](#_Toc70223552)

[9.2. PERT (Program Evaluation and Review Technique) chart 111](#_Toc70223553)

[9.3. Gantt Chart 113](#_Toc70223554)

[9.4. Project Meetings 114](#_Toc70223555)

[9.5. Project Quality Plan 120](#_Toc70223556)

[10. References 122](#_Toc70223557)

# Preface

We were given a task to develop an electronic Course Management System for Woodlands University College. Woodlands University College is an educational institute that offers various degrees. The university has been facing a lot of troubles due to its existing paper-based system. The course leader of the university whose name is Mr. White had come up with the idea of changing the paper-based system with computerized system. Mr. White discussed the chances of changing the system with us.

Ourfocus initially was to pilot system for computing department only. We started off by choosing our programming language and framework to be used in the project. We decided to go for JavaScript with **React** and **Nodejs** framework for backend.We used **MySQL** for database. An interview session was held with the key stake holders of the university: Mr. Mark Williams, Dr.Simon White, Dr.Raj Singh, Mr.Adam Blake to gather all the requirements and functionalities for the project. We studied all the functional requirements and did some research on other existing comparable system like **Edmodo, Canvas and Schoology.** We studied how course management systems work and must have functionalities and about the issues that we do not want our system to have. We then decided to work on implementation part. We have used Visual studio IDE for coding. Some GitHub repositories were created so that we could work on different parts of system in a more systematic way. Restful APIs for all the modules were created and the further tested using **postman** tool. The UIs for all the required modules were created using HTML and CSS.

There were some issues and some mistakes that we made while developing the system but eventually we solved all of them. Overall, it was a great learning experience on how to work together as a team along with time management.

# Introduction

In this year 2020, a huge pandemic brought a lot of trouble in almost all the sector. Education sector was also badly got affected. So, many things are going or running online from home and so the colleges

All the colleges are adopting online based classes, assignments because of this pandemic (COVID 19)

Many colleges are facing problem on how to conduct it more efficiently through online medium without putting students in trouble and difficulty. So because of this computerized course management system had to arrive. Many colleges, universities are now adopting computerized course management system for almost course.

Woodland University College is a college who also wants to adopt this system for their IT and IT related students. They are facing a huge problem and wanting to give access to students online. They want software that could manage all their records of college, staff, course, student, module, assignment, attendance, timetable diary, report. They need a good team who could fulfill their requirements.

This report's main objective is to keep records of all the methods which are adopted for the development of that software. In this report, everything about the software is mentioned, from initial to final stage. This report so is divided into many respective heads.

The software is being made from React, Node and firebase which are the leading programming language in the field of websites. Object Oriented Approach is being taken. MYSQL in the other hand is used for database.

## Project background

The main purpose of making this project is to digitalise college's existing system. Woodland University College

is a small college for taking higher education which has a lot of degrees to give to the students who are in need.

Dr. Simon White is the course leader and wants the software for computing related students which could handle all the all the current necessities of the college in an online way. In this pandemic and digitalized time, they are cutting their existing clerical system and are adopting the digital system.

## Project aim and objective

The aims and objectives of the project are listed below:

1. To handle attendance of students and staffs in an online way.

2. To keep the information of college, students and staffs in an online way.

3. To handle all the assignments, grades, exam, online classes and results in an online way.

4. To have all the study materials for the students in an online way.

5. To help student in maintain diary online.

6. To have a section that gives information about the schedule of classes.

7. To eliminate problem in pandemic and digitalize the system.

8. TO eliminate clerical system.

Here are some of the salient characteristics of this software:

1. Handles attendance of both students and staffs.

2. Handles all the information of college, students and staffs.

3. Handles assignments, grades, exam, online classes, results.

4. All the required study materials are available inside.

5. Students are given a feature of maintaining diary.

6. There is a section which informs about the schedule of classes.

## Project development methodology

To make the project we adopted some methods so that we can complete it in a easy and authentic manner. Firstly, we made a logo for our group and named our group "ACE". Then we talked with the respective persons on how we can make this project special and mind-blowing. We even questioned them, talked about the problems and solutions. After collecting all those things then we all started to make it happen. We used the tool **VSCODE** to write all our codes.

VSCODE is a development tool used for developing mobile application, desktop application, web services etc.

We have used **React** and **Node-Js** for backend. We had used **GitHub** for our overall group collaboration. Doing this we handled all our day to day work efficiently.Then, we started to build it We showed them our **wireframes** and **mockups**. They even gave us some suggestions then after finishing all those we started programming and building the software that they were waiting for their college. We did all testing using the **black box** and **white box** testing methods. We made sure that our work is working without any error and is now perfect to deploy. Then we **deployed** it to the server. And after finishing we validated, presented and handed over to them.

Creating logo and team’s name

Collecting relevant information

Handing over to the clients

Project Development Methodology

Choosing the tools, programming languages

Deploying, Validating and Testing the system

Making Wireframe and doing discussion

Building the system with all the necessary features

# Requirements Engineering

This part of report consists of all the necessary questions and information which were needed to make the system was illustrated. This helped us to understand and solve many problems.

## Elicitation activities

|  |  |
| --- | --- |
| Problem Domain Entities | Elicitation Strategy |
| Dr. Simon White  (Computing course leader) | 1. Do you want the courses to be translated to multiple languages? |
| 2. How often will the course contents change? |
| 3. Were there any e-learning training programs conducted before? |
| 4. Do you want a separate central portal where all of the courses can be accessed? |
| 5. What kind of content will people add? |
| 6. Do you need any password protected areas? |
| 7. Are there any technology constraints? |
| 8. Should we focus more on ease of implementation or really cool features? |

|  |  |
| --- | --- |
| Problem Domain Entities | Elicitation Strategy |
| C) Dr Raj Singh | 1) As a module leader, what types of files do you share the most and the least with students? Also, which file formats are used? |
| 2) Currently, which platform do you use to share files? Should the new system include file sharing feature? |
| 3) Do you need to assign multiple assignments in a module or is it a single assignment at a time? |
| 4) As a personal tutor, what are your responsibilities? |
| 5) Do students need to share/submit some videos to the college? |
| 6) Should a third party video service like Kaltura be integrated in the site or should we build such service form scratch? |
| 7) Should there be a way to directly communicate with the parents through the website? |
| 8) How do students pay their fees currently? |
| 9) Should there be a way to pay fees through the website? |

|  |  |
| --- | --- |
| Problem Domain Entities | Elicitation Strategy |
| C) Mr Adam Blake | 1) What cost and time you want to allocate for the completion of the project? |
| 2) What are the most common issues that students or tutors bring to the support team? |
| 3) Currently, how are the courses being managed? |
| 4) What information do you store about students and tutors? |
| 5) Can you describe what do you do as a administrator in the computing department? |
| 6) Can you elaborate the work and the limitation of admin and users? |
| 7) You beinganadministator, how many sub admins do you want to be in this system? |
| 8) What functionalities do you want to see in this system? |
| 9) Do you want parents to have a user and see daily records of college and students? |

|  |  |
| --- | --- |
| Problem Domain Entities | Elicitation Strategy |
| Mr Mark Williams | 1) What makes Woodlands University College different from others? |
| 2) Which features should be the most accessible for students? |
| 3) How often do you use a feedback system? And what information should be included in feedback? |
| 4) Taking about privacy, what are your views on sharing personal information like email address, phone numbers with course |
| leaders, personal tutors and other colleagues? |
| 5) What are your views on some exams being integrated on the website itself? |
| 6) What is the most irritating aspect of the current system from a student's perspective? |
| 7) How do you currently submit your assignments and recieve your grades? |
| 8) Have you used any educational portal before? Do you require any sort of training to use the software? |
| 9) Would you like to be notified about every small details through an automated email or will that be irritating? |

## 2.1.1. Interview plans

Interview Title: Initial Interview with Computing Course Leader-Dr. Simon White

Interview Date: 31st January, 2021

Duration: 1 Hour

Person in attendance:

* Kushal Upreti
* Prawesh Gautam
* Safal Sharma
* Salina Khadka

|  |  |  |
| --- | --- | --- |
| Topic Area | Motivation and Client Constraints | |
| Interviewer | Question Number | Question-Client Response |
| Salina Khadka | 01 | Do you want the courses to be translated to multiple languages?  Answer:  For now let’s stick with only one language that is English in the first version. So let’s not worry about other languages in the first version. We will think about it in the future. |
| 02 | How often will the course contents change?  Answer:  Normally, frequent changes will not happen in the course. Courses will be revised but minor changes will take place in annual basis but besides that major changes will not happen every year. So, the courses will take 2-3 years to be revised in a major way. |
| 03 | Were there any e-learning training programs conducted before?  Answer:  Not yet. Actually, we are using a kind of semi automated mechanism. So, we have not organized any kind of specific training that you are seeking for so far. |
| 04 | Do you want a separate central portal where all of the courses can be accessed?  Answer:  Yes that will be good. I will not put it as a must have item but I will consider it as optional item. It is wished to have item. |
| 05 | What kind of content will people add?  Answer:  People will not be able to change any content without any authentication. The general public should not have the access to modify the data. They should have access only in read only or view only mode. |
| 06 | Do you need any password protected areas?  Answer:  You can utilize this feature in the part of role. Normally there will be student role, module leader role, course tutor role and administrator role in which you can specify the workspace where they can put the password to get activated. |
| 07 | Are there any technology constraints?  Answer: In terms of language I don’t have particular restrictiongiving it to you guys. So you can choose any particular language in which you are comfortable .I would like you guys to provide the pros and cons of this particular language so that it will be easy for me to figure out the pros and cons and the things that I should consider while putting this system in the maintenance mode. So that is one thing and another thing is regarding the infrastructure and any other licenses, you can send the list of solution that can be software or infrastructure along with its cost according to which I will choose which option is suitable for me. |
| 08 | Should we focus more on ease of implementation or really cool features?  Answer:  Since this is the first version do we should focus more on functionality .The major functionalities should be available in the first version and we should also try our level best to minimize the issue that can be major or minor bug. So in the version we will prioritize the functionalities and the rest of the items like UI and user experience comes after the functionalities. |

Interview Title: Initial Interview with Computing Course Leader-Mr. Adam Blake

Interview Date: 31st January, 2021

Duration: 1 Hour

Person in attendance:

* Kushal Upreti
* Prawesh Gautam
* Safal Sharma
* Salina Khadka

|  |  |  |
| --- | --- | --- |
| Topic Area | Motivation and Client Constraints | |
| Interviewer | Question Number | Question-Client Response |
| Kushal Upreti | 01 | What makes Woodlands  University College different from  others?  Answers:  Woodlands college has set benchmark for  excellence in all academic disciplines through  an innovative, challenging and closely  considered curriculum, tempered with a  comprehensive sports timetable, and the  opportunity to express creativity in enriching  arts and culture activities. |
| 02 | Which features should be the most  accessible for students?  Answer:  The main features should be the easy access  to grades, assignments and class timetable. In  addition to that, practice tests, e-library and  online extra classes should be a nice addition. |
| 03 | How often do you use a feedback  system? And what information should  be included in feedback?  Answer:  I use a feedback system quite often in sites like  google services and small-scale websites to  help them grow as a platform. I think such a  system will be a nice addition in our university  portal. |
| 04 | Taking about privacy, what are  your views on sharing personal  information like email address, phone  numbers with course leaders,  personal tutors and other colleagues?  Answer:  The college already has access to ourinformation that we filled during our admissions.  So, the usage of the same information in the  system shouldn’t be a problem. |
| 05 | What are your views on some  exams being integrated on the  website itself?  Answer:  It would actually be a really useful feature to be  able to give exams directly from the website  itself. For exams that are in MCQ fashion or for  mock exams it is a viable option. Physical  disruption like power cuts or weak internet  connection or lack of access to proper device  should be kept in mind as well. |
| 06 | What is the most irritating aspect  of the current system from a  student’s perspective?  Answer:  Currently we receive our grades and  assignment in paper-based form and the  university mails are also unmanaged. The class  timetable keeps changing frequently so it is  hard to keep track of the actual routine. |
| 07 | How do you currently submit your  assignments and receive your grades?  Answer:  As I mentioned before, we currently receive ourgrades in paper-based form in report cards or  in lists pasted on the notice board. In case of  assignments, sometimes they are painstakingly  distributed through pen drive or through mail. |
| 08 | Have you used any educational  portal before? Do you require any  sort of training to use the software?  Answer:  I’ve used platforms like udemy, coursera and  edex in the past so I don’t think I need any kind  of training to use the platform. |
|  | 09 | Would you like to be notified aboutevery small detail through anautomated email or will that be  irritating?  Answer:  I think there should be an option to select a  category of emails that users want to receive.  There should also be an option to unsubscribe  from a mailing list. |

Interview Title: Initial Interview with Computing Course Leader-Mr Raj Singh

Interview Date: 31st January, 2021

Duration: 1 Hour

Person in attendance:

* Kushal Upreti
* Prawesh Gautam
* Safal Sharma
* Salina Khadka

|  |  |  |
| --- | --- | --- |
| Topic Area | Motivation and Client Constraints | |
| Interviewer | Question Number | Question-Client Response |
| Safal Sharma | 01 | As a module leader, what types of files do you share the most and the least with students? Also, which file formats are used?  Answer:  Normally as a module leader, we get access to module resources at first including 'PDF' files, "DOC" files and maybe some of the  pictures and videos as well and it totally depends on the module. The most preferred format is PDF and DOC and the least preferred  format is audio and video files. |
| 02 | Could you go through the process of assigning an assignment for a module?  Answer:  In the very beginning, a module leader will take a responsibility to take the class for the semester. Then they will review course  materials and might add scheduled assignments too, which will be visible later to the students. Similarly, they can also assign  assignments at any given time. The students should get a notification via an email when new assignments are visible.  The assignment will include document files to download as well as the links to reading materials to prepare. Finally, they will  upload the assignment work in the system. Once all the students submit their assignments within the due date, the module leader will  be able to view and mark each works and publish the results. Students should only be able to view their own grades, and not others. |
| 03 | Do you need to assign multiple assignments in a module or is it a single assignment at a time?  Answer:  We might have multiple active assignments at a time. |
| 04 | As a personal tutor, what are your responsibilities?  Answer:  At the very beginning, the admin will assign the personal tutor for each students. Tutor will create the classroom in which the students  are enrolled. After that, regular learning activities are conducted including assigning assignments, managing attendance and  also the option to record the session and so on. |
| 05 | Do students need to share/submit some videos to the college?  Answer:  There is no need to submit videos by students to the college at the moment. So, we can think about the functionality in later  versions of the system. |
| 06 | Should there be a way to directly communicate with the parents through the website?  Answer:  In the first version, let's not bring this feature. For now, the system will only interact with tutors, students and the course administrators.. |
| 07 | How do students pay their fees currently?  Answer:  Currently, students pay their fees through bank and then they submit their bank voucher to the college. Students can also visit  the college and pay the fee. |
| 08 | Should there be a way to pay fees through the website?  Answer:  There is no need to include payment integration for the first version of the system. |

Interview Title: Initial Interview with Computing Course Leader-Mr. Adam Blake

Interview Date: 31st January, 2021

Duration: 1 Hour

Person in attendance:

* Kushal Upreti
* Prawesh Gautam
* Safal Sharma
* Salina Khadka

|  |  |  |
| --- | --- | --- |
| Topic Area | Motivation and Client Constraints | |
| Interviewer | Question Number | Question-Client Response |
| Prawesh Gautam | 01 | Question: What cost and time you want to allocate for the completion of the project?  Answer: The project owner just wants a digital school management system. They havent allocated  any money for it. They are ready to pay any amount of money needed. All they wanted is a complete  all rounded digital system. |
| 02 | Question: Currently how are the courses being managed ?  Answer: The courses has been managed by the respective course leader and they are managed by  the course leader. All the courses are being taught as per the rules and guidance of the university |
| 03 | Question: What information do you want to store about students and users?  Answer: We don’t want to share any personal information of students and users. We just want to store  the academic informationaboutboth of them. |
| 04 | Question: Can you describe what you do as a administrator in computing department?  Answer: As an administrator in computing department, I manage all the basic need required for like providing  slides, contacting to counter-part, handling all the teachers and solving their problem. I am responsible  for the integral part of the college. |
| 05 | Question: Can you elaborate the work and limitations of admin, subadmin and users?  Answer: Admins, subadmins and users are the three people that will be engaged in this computer management system. Admin will be  autorised for the different high level tasks and it will be the respective program leader. And subadmins will be able to check the  day to activity of students, slides, exam, publishing results, attendance Where as the students who are the users will only allow to  access simple level tasks. The tasks which are related to them. They wont be provided access  to many things of the admin and subadmins. |
| 06 | Question: You being the admin, how many subadmin do you want?  Answer: Well, me being an admin, I want all the course teachers to be the subadminsin order to help me and help the students to get  their job done. And they will do their respective tasks as allocated. |
| 07 | Question: What functionalities do you want to see in this system ?  Answer: Since we are making an online school management system, we dont want to compromise any thing in our system. For that you can  see some of the examples of the different universities and colleges. It should perform all the possible things in an electronic way for  example classes, attendance, courses, exams, tests and many others. We want a overall good management system on which we can be proud. |
| 08 | Question: Do you want parents to have a user and see daily records of college and students?  Answer: No, idont think it will be a good idea because parents will not be convienced by looking on the internet. As they want to  know about overall thing of their child, it is better that they meet us personally and we will talk share them their required information.  Also many of the parents of our students are uneducated and they might not find it useful. |

## Other problem domain research

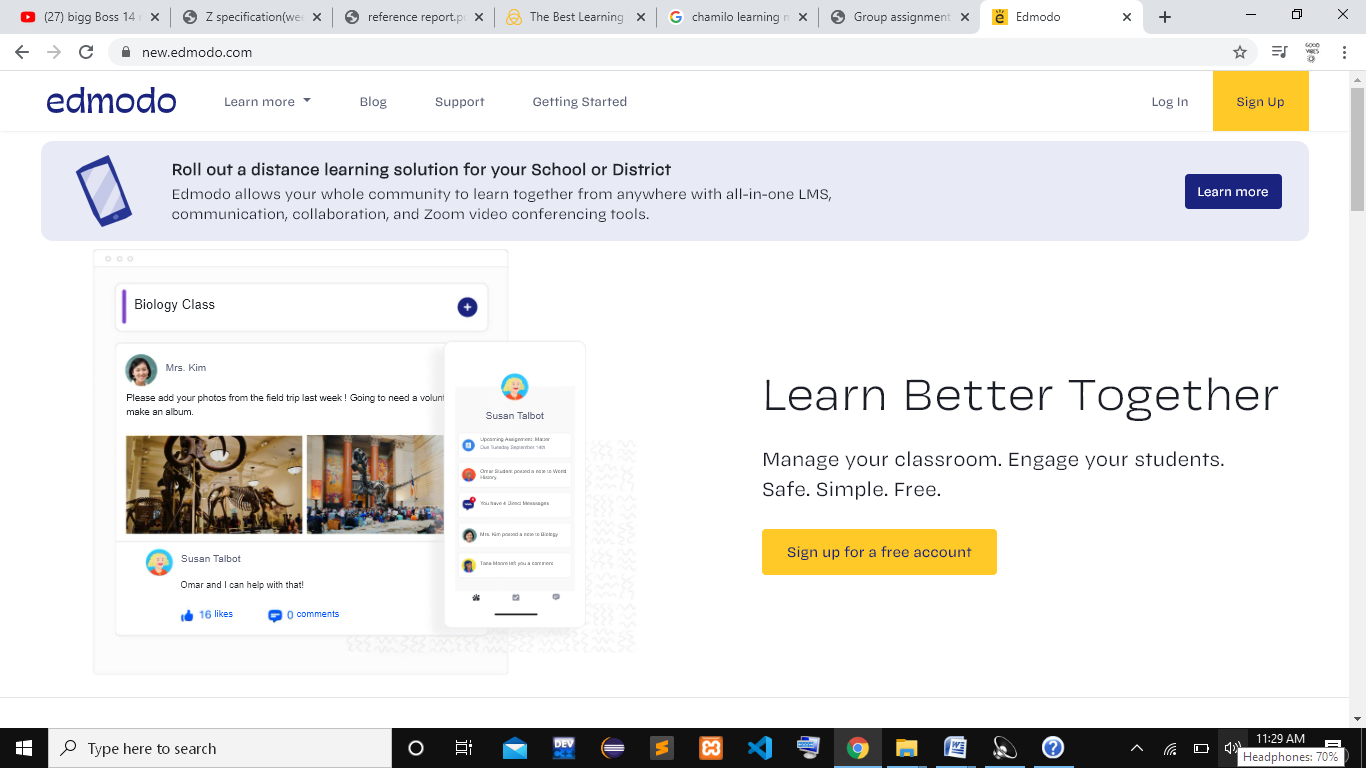
For solving the problem domain of our system, we have used the following techniques.

## Comparable software system review

We did some research on different course management system and shortlisted few of them which helped us to figure out how the system should work. We reviewed few learning management system which were quite similar to the system that we were designing.

## Edmodo

Edmodo is an online learning management system. Edmodo makes it easy for students and teachers to connect with each other with its brilliant features. Edmodo is considered to be one of the best learning management systems that is being utilized by students and instructors of different universities.



*Fig 1.1 Homepage of Edmodo*

**Some of the pros and cons of using Edmodo are listed below.**

**A) Pros:**

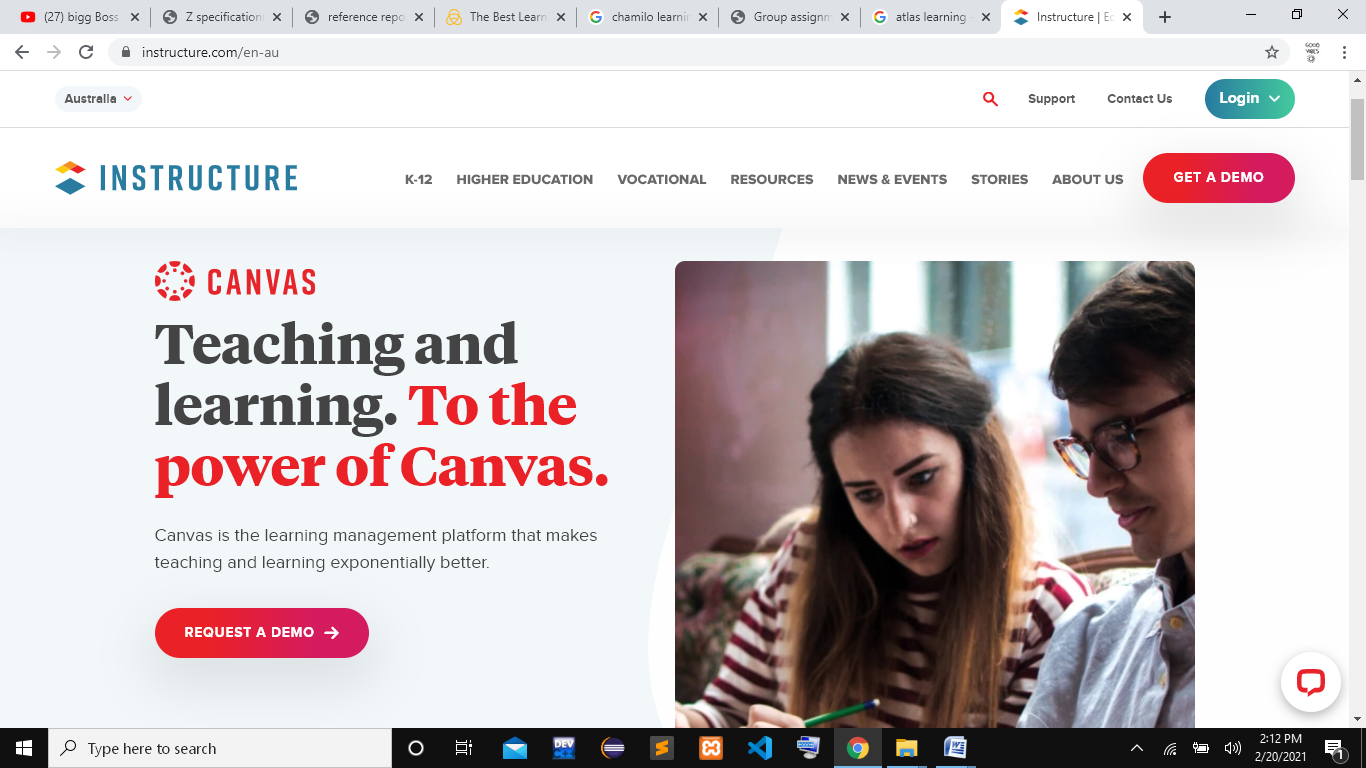
* The user interface of Edmodo is user friendly as it is based on the UI of Facebook and LinkedIn.
* It facilitates communication between teachers, students and parents.
* There is a planner feature which can be used by the students to prioritize their work.
* It is free of cost.
* It provides easy signup and facilitates third party like Gmail.
* Edmodo provides its users with a huge resource library with lots of contents which makes it easier for students and teachers to choose.
* It also facilitates online tests and online grading feature.

**B) Cons:**

* There is no virtual chat feature hence communication is not quite interactive.
* Too many irrelevant advertisements pop ups.
* The website tends to crash sometimes.

## Canvas

Canvas is a course management system. It is a well built CMS that facilitates virtual learning and teaching. It is used by universities and colleges all around the world as one of the most reliable course management systems.



*Fig 1.2 Homepage of Canvas*

**Some of the pros and cons of using Canvas are listed below.**

**A) Pros:**

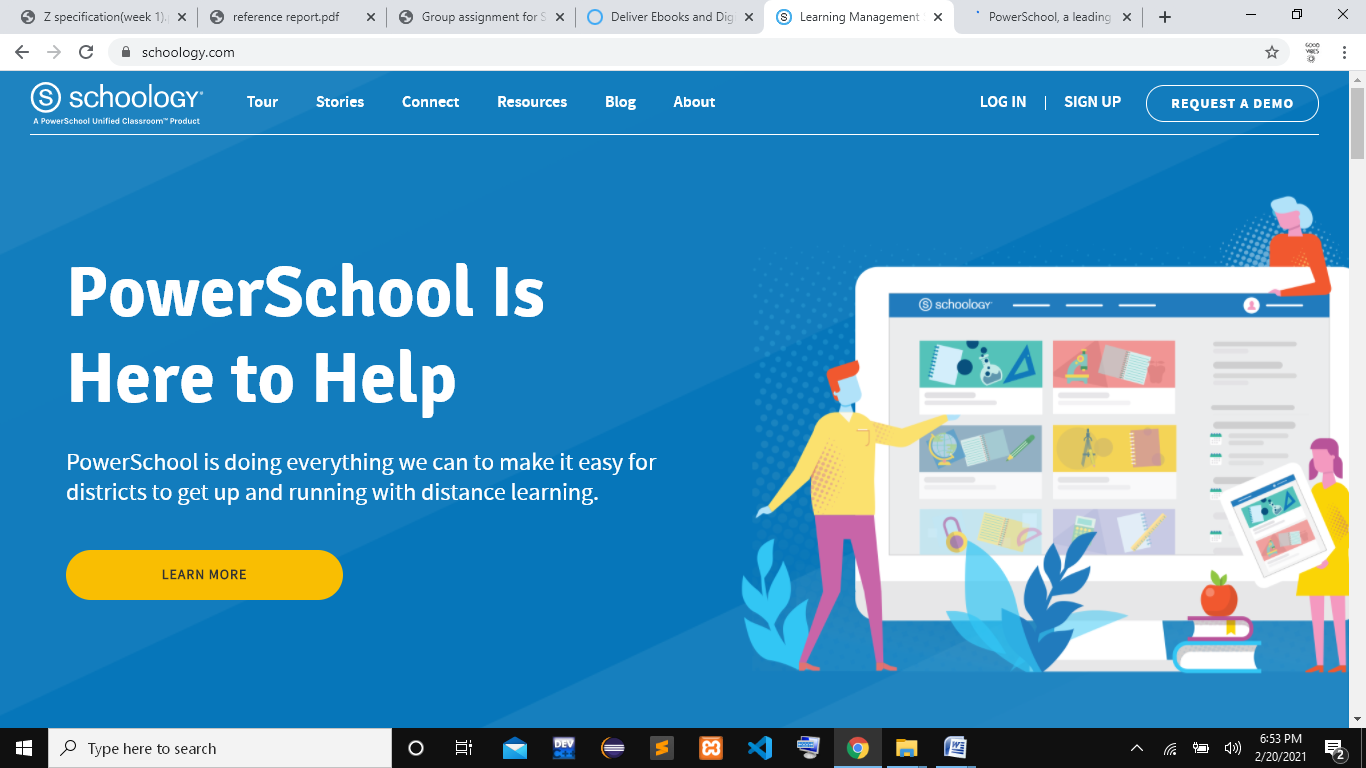
* Canvas is very simple to use as it provides user friendly user interface.
* The course management can be done easily with its brilliant features.
* It has well-defined API which makes information sharing and database access easier.
* It has smooth navigation between pages.
* Chat feature is available.

**B) Cons:**

* The interface is student centric.
* Some technical issues include delayed notifications.

## Schoology

Schoology is an online course management system which makes it easier for its users to have access to course materials anytime and anywhere. It is utilized my several education institutes to create manage and share academic contents.



*Fig 1.3 Homepage of Schoology*

**Some of the pros and cons of using Schoology are listed below.**

**A) Pros:**

* It is very simple to use so no training is required.
* Assignments can be created as well as tracked by the instructors.
* Course materials are well managed which includes create, delete, update operations.
* Attendance feature is also available.
* Courses are assigned to different categories which make it easier to search for them.

**B) Cons:**

* It has poor user interface.
* Some technical glitches include limited admin privileges.

## Development relevant legislation

Any kind of work we perform must abide by a particular set of rules and regulations. Similarly, while developing software, the software must follow some guidelines to achieve quality control in the system. One of the main aspects to develop quality software is to maintain its accountability. Hence, developers must make sure that they follow all the guidelines while developing software in order to protect their work. If we breach the development related acts and do not follow the guidelines while developing software we may end up losing our work and face several consequences.

We have listed some of the legislative acts of United Kingdom that prevents our software from unnecessary copyright issues, piracy, and spying from hackers.

* The computer misuse act 1990
* The copyright, design and patent act 1988
* Data protection act 2018

## The computer misuse act 1990

The electronically stored data have high risk of being misused. This might result in data corruption either accidently or deliberately. The computer misuse act prevents our personal data from being modified or deleted through unauthorized access. This refers to entering to a computer system to steal protected data or to cause damage to the device with the help of virus. This act prevents hacking of computer system which covers the introduction of electronic vandalism and stealing of data via hacking. If an individual fails to follow the guidelines of this act he or she will be subjected to fines or imprisonment.

## The copyright, design and patent act 1988

Copying of ideas from original creator of a work is very common especially in digital world. This is considered to be one of the serious crimes but is prevented by the copyright, design and patent act 1988.This act provides original makers with the exclusive right to control how their work is used and distributed. Copyright can be referred to as a property hence it can be sold or transferred .The person who has or buys the copyright of a work has right to replicate the work, issue or lend copies to others, broadcast the work in any platforms and adapt the work in their own way but he or she may be subjected to copyright infringement if any of the mentioned ac t is performed without the permission of copyright owner.

## Data protection act 2018

Important pieces of information that are stored by a business or an organization like customer records and staff records that can be stolen by third party for phishing scams. The data protection act 2018 prevents the key pieces of information from being misused by ensuring that an individual has access to their personal data and can modify it according to their necessity. It provides organizations guidance on how to utilize personal data by maintaining its privacy and integrity. If an organization violates data protection law, it has to face serious consequences.

## Any other relevant problem domain investigation data

We studied various other clerical management systems of other universities as well and extracted some common problems that the universities have to face on daily basis.

* Admission and enrollment process is long and time consuming
* Lack of creativity in teaching style among teachers
* Lack of storage space and inefficiency in work in paper based system
* Difficult to maintain attendance records
* Maintaining paper based system is expensive.
* Courses and schedules are difficult to manage.
* Student monitoring is poor.

## Requirements specification

This section includes the features and functionalities of the software as per the client’s requirement. This section is basically a document prepared keeping client’s requirement in mind, so that they can review the features of the software.

## Problem domain description

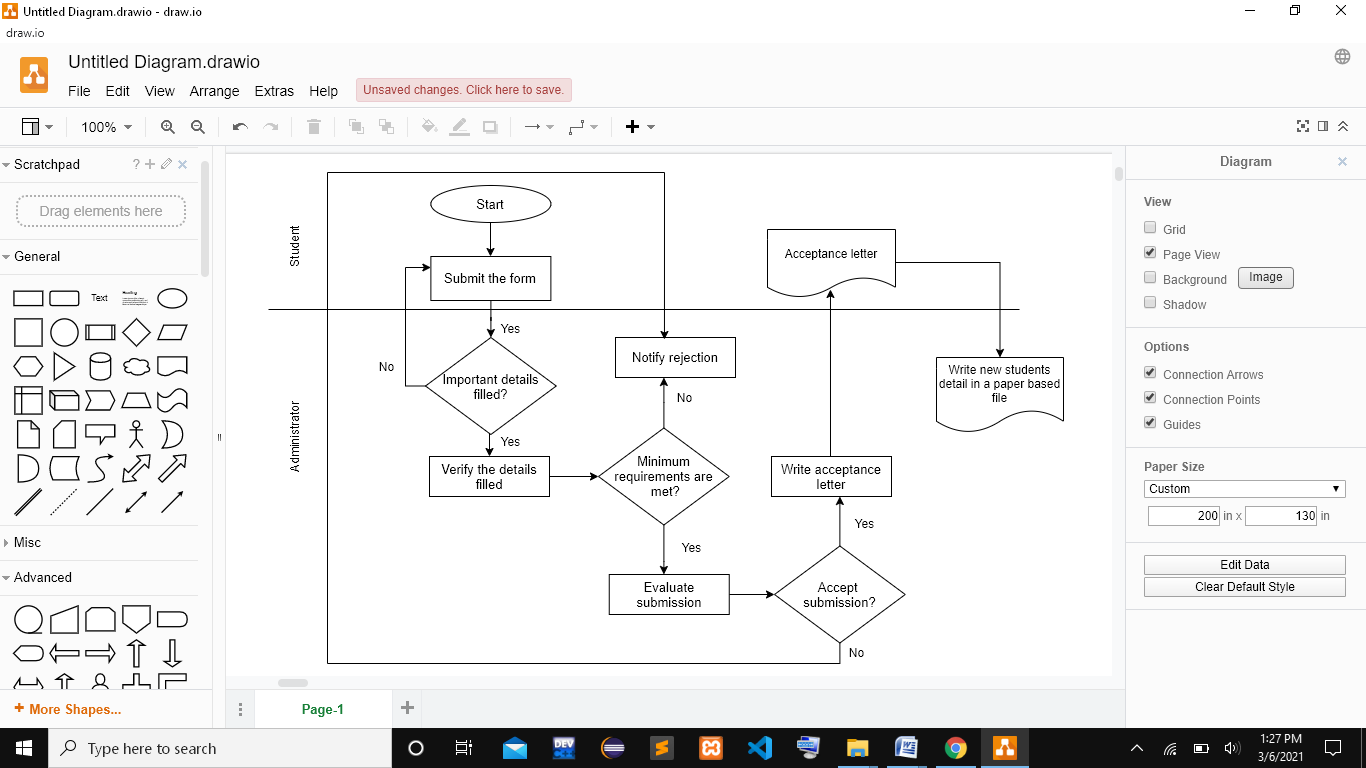
Problem domain is the field of interest that should be inspected to solve a problem. It is basically focusing at what are the things that we need to solve the problems and ignoring irrelevant things. In order to identify the solution domain, the limitations in the current business operations must be known so that we can solve them in our software.

## Existing business operations

The university uses paper based system to manage its operations like enrolling a student, taking attendances, taking examinations, scheduling PAT day etc. We have tried to illustrate these operations with the help of flow charts.

## Enrolling a student to the university

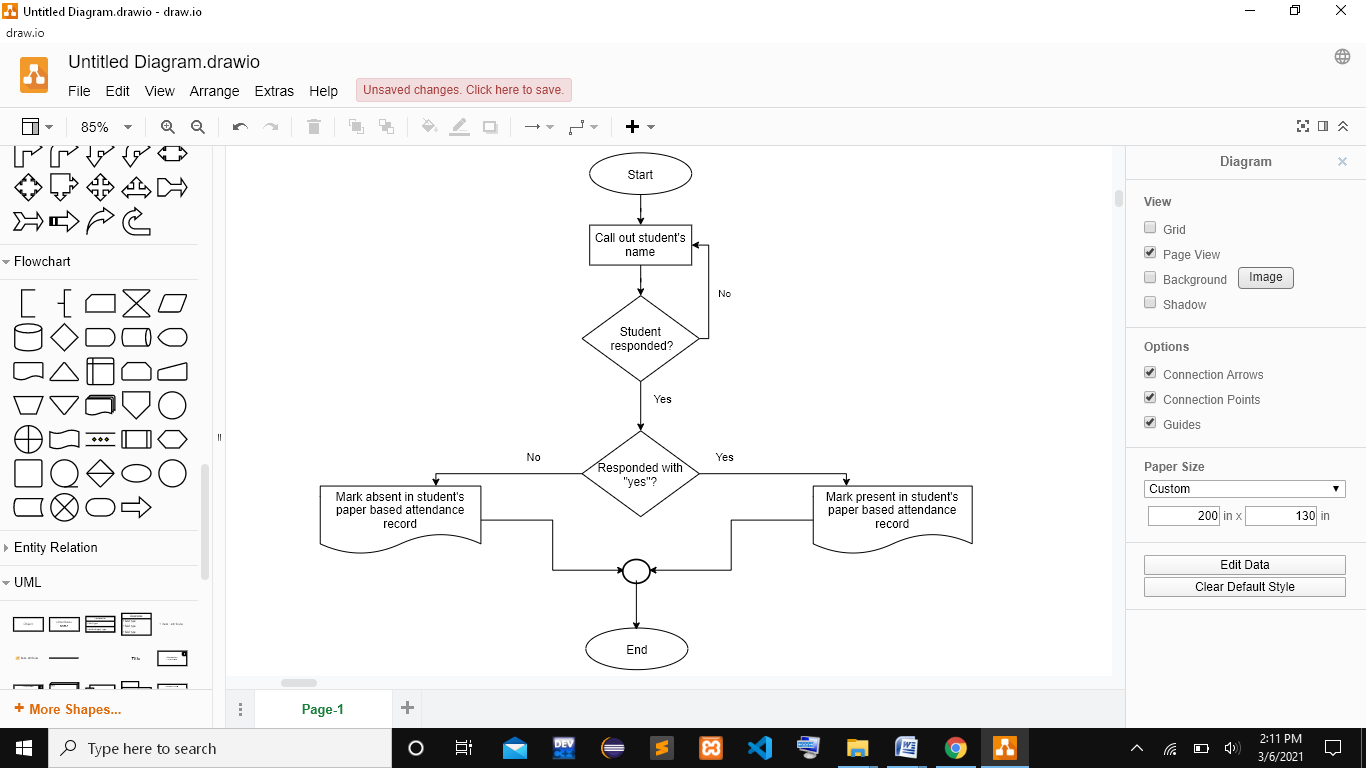
The following flowchart shows how students are enrolled to the university via paper-based system.

****

***Fig 1.4 Flowchart of enrolling a student to the university***

## Taking attendances

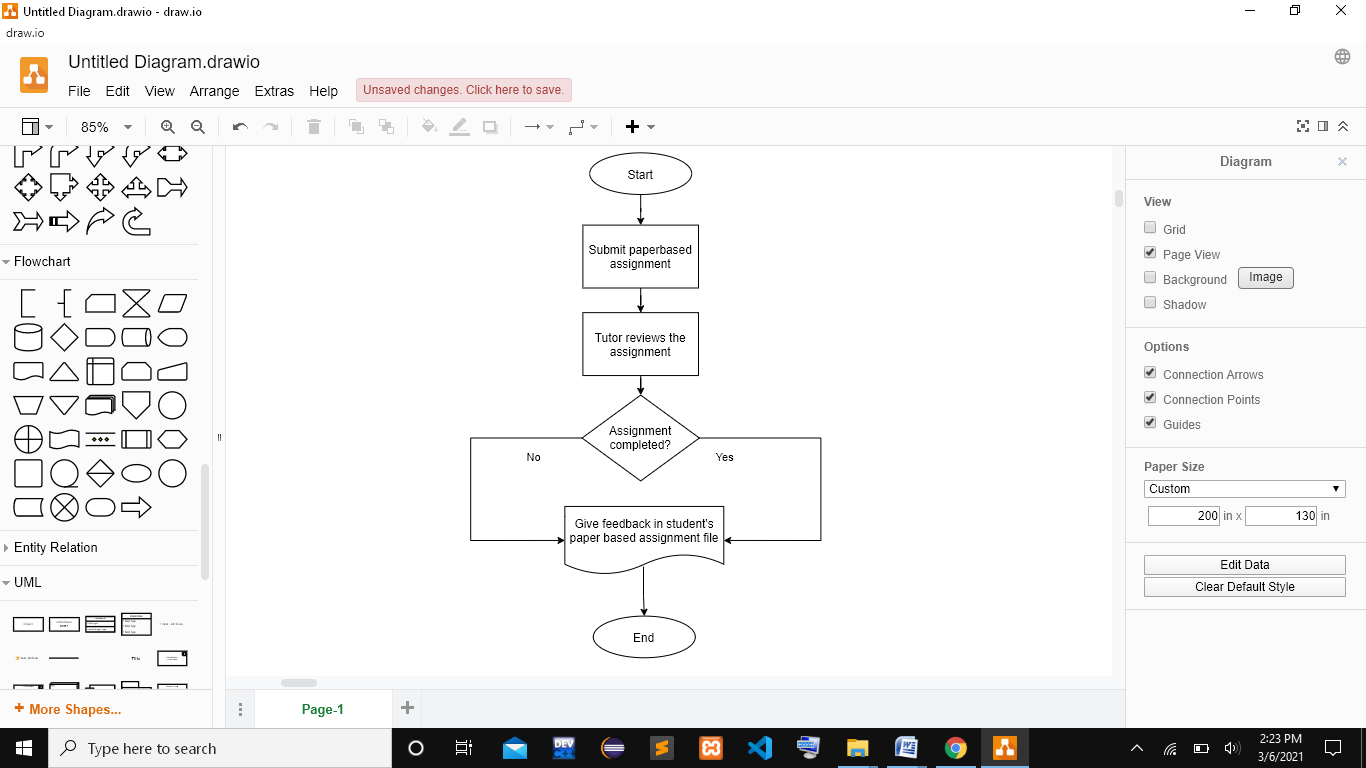
The following flowchart shows how attendance of students is taken in university via existing paper based system.

****

***Fig 1.5 Flowchart of taking attendance***

## Submitting assignments and giving feedbacks

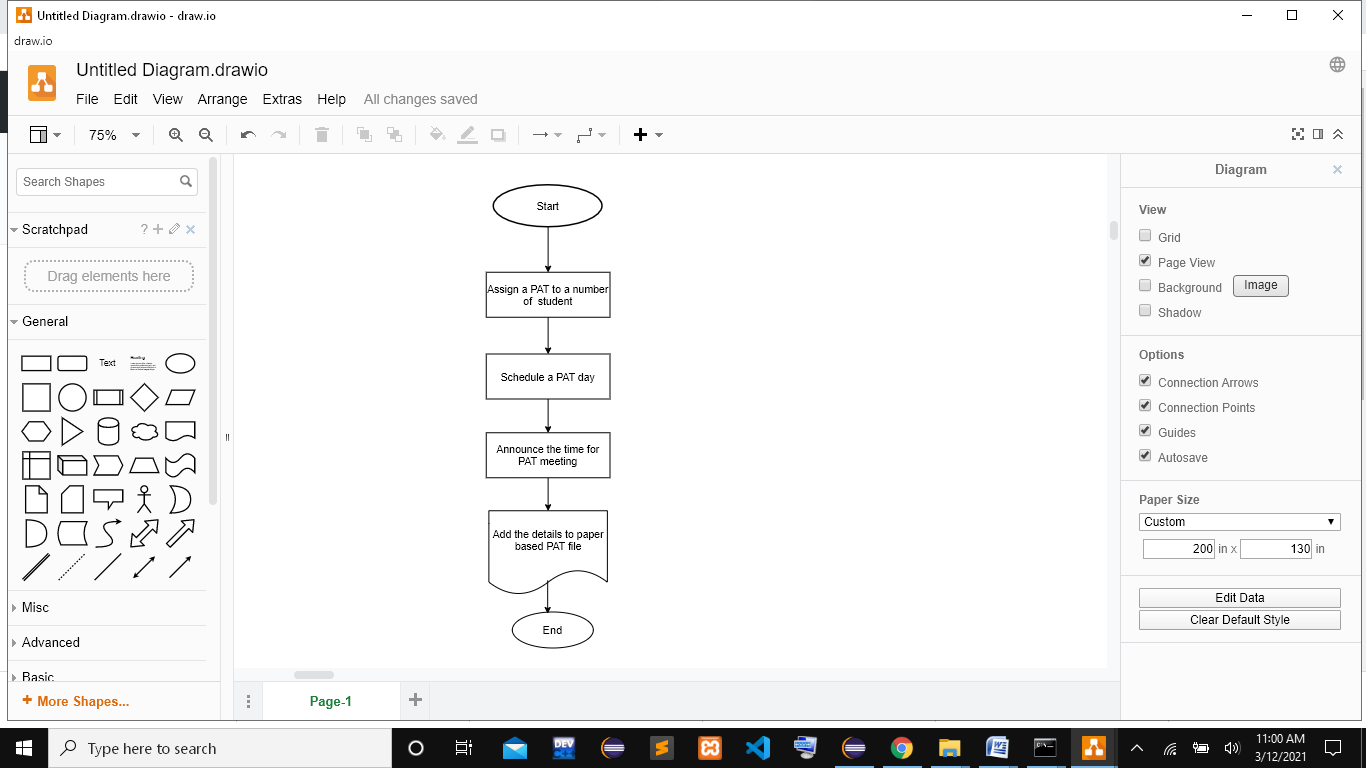
The following flowchart shows how assignments are submitted and feedbacks are given to the students in the university via existing paper based system.

****

***Fig 1.6 Flowchart of submitting and reviewing assignment***

## Scheduling PAT day

The following flowchart shows how PAT day is scheduled in the university via existing paper based system.

****

***Fig 1.7 Flowchart for scheduling PAT day***

## Summary of existing business operations

There are few limitations in the existing business operations of the university. The operations are expensive and time consuming. Some of the limitations are listed below.

* The student enrollment process is paper based so it is time consuming.
* Attendance records are added in paper based document so there is a risk of data being lost.
* Assignments checking and giving feedbacks is quite hectic and slow process.
* PAT assignment and scheduling is not done in systematic way.
* Data can be lost and damaged.
* Storage is a problem in paper based system.

Functional requirements

Here we will be discussing about the different functionalities which are in our system. All the functionalities are elaborated in this section. The functions are kept according to the need of the owner. All the basic functions are required to run a school is kept in our system. Only difference here is the it is in a digital form. This part also consists of all the access levels of admin, module leader, students.

## Required features of the system

First, we identified the entire existing feature in the college and all the problems existed and then we started to solve it in a digital form, and they are our features of our system.

## Database and backup

As we are moving towards the digital way from a clerical based system, all the data are stored in the database. Paper-based management system was here but in our digital system’s all data are safely and soundly kept at the database. It is better than the paper-based management system. We can also fetch all those data within a minute. There is no risk of loss, extra space, fire. Only authorized person will be able to read and fetch the data, so it is more secured.

## Students records

It is very important to any college management system to save the data of the students. Our system also does the same. We save data of three types of the students and handle accordingly. One is those who are passed out and one is who are not and last one is the student who just got engaged in the college for the specific period of time.  Students can get all their information whenever required. All the functionalities of the system will only be used by the students who are still reading here.

## Staff records

Our system will also store the data of the different level of staffs which are ever being associated to our school. There are 3 different types of staffs that we record. One will be the teaching-staff, one will be the non-teaching staffs and last will be the one who can be both but are currently not charged in college.

## Courses and modules records

There is also the data stored related to courses. These records are handled by the authorized person. All the student must be enrolled in their respective course. And inside course there will be many modules.

## Management of module

* The admin is only responsible for the adding, editing and deleting operation of the modules.
* All the students are assigned in course consisting different modules and all those modules are taught by the teachers.
* Student can download all the files required to study of that particular module.

## Management of assignments

* The module teachers will be responsible for the adding, editing, deleting the assignments.
* They will also provide the submission link and the date for the submission.
* Student can submit and view their assignments directly from the system.

## Management of attendance

* The module teacher will be again responsible for the attendance of the students. They can mark present or mark absent.
* Module teacher can see the attendance of all the students.

## Management of personal tutor

* All the students will have their respective personal tutor (PAT).
* PAT can directly talk with the students on the topic other than of study.

## Management of the grades

* The module teacher will give assignments and then will give respective grade to the students.
* The module teacher can also add the comments or the remarks to the students. This will help the students to know their mistake and place for the improvements.
* The students will be able to see the grades and the comments and remarks from the module teacher.

## Access rights/Level of access

The table below consists of the access rights that a particular user in the system may have. It is important to identify which operations should be accessed by which user group.

## Administrator, staff & student management

1. **Administrator records management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add Administrator | Can add an administrator | Can’t add an administrator | Can’t add an administrator |
| To edit administrator | Can change the information of administrator | Can’t change the information of administrator | Can’t change the information of administrator |
| To delete administrator | Can delete the information of administrator | Can’t delete the information of administrator | Can’t delete the information of administrator |
| To view administrator | Can view the information of administrator | Can’t view the information of administrator | Can’t view the information of administrator |

**Table Name: Admins**

**Description: It is the table that consists the different fields related to the admin. It has all the necessary and role factors related to the admin and they are needed in the system.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraints** | **Remarks** |
| Assignment\_id | Int | NOTNULL | - |
| Module\_id | Int | Fk | - |
| Deadline | Datetime | NOTNULL | - |
| Content | Longtext | NOTNULL | - |
| Semester | Int | NOTNULL | - |

1. **Module Leader records management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add module teacher | Can add a new module teacher | Can’t add the module teacher | Can’t add the module teacher |
| To edit module teacher | Can change the information of the module teacher | Can’t edit the information of the module teacher | Can’t edit the information of the module teacher |
| To delete module teacher | Can delete the information of module teacher | Can’t delete the information of the module teacher | Can’t change the information of the module teacher |
| To view Module leader | Can view all the information of the module teacher | Can view their own information. | Can’t view the information of the module teacher. |

**Table Name: Staff**

**Description: This is the table which has attributes related to the staff. It has all the necessary and role attributes which are related to the staff and they are needed in the system. It consists of the type and the constraint of the attributes.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraints** | **Remarks** |
| staff\_id | Int | NOTNULL | - |
| Course\_id | Int | Fk | - |
| name | varchar | NOTNULL | - |
| email | varchar | NOTNULL | - |
| password | varchar | NOTNULL | - |
| surname | varchar | NOTNULL | - |
| address | Varchar | NOTNULL | - |
| Date\_of\_join | Date | NOTNULL | - |
| salary | varchar | NOTNULL | - |
| role | Varchar | NOTNULL | - |
| Module\_id | Int | NOTNULL | - |

1. **Module Record Management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add module | Can add a new module | Can’t add the module | Can’t add the module |
| To edit module | Can change the information of the module | Can’t edit the information of the module | Can’t edit the information of the module |
| To delete module | Can delete the information of module | Can’t delete the information of the module | Can’t change the information of the module |
| To view model | Can view all the information of the module | Can’t view the other modules except theirs. | Can’t view the information of the module |

**Table Name: Modules**

**Description: This is the table which has attributes related to the module. It has all the necessary and role attributes which are related to the module and they are needed in the system.It consists of the type and the constraint of the attributes and remarks if necessary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraint** | **Remarks** |
| Module\_id | Int | NOTNULL | - |
| Course\_id | Int | NOTNULL | - |
| Module\_name | Varchar | NOTNULL | - |
| Module\_credit | Int | NOTNULL | - |
| Module\_level | Int | NOTNULL | - |
| Ass\_1 | Varchar | NOTNULL | - |
| Ass\_2 | Varchar | NOTNULL | - |
| Exam | Varchar | NOTNULL | - |

1. **Course Record Management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add course | Can add a new course | Can’t add the course | Can’t add the course |
| To edit course | Can change the information of the course | Can’t edit the information of the course | Can’t edit the information of the course |
| To delete course | Can delete the information of course | Can’t delete the information of the course | Can’t change the information of the course |
| To view course | Can view all the information of the course | Can’t view the information of the courses | Can’t view the information of the course in which they are being enrolled. |

**Table Name: Courses**

**Description: This is the table which has attributes related to the course. It has all the necessary and role attributes which are related to the course and they are needed in the system. It consists of the type and the constraint of the attributes and remarks if necessary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraint** | **Remarks** |
| Course\_id | Int | NOTNULL | - |
| Course\_head\_staff\_id | Int | NOTNULL | - |
| Course\_name | Varchar | NOTNULL | - |
| Start\_date | Datetime | NOTNULL | - |

1. **Content Record Management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add content | Can’t add a new content | Can add the content | Can’t add the module |
| To edit content | Can’t change the information of the content | Can edit the information of the content | Can’t edit the information of the module |
| To delete content | Can’t delete the information of content | Can delete the information of the content | Can’t change the information of the module |
| To view content | Can’t view all the information of the content | Can view all the thing of the content | Can see all the content of their respective modules. |

**Table Name: Contents**

**Description: This is the table which has attributes related to the content. It has all the necessary and role attributes which are related to the content and they are needed in the system. It consists of the type and the constraint of the attributes and remarks if necessary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraint** | **Remarks** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. **Assignment Record Management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add assignment | Can’t add a new assignment | Can add the assignment | Can’t add the module |
| To edit assignment | Can’t change the information of the assignment | Can edit the information of the assignment | Can’t edit the information of the module |
| To delete assignment | Can’t delete the information of assignment | Can delete the information of the assignment | Can’t change the information of the module |
| To view assignment | Can’t view all the information of the assignment | Can view all the thing of the assignment | Can see all the content of their respective modules. |
| To Submit the assignment | Can’t submit the assignment | Can’t submit the assignment | Can submit the assignment of the respective module |

**Table Name: Assignments**

**Description: This is the table which has attributes related to the assignment. It has all the necessary and role attributes which are related to the assignment and they are needed in the system. It consists of the type and the constraint of the attributes and remarks if necessary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraints** | **Remarks** |
| Assignment\_id | Int | NOTNULL | - |
| Module\_id | Int | Fk | - |
| Deadline | Datetime | NOTNULL | - |
| Content | Longtext | NOTNULL | - |
| Semester | Int | NOTNULL | - |

1. **Grade Record Management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add grade | Can’t add a new grade | Can add the grade | Can’t add the grade |
| To edit grade | Can’t change the information of the grade | Can edit the information of the grade | Can’t edit the information of the grade |
| To delete grade | Can’t delete the information of grade | Can delete the information of the grade | Can’t change the information of the grade |
| To view grade | Can’t view all the information of the grade | Can view the grade of all the students | Can see all the grades of their respective courses. |

**Table Name: Grades**

**Description: This is the table which has attributes related to the grade. It has all the necessary and role attributes which are related to the grade and they are needed in the system. It consists of the type and the constraint of the attributes and remarks if necessary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraints** | **Remarks** |
| Assignment\_id | Int | NOTNULL | - |
| Module\_id | Int | Fk | - |
| Deadline | Datetime | NOTNULL | - |
| Content | Longtext | NOTNULL | - |
| Semester | Int | NOTNULL | - |

1. **Attendance Record Management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add attendance | Can’t add a new attendance | Can add the attendance | Can’t add the grade |
| To edit attendance | Can’t change the information of the attendance | Can edit the information of the attendance | Can’t edit the information of the grade |
| To delete attendance | Can’t delete the information of attendance | Can delete the information of the attendance | Can’t change the information of the grade |
| To view attendance | Can’t view all the information of the attendance | Can view the attendance of all the students | Can see their attendance ratio |

**Table Name: Attendances**

**Description: This is the table which has attributes related to the attendance. It has all the necessary and role attributes which are related to the attendance and they are needed in the system. It consists of the type and the constraint of the attributes and remarks if necessary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraint** | **Remarks** |
| Attendance\_id | Int | NOTNULL | - |
| Attendance\_module\_id | Int | NOTNULL | - |
| Student\_id | Int | NOTNULL | - |
| Attendance\_time | date | NOTNULL | - |

1. **Submission Record management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Submission | Module Teacher | Student |
| To add submission | Can’t add the Submission | Can’t add theSubmission | Can add the Submission |
| To edit submission | Can’tchangethe information of Submission | Can’t change the information of Submission | Can change the information of Submission |
| To delete submission | Can’t delete the information of Submission | Can’t delete the information of Submission | Can delete the information of Submission |
| To view submission | Can view the information of Submission | Can view the information of Submission | Can view the information of Submission |

**Table Name: Submissions**

**Description: This is the table which has attributes related to the submission. It has all the necessary and role attributes which are related to the submission and they are needed in the system. It consists of the type and the constraint of the attributes and remarks if necessary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraints** | **Remarks** |
| Submission\_id | Int | NOTNULL | - |
| Student\_id | Int | NOTNULL | - |
| Assignment\_id | int | NOTNULL | - |
| Submission\_date | datetime | NOTNULL | - |
| content | longtext | NOTNULL | - |

1. **Students record management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add students | Can add a student | Can’t add a student | Can’t add a student |
| To edit students | Can change the information of students | Can’t change the information of students | Can’t change the information of students |
| To delete students | Can delete the information of students | Can’t delete the information of students | Can’t delete the information of students |
| To view students | Can view the information of students | Can’t view the information of students | Can view the information of students |

**Table Name: Students**

**Description: This is the table which has attributes related to the content. It has all the necessary and role attributes which are related to the content and they are needed in the system. It consists of the type and the constraint of the attributes and remarks if necessary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraints** | **Remarks** |
| Student\_id | Int | NOTNULL | - |
| Course\_id | Int | Fk | - |
| name | varchar | NOTNULL | - |
| email | varchar | NOTNULL | - |
| password | varchar | NOTNULL | - |
| surname | varchar | NOTNULL | - |
| address | Varchar | NOTNULL | - |
| phone | Varchar | NOTNULL | - |
| gender | Char | NOTNULL | - |
| Date\_of\_birth | Date | NOTNULL | - |
| Registration\_year | date | NOTNULL | - |
| Student\_status | Varchar | NOTNULL | - |

1. **RoutinesRecord Management**

|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Administrator | Module Teacher | Student |
| To add routines | Can add a new routine | Can add the routines | Can’t add the routines |
| To edit routines | Can change the information of the routines | Can edit the information of the routines | Can’t edit the information of the routines |
| To delete routines | Can delete the information of routines | Can delete the information of the routines | Can’t change the information of the routines |
| To view routines | Can view all the information of the routines | Can view the routines. | Can see the information of the routines |

**Table Name: Routines**

**Description: This is the table which has attributes related to the routine. It has all the necessary and role attributes which are related to the routine and they are needed in the system. It consists of the type and the constraint of the attributes and remarks if necessary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Type** | **Constraints** | **Remarks** |
| Routine\_id | Int | NOTNULL | - |
| Course\_id | Int | Fk | - |
| Day\_number | Datetime | NOTNULL | - |
| Day | Longtext | NOTNULL | - |
| Module\_id | Int | NOTNULL | - |
| Start\_time | Varchar | NOTNULL | - |
| End\_time | varchar | NOTNULL | - |
| Second\_module\_name | varchar | NOTNULL | - |
| Second\_start\_time | varchar | NOTNULL | - |
| Second\_end\_time | varchar | NOTNULL | - |
| Semester | int | NOTNULL | - |
| Class\_type\_1 | varchar | NOTNULL | - |
| Class\_type\_2 | varchar | Default NULL | - |

## Output formats for onscreen displays

In order to make the course management system more user friendly, different kinds of emails are sent to particular users on different occasions. These emails are triggered to notify the users of certain events in the system. The identical formats of the email that are sent to the users on certain events are illustrated below:

## Email sent after student registration

**Hi<student’s first name>,**

**Congratulations! You have successfully been registered as a student to the Woodlands University. You will need to set up an account to have access to the university’s course management system .Your username is your university id i.e. 3098776 .You have been given a temporary password by the university. The password consists of:**

* **The % symbol.**
* **The last two letters of your family name, both letters in capital letter.**
* **The first four digits of your phone number 10 digit phone number which excludes the country code.**

**If your family name is ‘Jackson’ and your phone number is 1234567890 then your password will be %JA1234.To change your password click on this<LINK>.**

**If you have any issues on accessing the university’s virtual learning environment you can mail us at** [**wlds@woodlands.edu.uk**](mailto:wlds@woodlands.edu.uk)**.**

**Best Regards,**

**Woodlands University**

**Note: *This message was automatically generated. Please do not respond to this email.***

## Email sent after publishing course content

**Hi<student’s first name>,**

**Your course content for <module name> has been posted. Please click on this <LINK>to view the course content.**

**If the link is not working you can email me at <email address>.**

**Best Regards,**

**<Module leader’s name>**

**Note: *This message automatically generated. Please do not respond to this email.***

## Email sent after publishing grades

**Hi<student’s first name>,**

**The grades for <module name>of <term level> have been posted. Please click on this <LINK>to view your grades.**

**If the link is not working you can email me at <email address>.**

**Best Regards,**

**<Module leader’s name>**

**Note: *This message automatically generated. Please do not respond to this email.***

## Email sent after publishing assignment

**Hi<student’s first name>,**

**Your assignment for <module name> of <term level> for <year> has been added.Please click on this <LINK>to view your assignment.**

**If the link is not working you can email me at <email address>.**

**Best Regards,**

**Woodlands University**

**Note: *This message automatically generated. Please do not respond to this email.***

## Email sent after submitting assignment

**Hi<student’s first name>,**

**Submission id: <submission id number>**

**You have successfully submitted <submission title> to the assignment <assignment name> for <module name> on <submission time>.**

**Best Regards,**

**Woodlands University**

**Note: *This message automatically generated. Please do not respond to this email.***

## Email sent after PAT allocation

**Dear students,**

**I have been assigned as your PAT. We have a PAT day on <date>.I would like you all to gather at <place> from <start time> to <end time> sharp. Hope to see you all with your queries.**

**Best Regards,**

**<Personal Assistant Tutor’s (PAT) Name>**

**Note: *This message automatically generated. Please do not respond to this email.***

## Performance requirements

This section consists of the overall speed, capacity, reliability, usability, of our system.

It helps us to determine how well the system is.

## Speed

Throughput: Since there are many data, throughput will get high but the data will be a smoothprocess.

Response Time: The total estimated response time is 3 to 5 sec. The worstscenario will

be of around 10 sec.

## Capacity

The total capacity of our system has been thought initially to be around 2000. This is the number up-to which we will be accessing the system without any lag. We have a large storage space and the storage can be increased as per our need.

## Reliability

The system will work continuously and only stops at the time of server maintenance. We also have multiple servers to serve a good and fluent experience to the user. If, unfortunately one stops working, another will then act as a secondary server and the user would have no problem on accessing the page. The system will be available 24 hours.

## Usability

Our system is very user friendly and easy to user. It is very interactive system. Anyone can learn to use it in no time. To operate it we don’t even need a super talented guy. Any person who is good in English can handle it very well. Also, it will receive many updates in the near future which will again increase the efficiency of our system as well to user experience.

## Security

Our system is much secured too. Only the authorized person with the respective email passwords can only enter to the site and perform various activity. And all the passwords of all the user are safely encrypted in order to reduce the risk of password leak. It is also safe from external attacks.

## Design constraints

This part of report elaborates the non-functional requirements which could ever affect our system. All the necessary constraint is explained here. The constraints are being taken by the client and using those constraints whole system has been made. Following are the constraints:

Our system will be available for Windows, Linux, MacOS, Android, IOSetc.Thiswill open in almost all the devices like pc or mobile phones. Our main theme color is yellow and other will be white.

Since, there is no any specific programming language told, we used JavaScript to complete it.

We have used functional approach. MySQL is used for storing data.

The system has been built within the specified time and budget.

## Commercial constraints

The cost for the project development is illustrated below:

|  |  |  |  |
| --- | --- | --- | --- |
| Project information |  |  |  |
| Total length of project | 22 weeks |  |  |
| Hourly wage | £25 |  |  |
| Number of workers | 4 |  |  |
|  |  |  |  |
| Project breakdown |  |  |  |
| Sections | Weeks | Hour per week per person | Total wage |
| Requirement specification | 5 | 5 | £2500 |
| Design and analysis | 5 | 10 | £5000 |
|  |  |  |  |
| Project built breakdown | 10 |  |  |
| Console | 30% | 10 | £6000 |
| Website | 25% | 10 | £5500 |
| Testing | 30% | 10 | £6000 |
| Evaluation | 5% | 10 | £1000 |
|  |  |  |  |
| Additional cost |  |  |  |
| Reason | Cost |  |  |
| Software/hardware | £1000 |  |  |
| Office cost | £1500 |  |  |
|  |  |  |  |
| Total built cost | £2500 |  |  |
|  |  |  |  |
| Other expenses (10%)  Profit margin (30%)  Profit rights | £2000  £6000  Negotiable |  |  |
| Total project cost | £8000 |  |  |

# System analysis and design

System analysis and design is a way to analyze an existing system or create a new one in order to update or upgrade the system according to its requirements. According to the description of the client, we need to design and develop an electronic course management system which will replace the existing clerically based course management system.

In this section, we have discussed what the system should do and how to accomplish the required features of the system.

## Preliminary design stages

In contrast to the analysis phase, the design phase determines how the system will accomplish the objectives. During the design phase each portion of the new system is analyzed and designed in detail.

The design phase often ends with a prototype, which is a working model of the system. The design phases consists of different stages, hence it is one of the most crucial phases of software development which will affect the final product.

## Textual Analysis

In textual analysis in design phase we have extracted the useful information from the textual brief of the requirements of the software we are going to develop. After analyzing the problem domain, candidateclasses (entities) and responsibilities (roles) of the candidate classes are identified.

The nouns are identified as the class candidates and the verbs are identified as responsibility of candidate classes. The following illustrates the candidate classes along with their responsibilities.

|  |  |
| --- | --- |
| **Candidate Classes** | **Responsibilities of Candidate Classes** |
| Students | view grades,viewassignment,viewattendance,viewcourses,uploadassignment,viewannouncements,view content |
| Module contents | add content,deletecontent,editcontent,viewcontent,download content |
| Courses | add course,editcourse,deletecourse,view course |
| Modules | add module, delete module, edit module, view module |
| Module leaders | Add module content, edit module content, delete module content, add assignment content, edit assignment content, delete assignment content, add announcement, grade assignment |
| Assignments | add assignment, edit assignment, delete assignment, view assignment, submit assignment |
| Announcements | add announcements, edit announcements, delete announcements, view announcements, submit assignments |
| Admin | add admin, edit admin, delete admin, add module leader, edit module leader, delete module leader, delete module leader, make student live, make student dormant, add announcements |
| Grades | add grade, edit grade, delete grade, view grade |
| Timetables | add timetable, edit timetable, delete timetable, view timetable |

## Significant event analysis

Significant event analysis is the process of analyzing various events or functionalities that the system includes.

Performers are those entities who have access to that event and candidate attributes are those attributes that are required to perform the events. The table below illustrates different events in the system along with its performers and candidate attributes.

|  |  |  |
| --- | --- | --- |
| **Event** | **Performers** | **Candidate Attributes** |
| **Login** | * Admin * Module Leader * Student | * Staff\_id * Name * Surname * Email * Address * Date\_of\_join * Course\_id * Module\_id * Salary * Role * Password |
| **Logout** | * Admin * ModuleLeader * Student | * Admin\_id * Name * Surname * Email * Address * Date\_of\_join * Salary * Password * Staff\_id * Name * Surname * Email * Address * Date\_of\_join * Course\_id * Module\_id * Salary * Role * Password * student\_id * course\_id * name,emai * password * surname * address * phone * gender * date\_of\_birth * registration\_year * student\_status |
| **Add/edit/delete Admin** | * Admin | * Admin\_id * Name * Surname * Email * Address * Date\_of\_join * Salary * Password |
| **Add/edit/delete Module**  **Leader** | * Admin | * Admin\_id * Name * Surname * Email * Address * Date\_of\_join * Salary * Password * Staff\_id * role |
| **Add/edit/delete assignments** | * Module Leader | * Staff\_id * Name * Surname * Email * Address * Date\_of\_join * Course\_id * Module\_id * Salary * Role * Password * Assignment\_id |
| **Add/edit/delete grades** | * Module Leader | * Staff\_id * Name * Surname * Email * Address * Date\_of\_join * Course\_id * Module\_id * Salary * Role * Password |
| **Add Course** | * Admin | * Admin\_id * Name * Surname * Email * Address * Date\_of\_join * Salary * Password * Staff\_id * role * course\_id |
| **Add Module** | * Admin | * Admin\_id * Name * Surname * Email * Address * Date\_of\_join * Salary * Password * Staff\_id * role * module\_id |
| **Add/edit/delete attendance** | * Module Leader | * Staff\_id * Name * Surname * Email * Address * Date\_of\_join * Course\_id * Module\_id * Salary * Role * Password * Student\_id |
| **Add/edit/delete routines** | * Module Leader | * Staff\_id * Name * Surname * Email * Address * Date\_of\_join * Course\_id * Module\_id * Salary * Role * Password * Course\_id * Module\_id |
| **Add submissions** | * Module Leader | * Staff\_id * Name * Surname * Email * Address * Date\_of\_join * Course\_id * Module\_id * Salary * Role * Password * Module\_id * Assignment\_id |
| **Add/view/edit diaries** | * Student | * student\_id * course\_id * name,emai * password * surname * address * phone * gender * date\_of\_birth * registration\_year * student\_status |
| **View Assignments** | * Student | * student\_id * course\_id * name,emai * password * surname * address * phone * gender * date\_of\_birth * registration\_year * student\_status |
| **View Routines** | * Student | * student\_id * course\_id * name,emai * password * surname * address * phone * gender * date\_of\_birth * registration\_year * student\_status |
| **View Grades** | * Student | * student\_id * course\_id * name,emai * password * surname * address * phone * gender * date\_of\_birth * registration\_year * student\_status |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CLASS** | | **Module** | | **PART: 1/1** |
| TYPE OF OBJECT  Add,delete,edit and view module | | |  | |
| Queries | staff\_id,name,surname, email, address,date\_of\_join,course\_id,module\_id,salary,role,password | | | |
| Commands | Add, Edit, Delete Module Leader | | | |
| Constraints | Module Leader must be teaching respective modules. | | | |

## Command queries and constraints

Command Queries and Constraints is a method of evaluating the system in a more refined way. It consists of Queries which consists of the different attributes and of that function. Commands are the action which is to be performed and the constraints are the necessary criteria that should be adopted to perform the task under the given module.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CLASS | | Module | | PART: 1/1 |
| TYPE OF OBJECT  Add, delete, edit and view assignments | | |  | |
| Queries | assignment\_id, module\_id, deadline, content, semester, staff\_id,name,surname, email, address,date\_of\_join,course\_id,module\_id,salary,role,password | | | |
| Commands | Add, Edit, Delete assignments | | | |
| Constraints | * There must be a brief file to every assignment. * Assignment must have a submission date and deadline. * It must be changed as per the nature. | | | |

## Detailed Static / Dynamic System Designs

**Use case diagram**

Use case diagram holds a dynamic nature of a system. It consists of an actor and use cases where use case means the particular functionality of a system.

**UCD of Student:**

Below is the use case diagram of Student. It represents the different functionalities of the Students.

**Diagram

Description automatically generated**

**UCD OF Module Leader:**

Below is the use case diagram of Module Leader. It represents the different functionalities of the Module Leaders.

**Diagram

Description automatically generated**

**UCD of Admin:**

Below is the use case diagram of admin. It represents the different functionalities of the Admin.

**Diagram

Description automatically generated**

## First Draft BON System Architecture Diagram

Database/System

Website

University

User

Web App

System/Database

Administrator

Student

Module Leader

## BON System Chart

|  |  |  |  |
| --- | --- | --- | --- |
| **SYSTEM**  SCHOOL MANAGEMENT SYSTEM | | | **PART:1/1** |
| **PURPOSE:** SYSTEM KEEPING DETAIL OF UNIVERSITY DIGITALLY | | **INDEXING:** author: Sujal Gautam  Keywords**:** admin, module leader, student | |
| **Cluster** | **Description** | | |
| LOGIN\_ELEMENTS Element that are needed to login such as username   password.  ADMIN\_PROPERTIES Property of admin such as their name, address email  So on.  MODULE\_LEADER\_PROPERTIES Properties of module leader such as their name, address,  Email, password and so on.  STUDENT\_PROPERTIES Properties of student such as name, address, email, password  So on.  OPTIONS Option to perform add, delete, view, edit tasks. | | | |

## BON Cluster Charts

|  |  |  |  |
| --- | --- | --- | --- |
| **CLUSTER**ADMIN PROPERTIES | | | **PART:1/1** |
| **PURPOSE:** SYSTEM KEEPING DETAIL OF UNIVERSITY DIGITALLY | | **INDEXING:** author: Sujal Gautam  Keywords**:** admin, module leader, student | |
| **Cluster** | **Description** | | |
| ADD Adds the required selected criteria.  VIEW Adds the required selected criteria.  EDIT Edits the required selected criteria.  DELETE Delete the required selected criteria. | | | |

|  |  |  |
| --- | --- | --- |
| CLASS MODULE | | PART: 1/1 |
| TYPE OF OBJECT  Add,delete,edit and view module | | INDEXING:  Cluster: MODULE\_LEADER\_PROPERTIES |
| Queries | staff\_id,name,surname, email, address,date\_of\_join,course\_id,module\_id,salary,role,password | |
| Commands | Add, Edit, Delete Module Leader | |
| Constraints | Module Leader must be teaching respective modules. | |

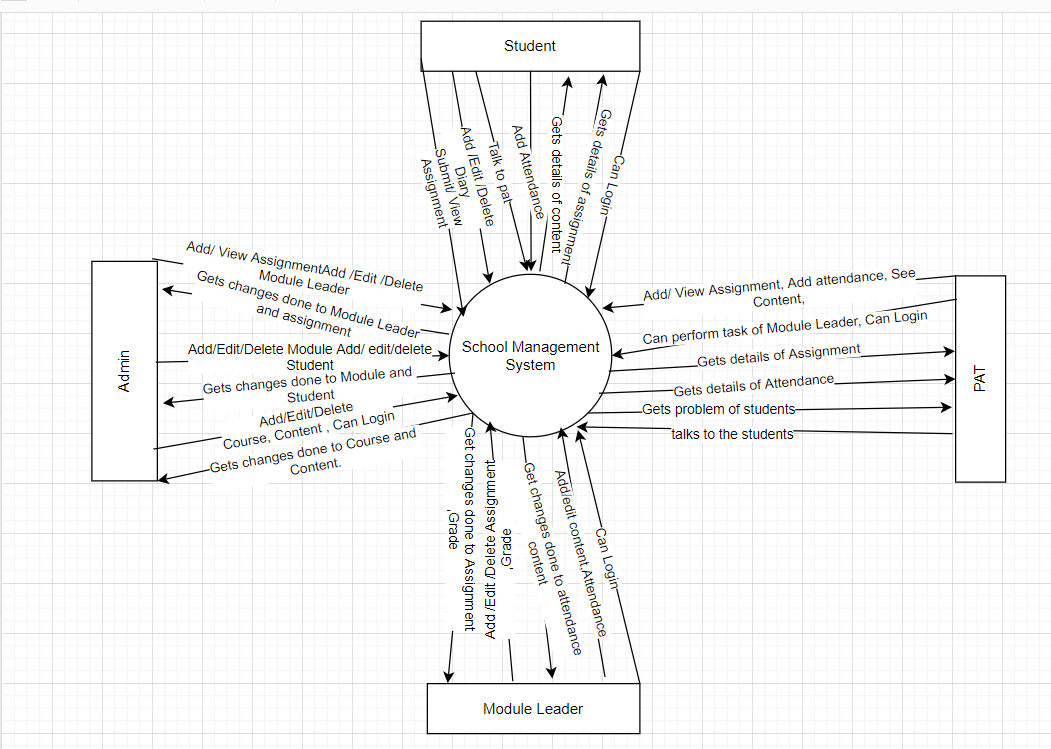
## BON Class Charts

## Data flow Diagram

A data flow diagram is a flow of system with respect to inputs and outputs.

## Context level

Below is the Zero Level / Context Level Data Flow Diagram of our system. In this diagram. It describes the high-level process of our system. Only few had been described here. It is the fundamental overview of our system. It gives a quick glance about some of the functionalities superficially. Rounded Rectangle denotes entities, arrow determined the flow, circle denotes process.

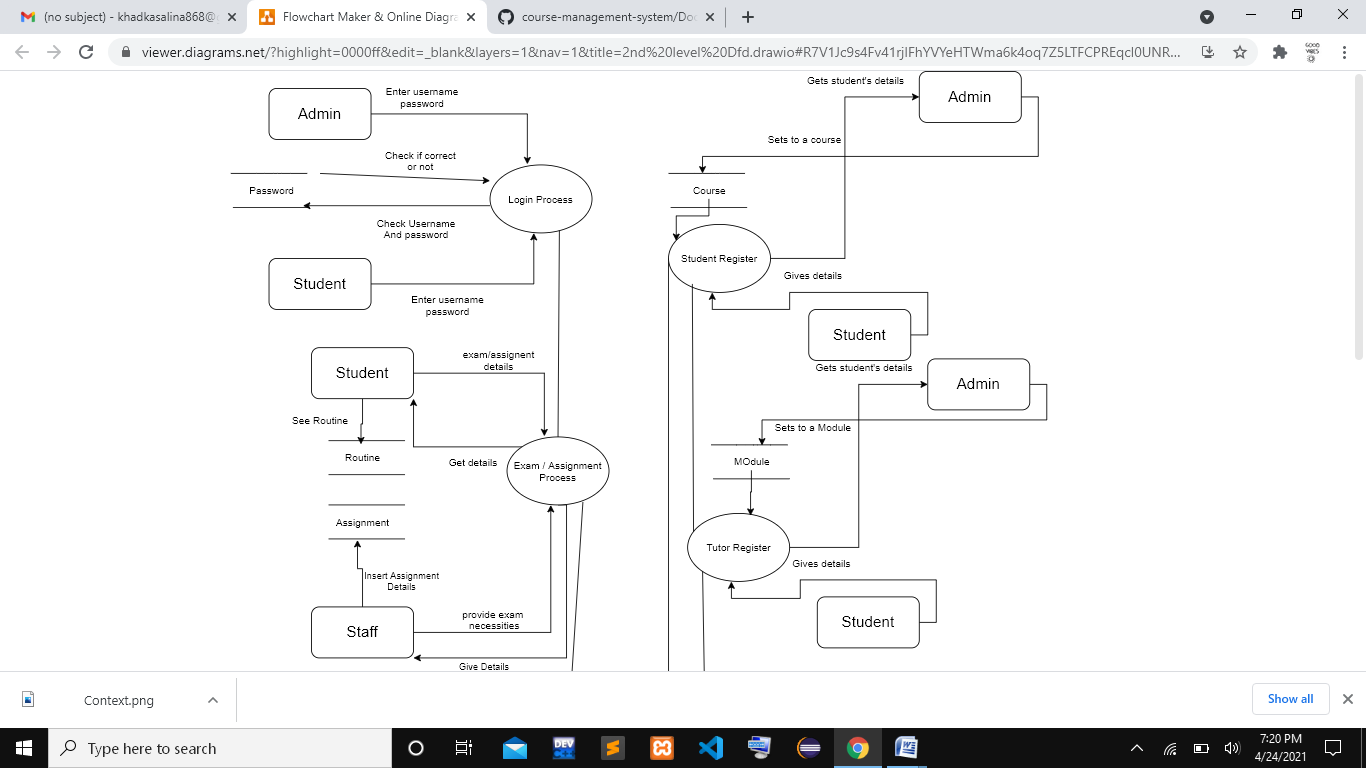
****

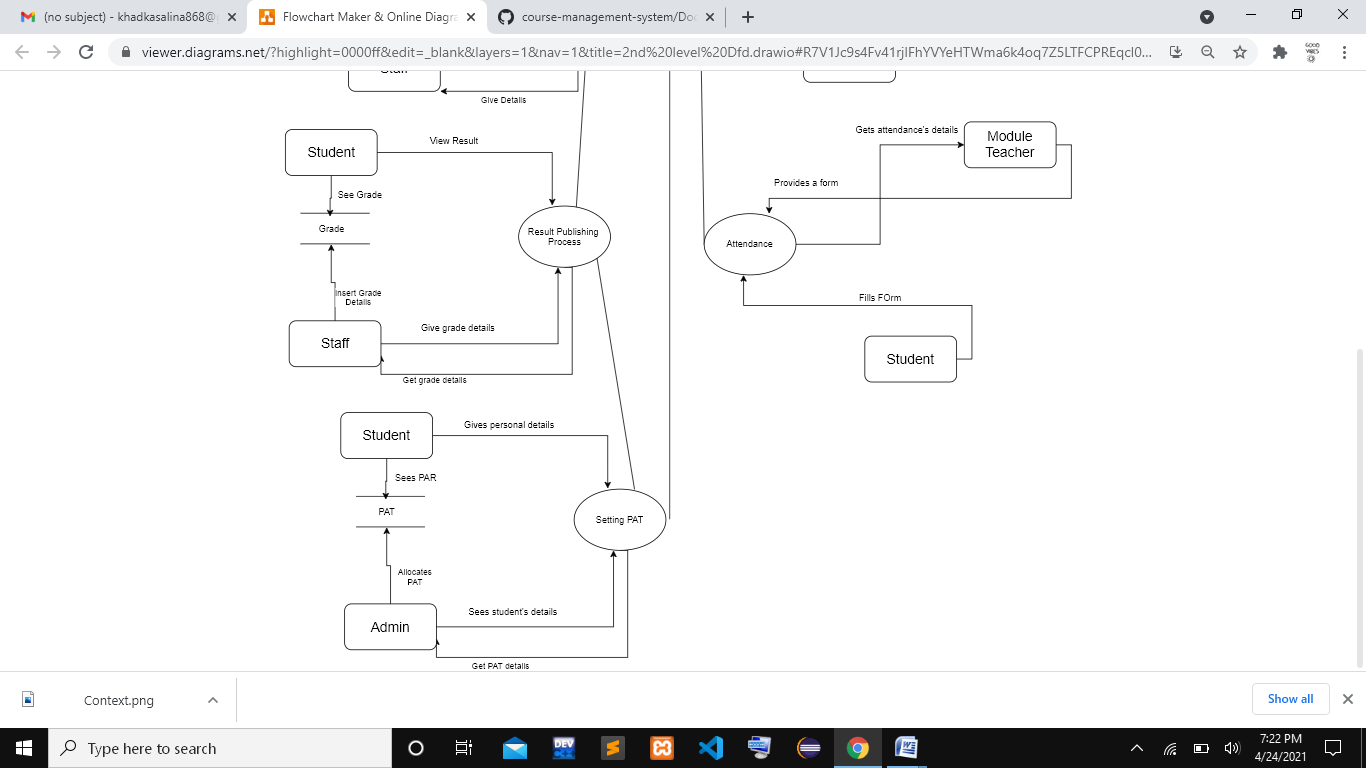
**Fig: Context Level DFD (0 Level)**

## First level data flow diagram

This is the First Level Data Flow Diagram of our system. It shows the different process and how it flows. It goes one step next deeper than the context one. Rounded Rectangle denotes entities, arrow determined the flow, circle denotes process and the two lines denotes database or file.

Below is the DFD of the Admin which describes the role of the admin. Admin is the supreme person in this project.

****

****

**Fig: Continuous screen shots of 1st Level Admin Part DFD**

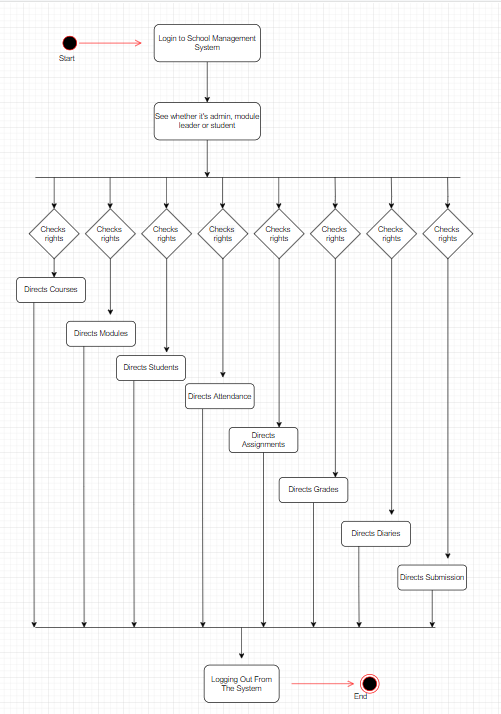
**Diagram

Description automatically generated**

**Fig: 2nd Level DFD – 5.0**

## Activity diagram

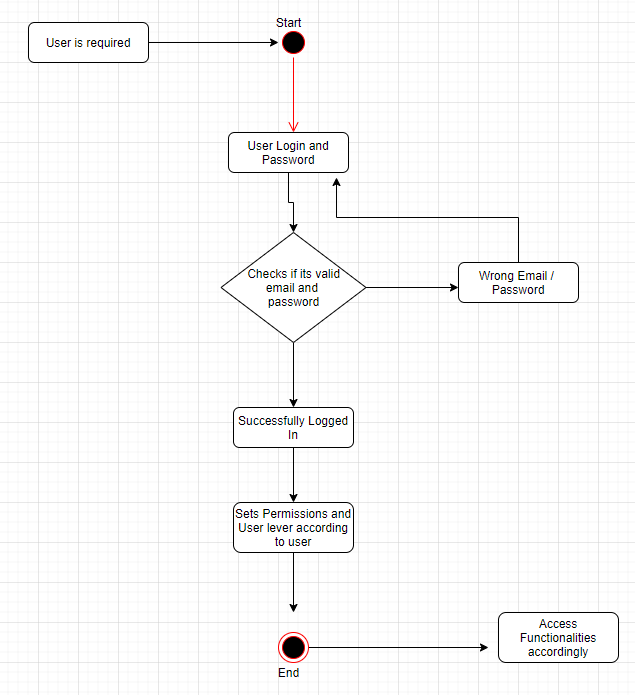
Below is the overall Activity Diagram of our system. After Login in by the respective email and passwords they will be verified as admin, module leader or student. Then, according to their permissions and level they can perform different type of tasks in the respective sector. It shows the complete flow of our system. They all are linked with each other by some means.

****

**Fig: Activity Diagram of SMS**

## Login activity diagram

Below is the Login Activity Diagram of our system, which is showing the flow of Login System. First there must be user with a certain email and password. User can be admin, module-leader or student. According to their position they can login and perform the tasks respectively. The diagram also shows the functionality of error handling. In order to access the features of our system, one must be logged in with their respective email and password.

****

**Fig: Login Activity Diagram of Login System**

## System database design

## E-R model

The ER (Entity Relationship) Diagram generally denotes the actual model of our system. It shows all the entities of our system- which is the real word objects, their attributes and their relationship between each other. Structured data is used. It helps to make a perfectly designed database for the system.

A picture containing diagram

Description automatically generated

## Sequence diagram

Sequence Diagram is a detailed and high-level interactive diagram which enlightens on how the functionality in a system is working. It is usually used to flush out the system’s design. What output is seen when what input is got. This is the basic function of Sequence Diagram. It consists of a participant which is generally an object, activation which is the total time needed to finish a task, lifelines and the messages and the output. Below is the sequence diagram of our few modules.

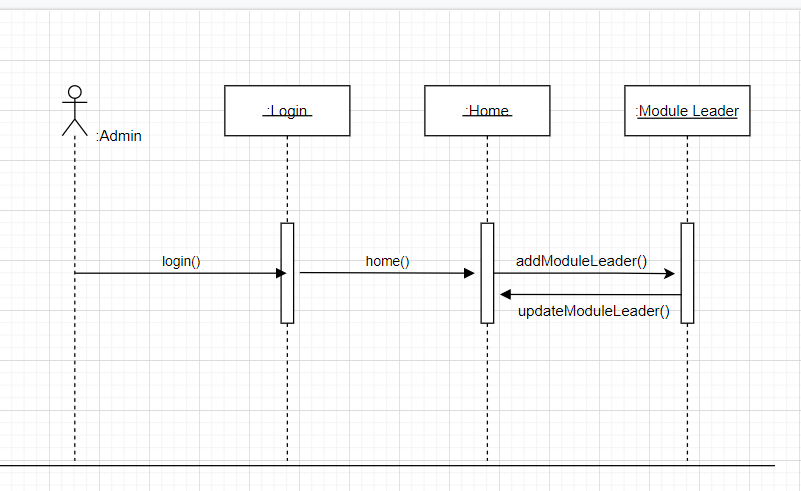


Fig: Sequence Diagram of for adding a Module Leader

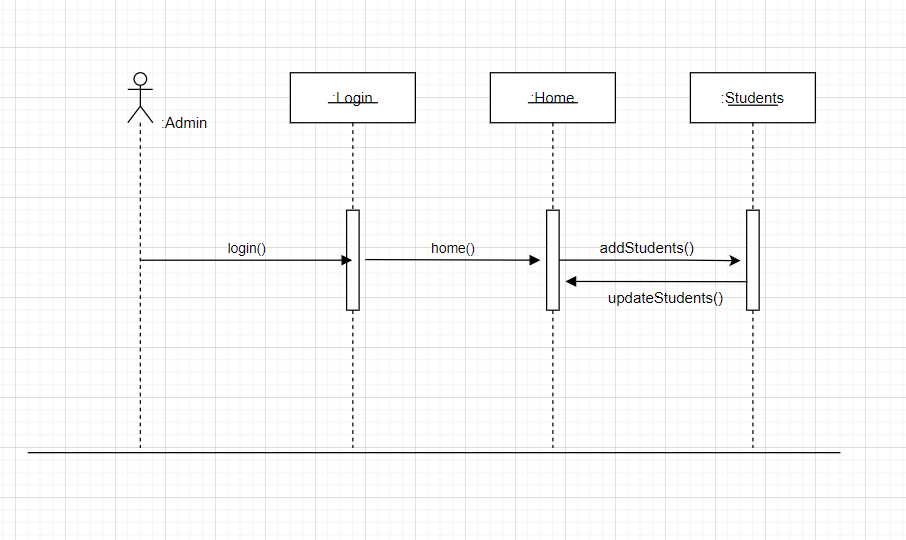


Fig: Sequence Diagram of for adding a Student

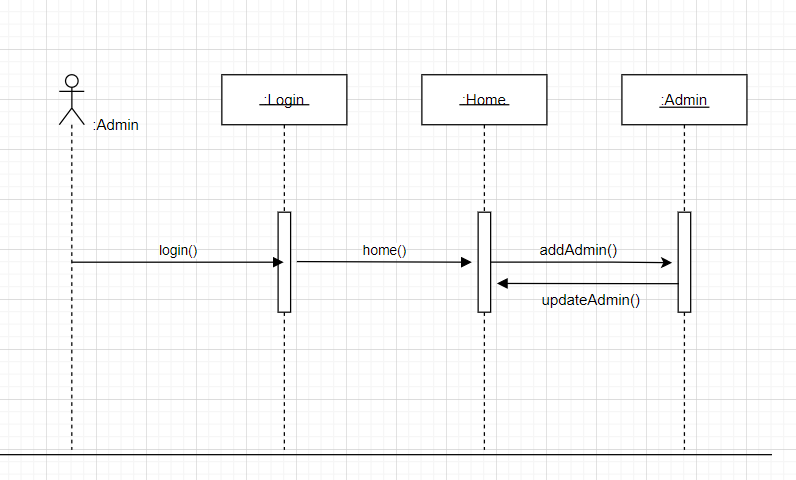


Fig: Sequence Diagram of for adding an admin

## 3.4.3. Attribute listing

|  |  |  |
| --- | --- | --- |
| S.no | Table Name | Attributes |
| 1. | assignments | assignment\_id, module\_id, deadline, content, semester |
| 2. | AttendanceModules | attendance\_modules\_id, module\_id,attendance\_time,semester,attendance\_status,week,class\_type |
| 3. | attendances | attendance\_id,attendance\_module\_id,student\_id,attendance\_time |
| 4. | Courses | course\_id, course\_head\_staff\_id, course\_name, start\_date |
| 5. | Diaries | diary\_id,student\_id,title,body,date\_created |
| 6. | Modules | module\_id,course\_id,module\_name,module\_credit,module\_level,ass\_1,ass\_2,exam |
| 7. | Routines | routine\_id,course\_id,day\_number,day,module\_id,start\_time,end\_time,second\_module\_name,second\_start\_time,second\_end\_time,semester,class\_type\_1,class\_type\_2 |
| 8. | Staff | staff\_id,name,surnameemaill, address,date\_of\_join,course\_id,module\_id,salary,role,password |
| 9. | Students | student\_id, course\_id, name,email, password, surname, address, phone, gender, date\_of\_birth, registration\_year,student\_status |
| 10. | Submissions | submission\_id,student\_id,assignment\_id, submission\_date,content |

# System interface design

This part of the report includes user interface and experience of Woodlands University College’s new course management system with step wise design process. The focus was given to minimalism and usability throughout the design process to reduce time consumption to perform any tasks while also increasing creativity and awareness of students as well as staffs. The major functions of the apps are accessible with fewer UI interactions and are easier to navigate. The following technologies is to be used for designing the final frontend web application:

• HTML

• CSS

• JavaScript

• React

• Nodejs

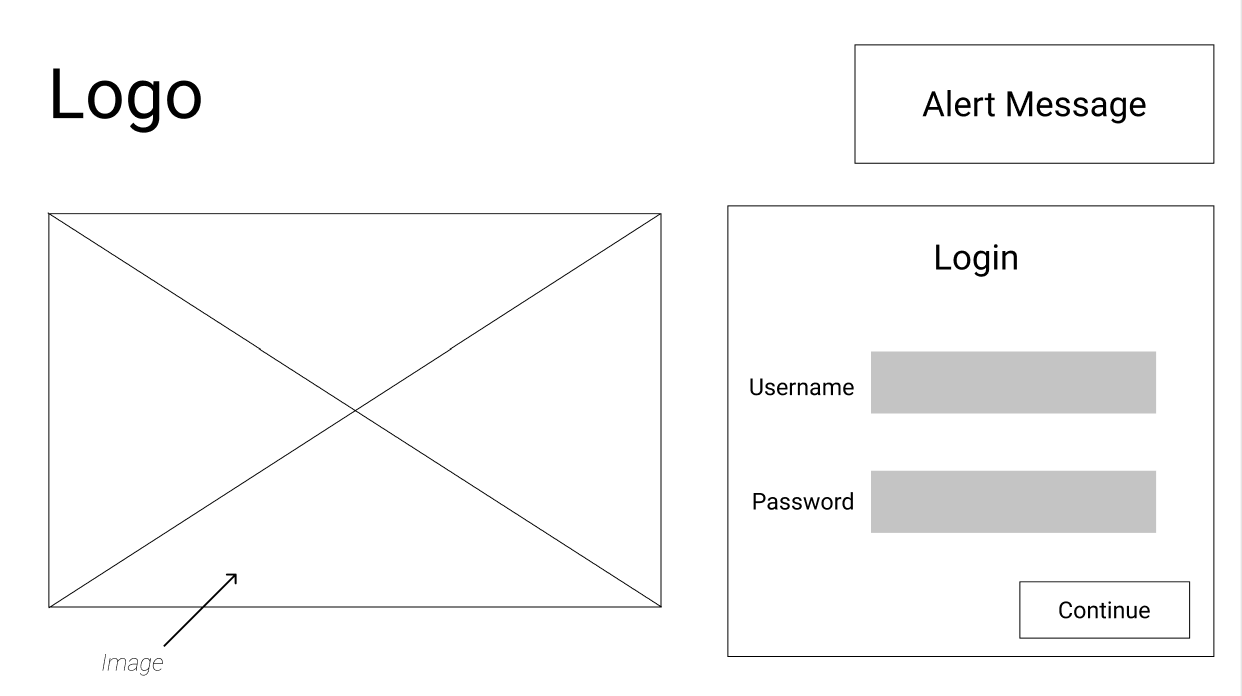
## Draft Interface Designs

This section includes drafts of visual elements to describe the proposed interface design including wireframes, system navigation diagrams, mockup designs and systemactivity event diagrams.

## Wireframes

Wireframe designs of the application is included in this section. The design concept was drawn using an online tool—Figma (*Figma,2021*), by consulting the client about the functionality and the expectations.

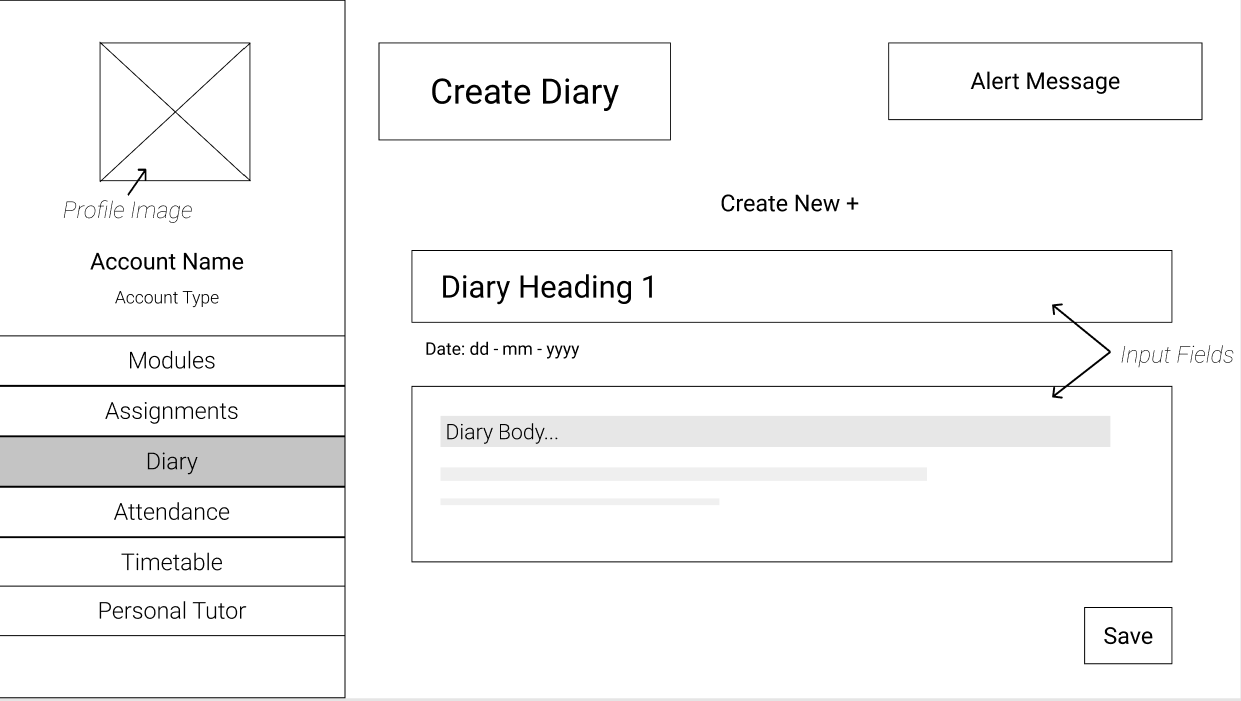
#### Login Screen



#### Diary List

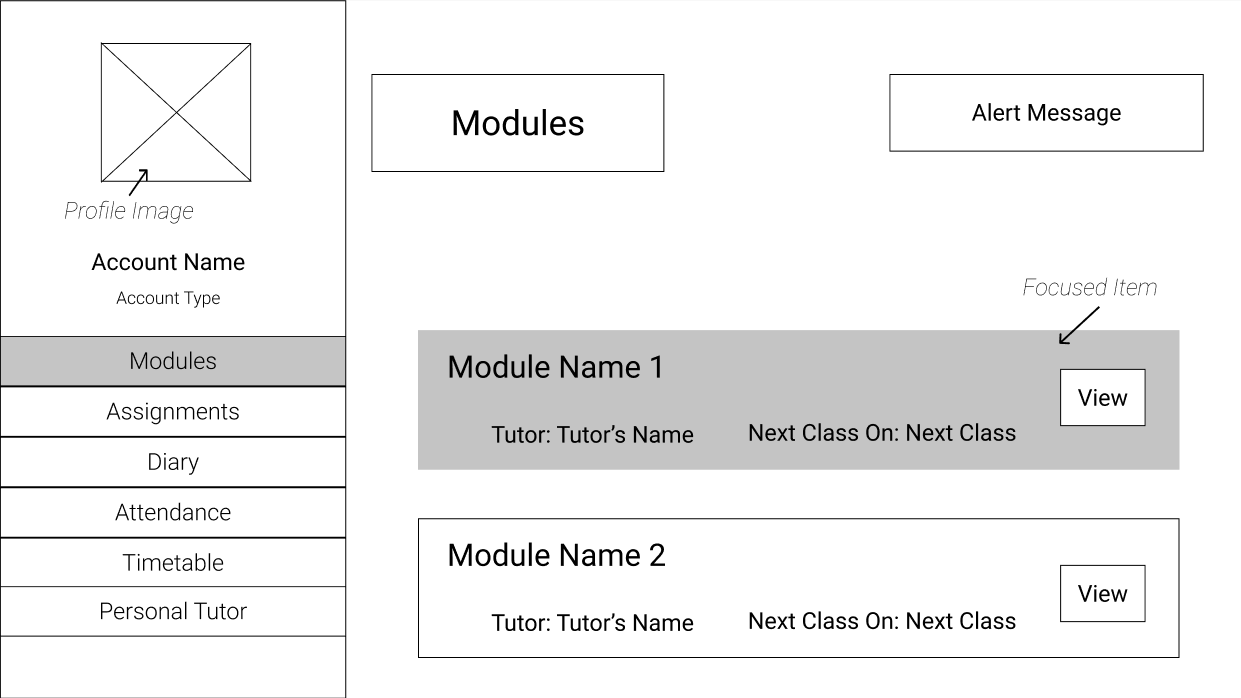


#### Create New Diary

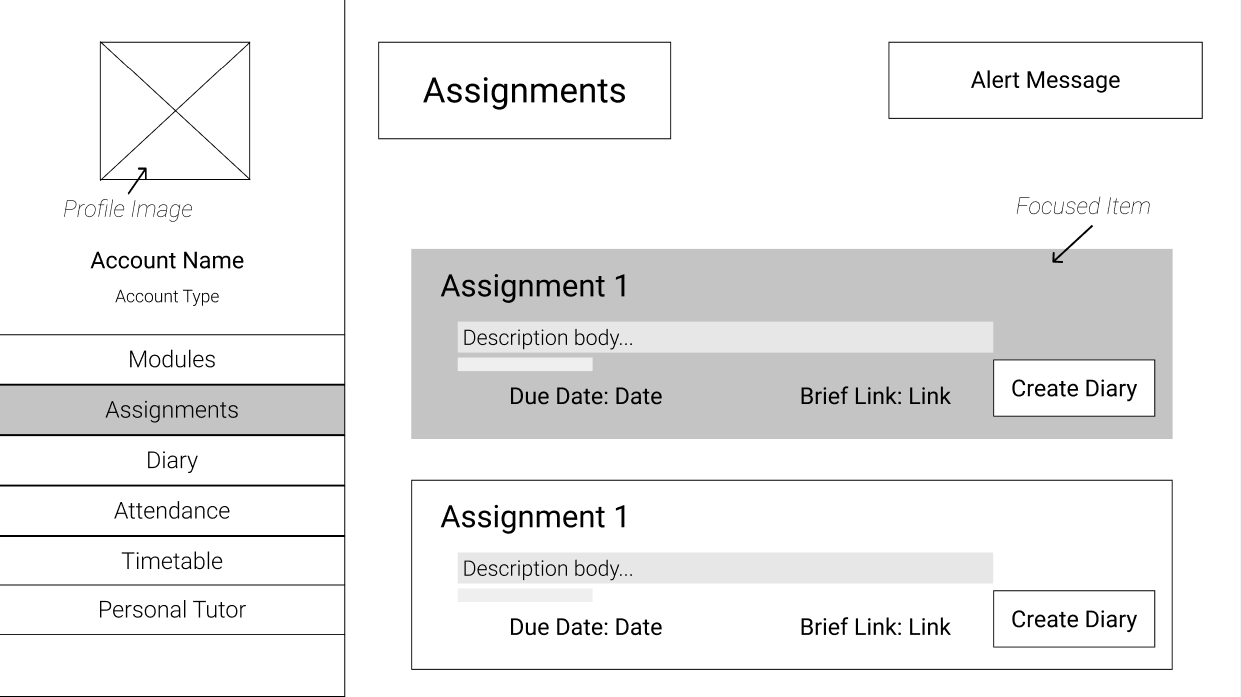


#### Student’s View

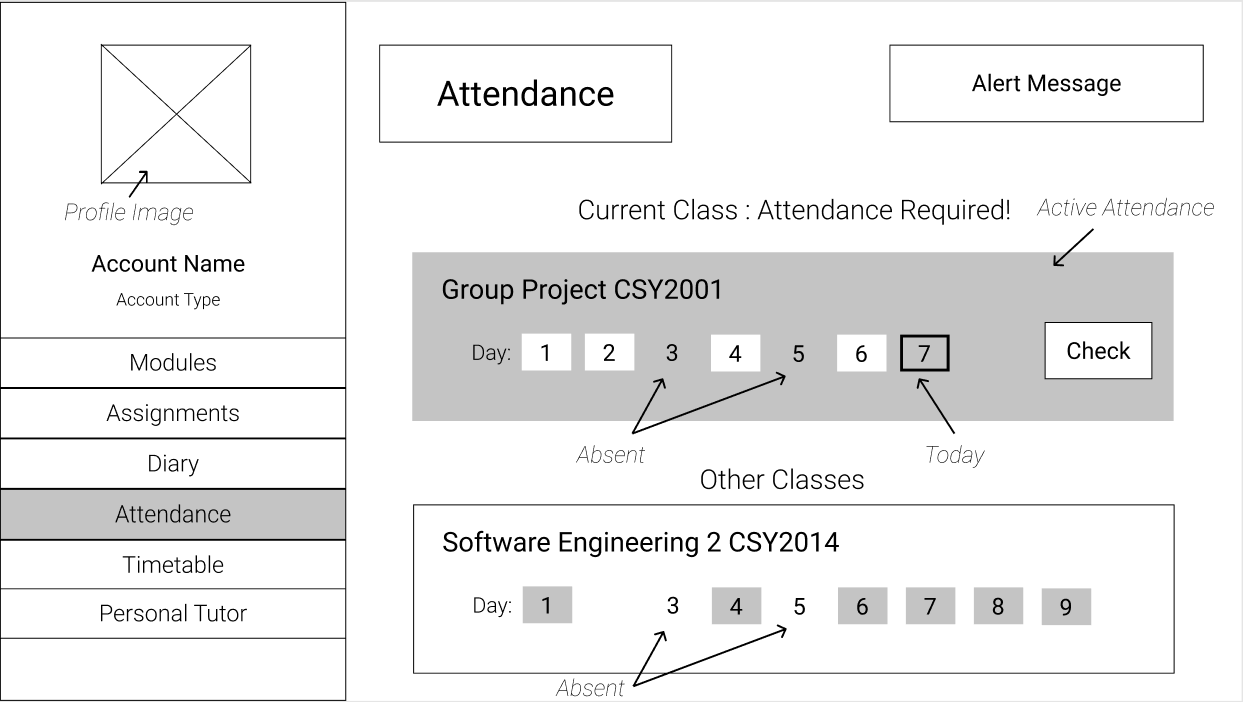
##### ***Modules***



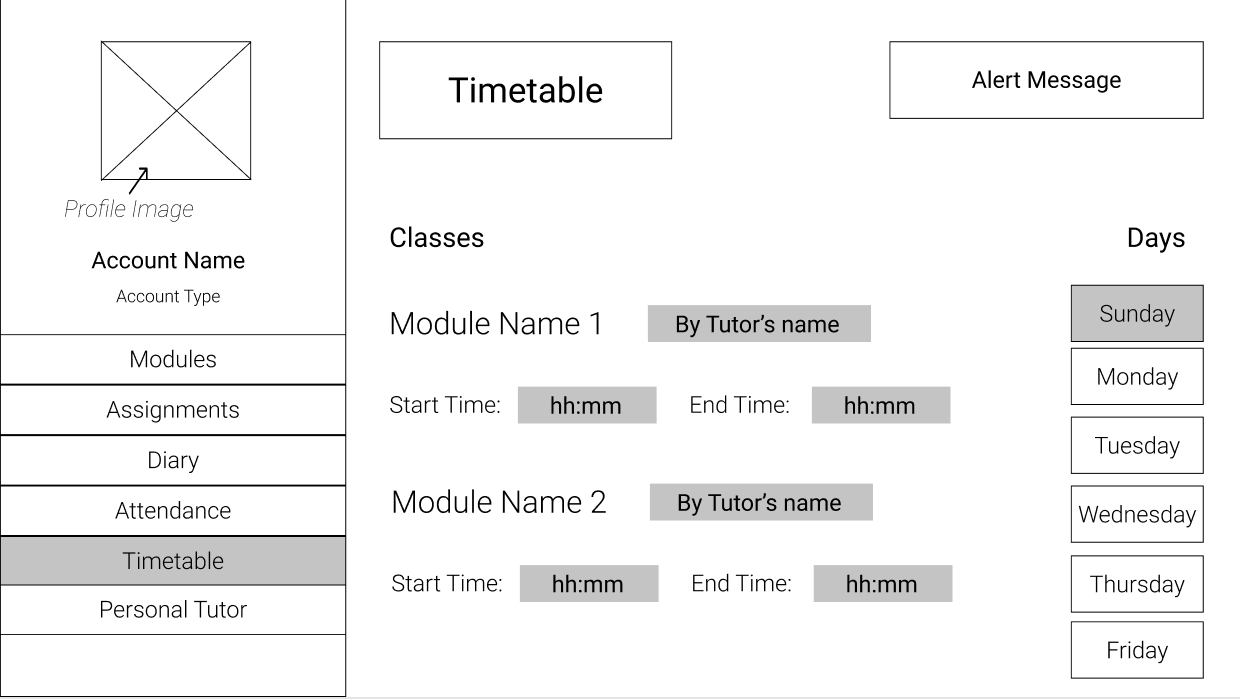
##### *Assignments*



##### *Attendance*

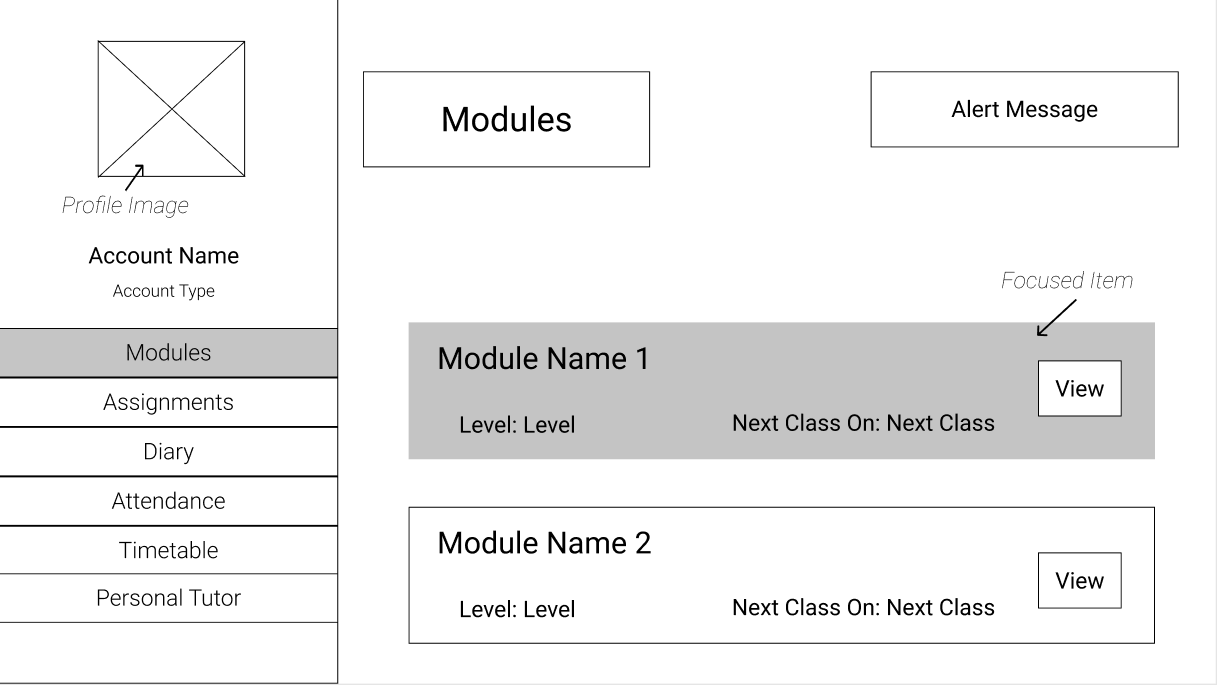


##### *Timetable*

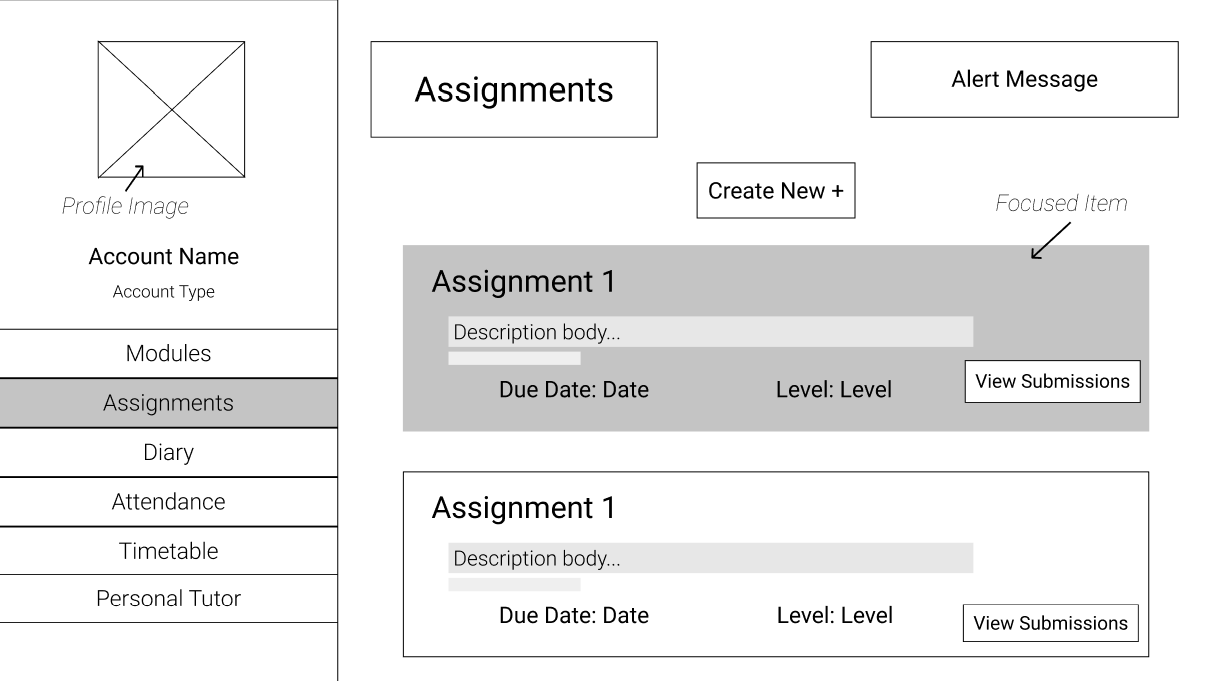


#### Module Leader’s View

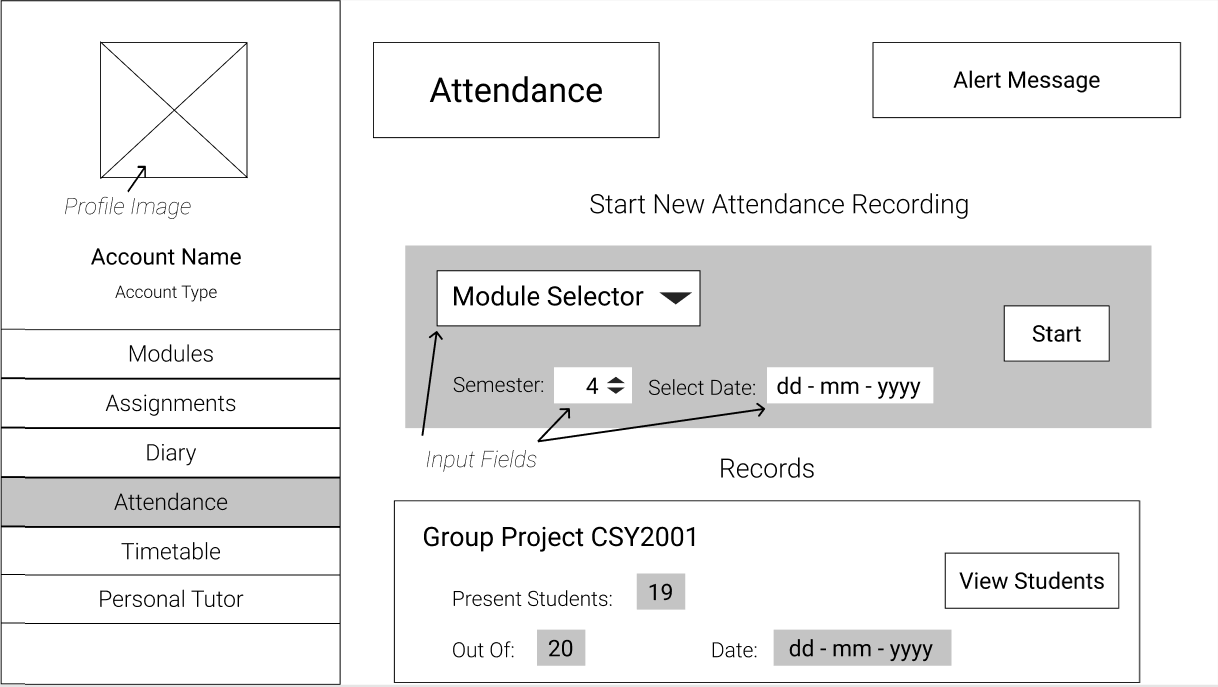
##### ***Modules***



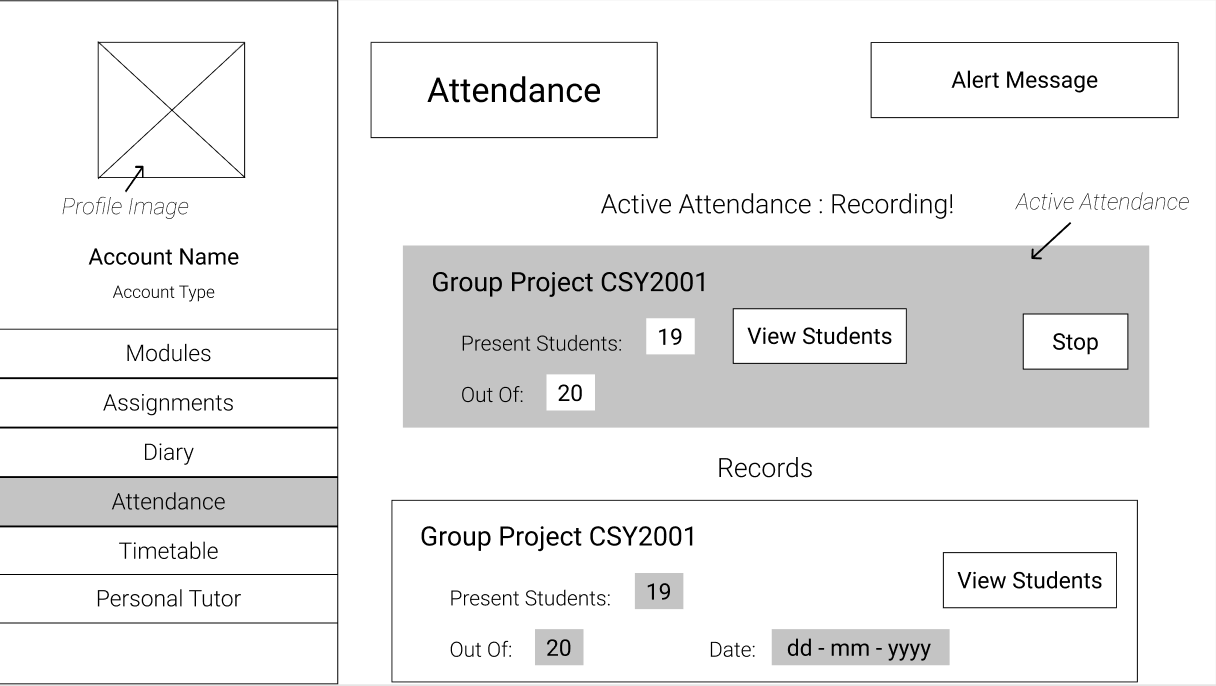
##### *Assignments*



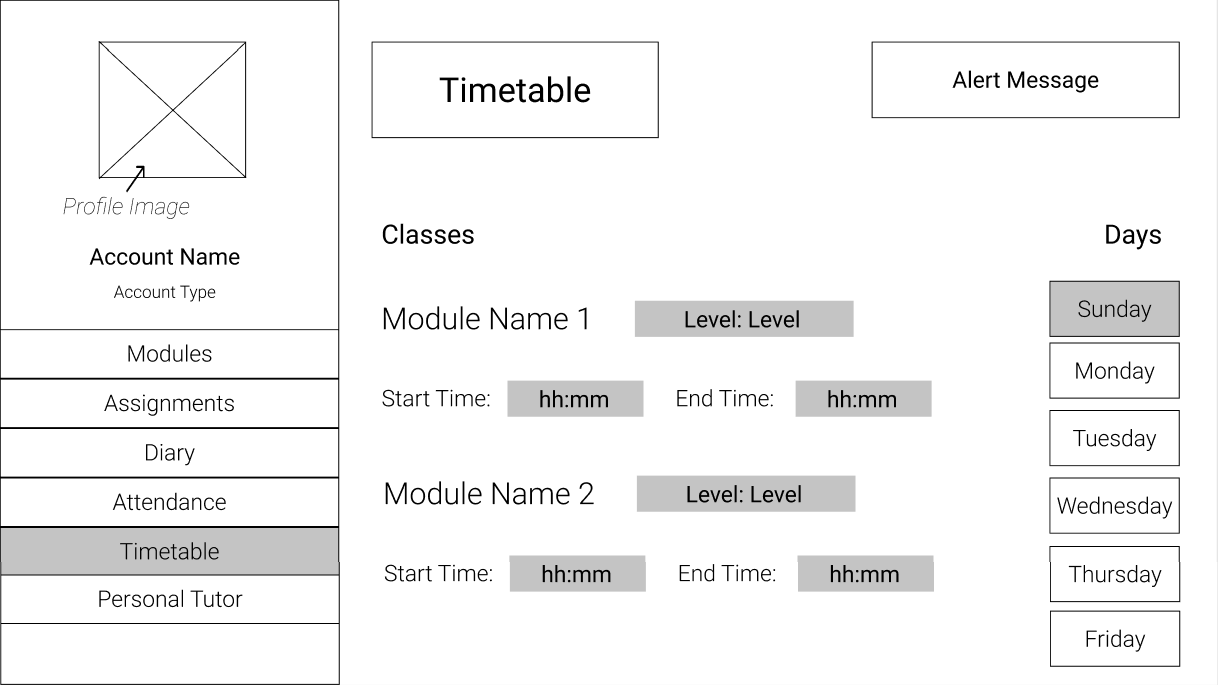
##### *Attendance List*



##### *Attendance Recording*



##### *Timetable*

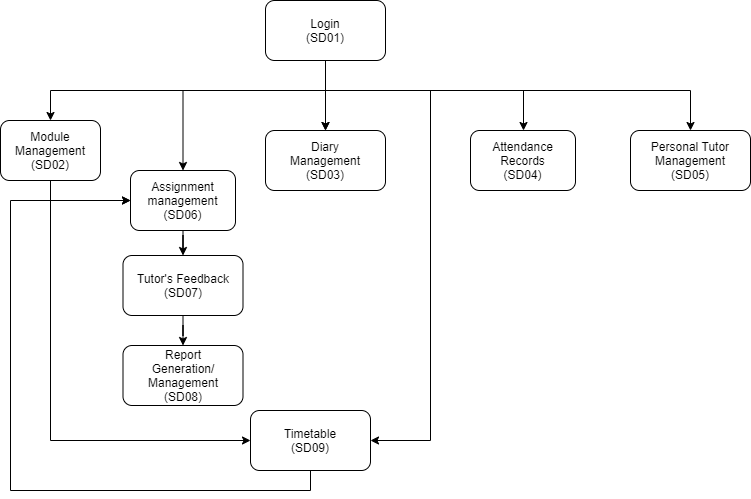


##### 

## System Navigation Diagram

The visual representation of flow of screens during navigation of thecourse management system is included in this section. The web application can either be used by staff or students. Each screen is represented by a box and is represented with the screen title and a screen number.

#### For Students

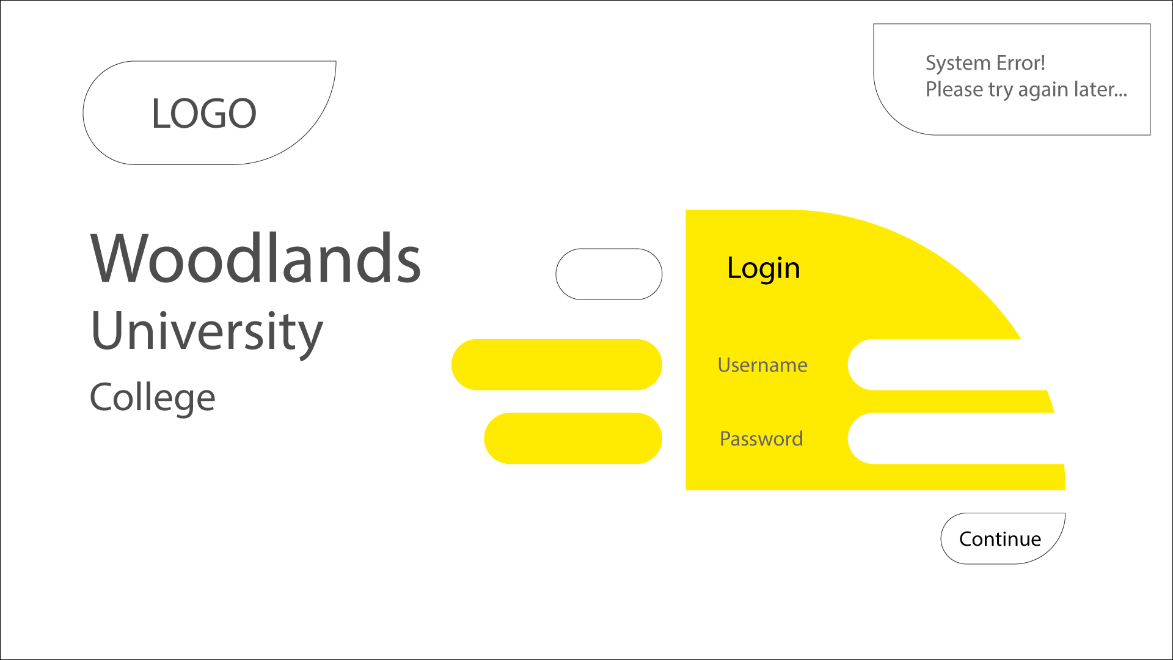


#### ssFor Admin



## System Screen Mock-ups

#### Login Screen – Done

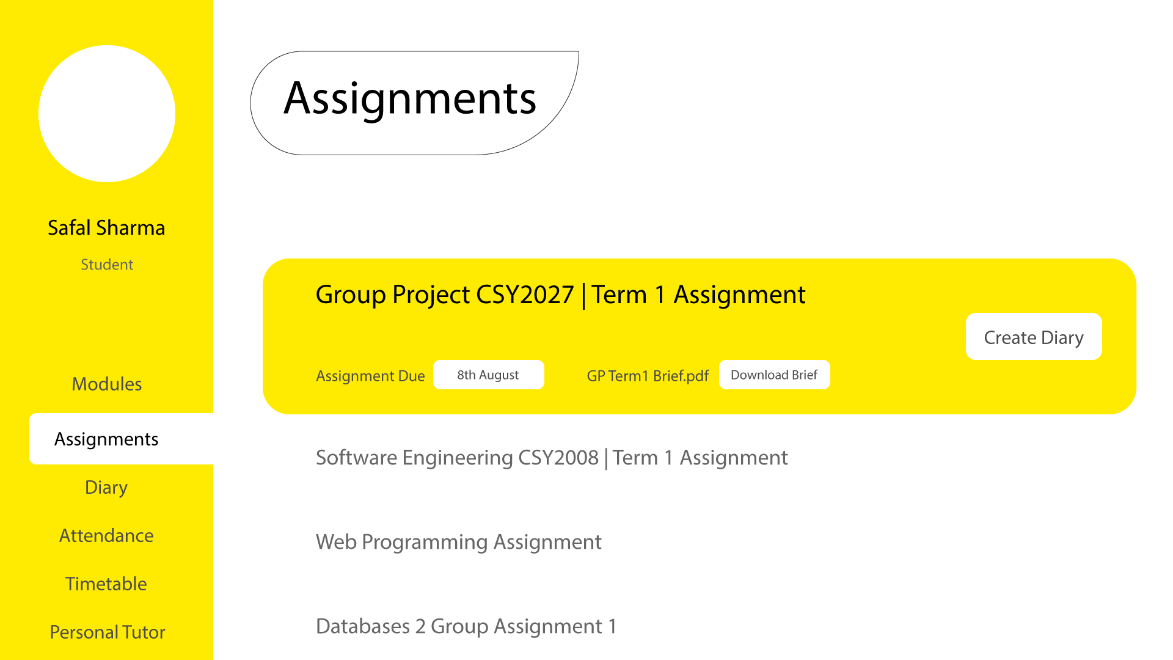


#### Student’s View

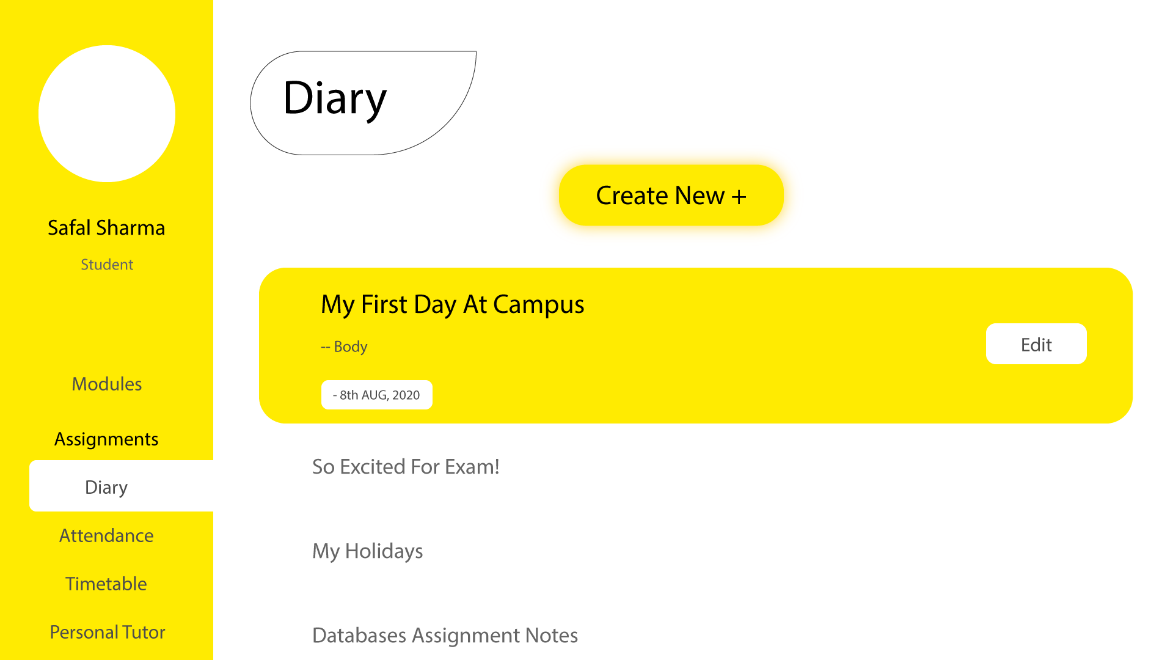
##### Modules View -Done

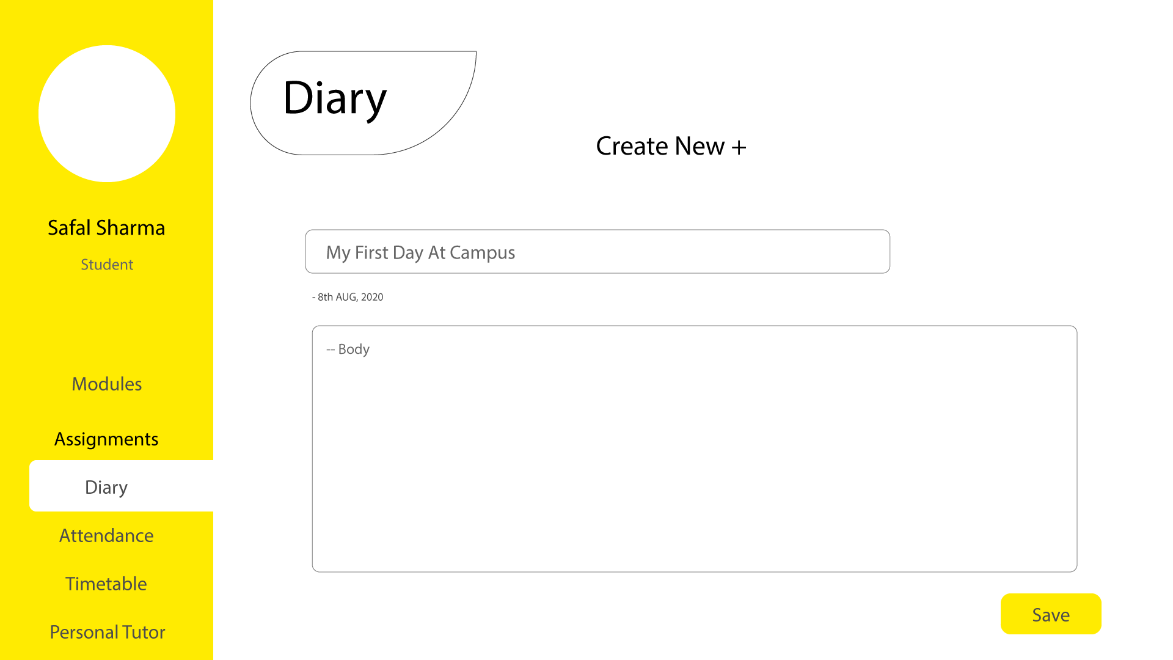


##### Assignments – Done (--TODO Tutor’s Feedback)

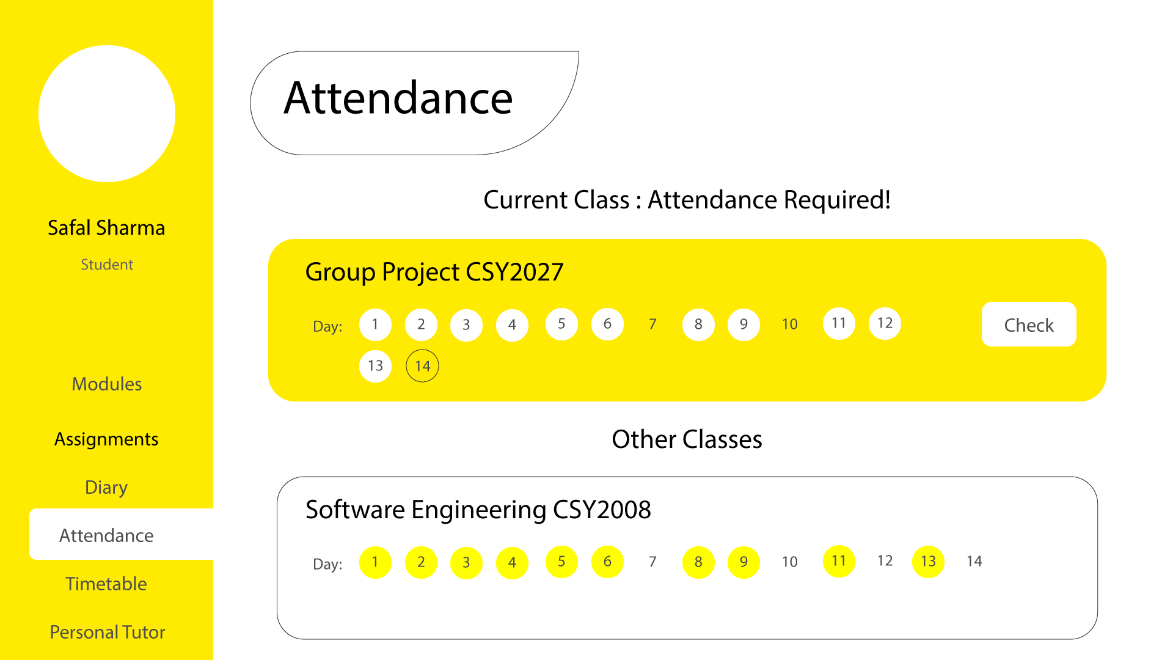


##### Diary– Done

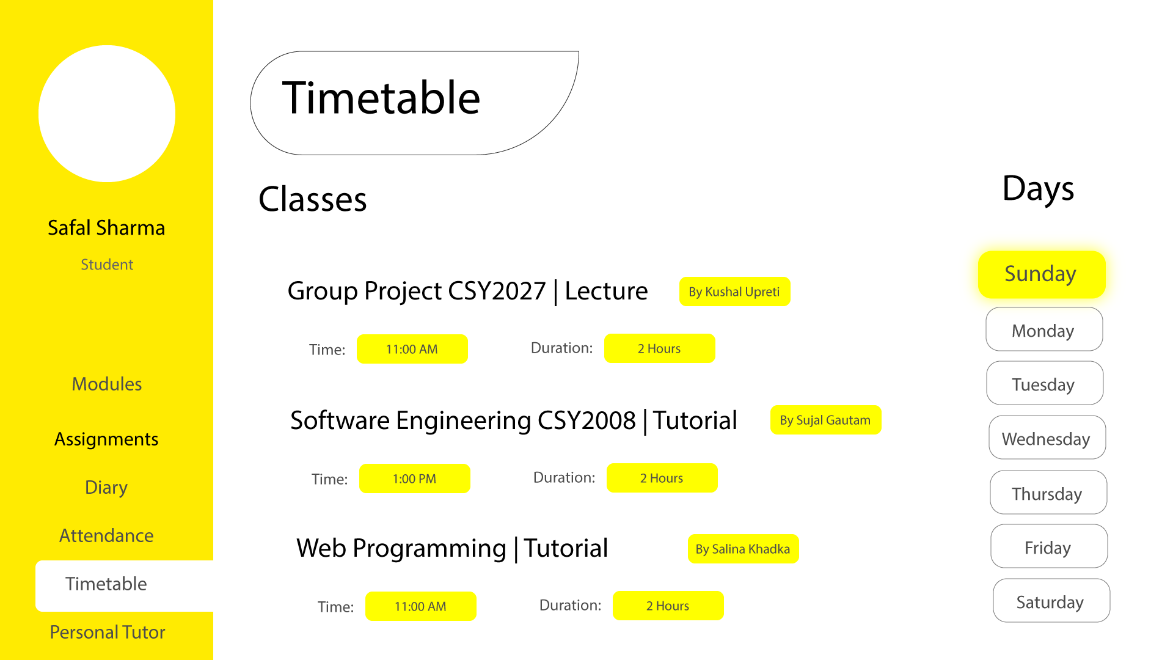




##### Attendance -Done



##### Timetable -Done

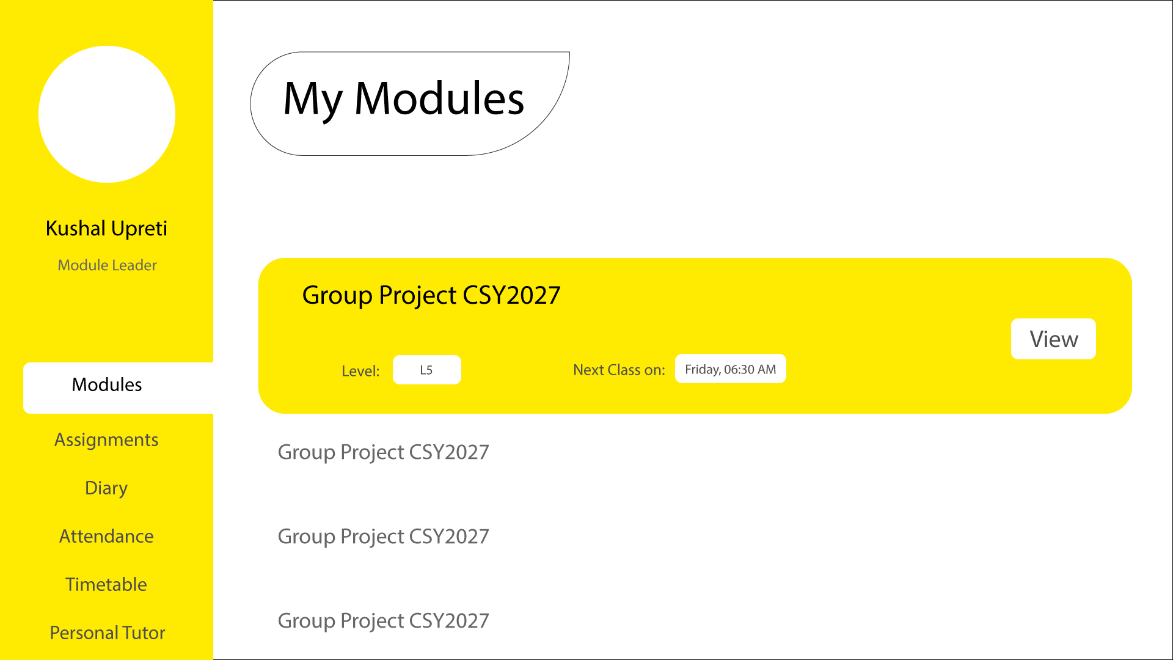


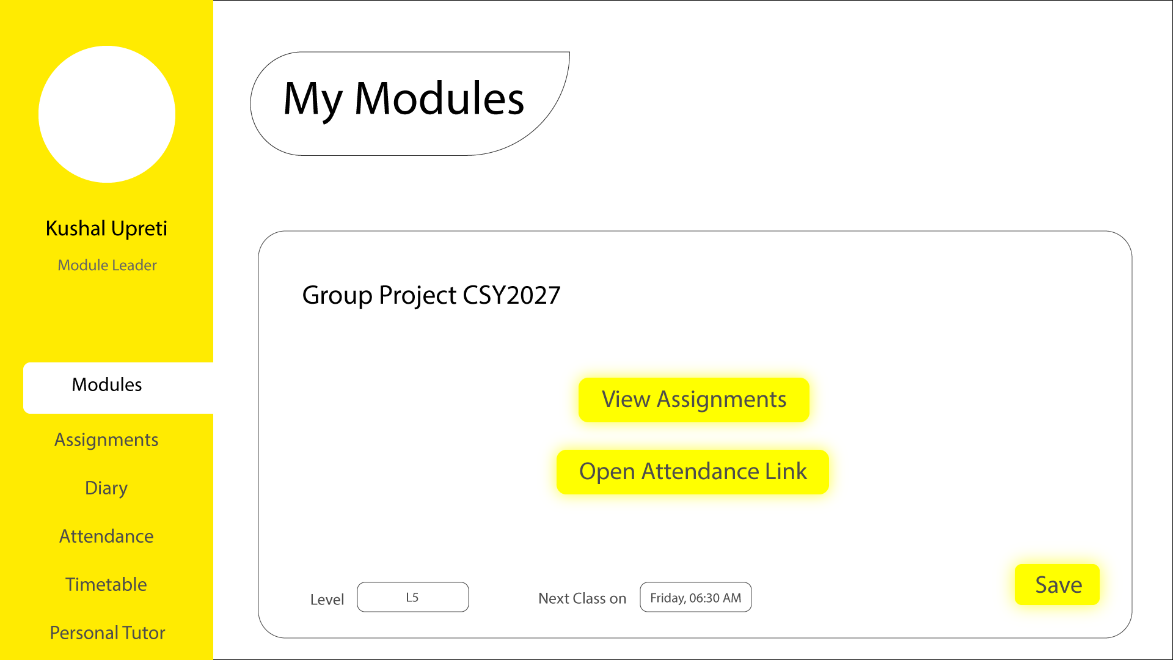
##### Personal Tutor -TODO

##### Tutor’s Feedback -TODO

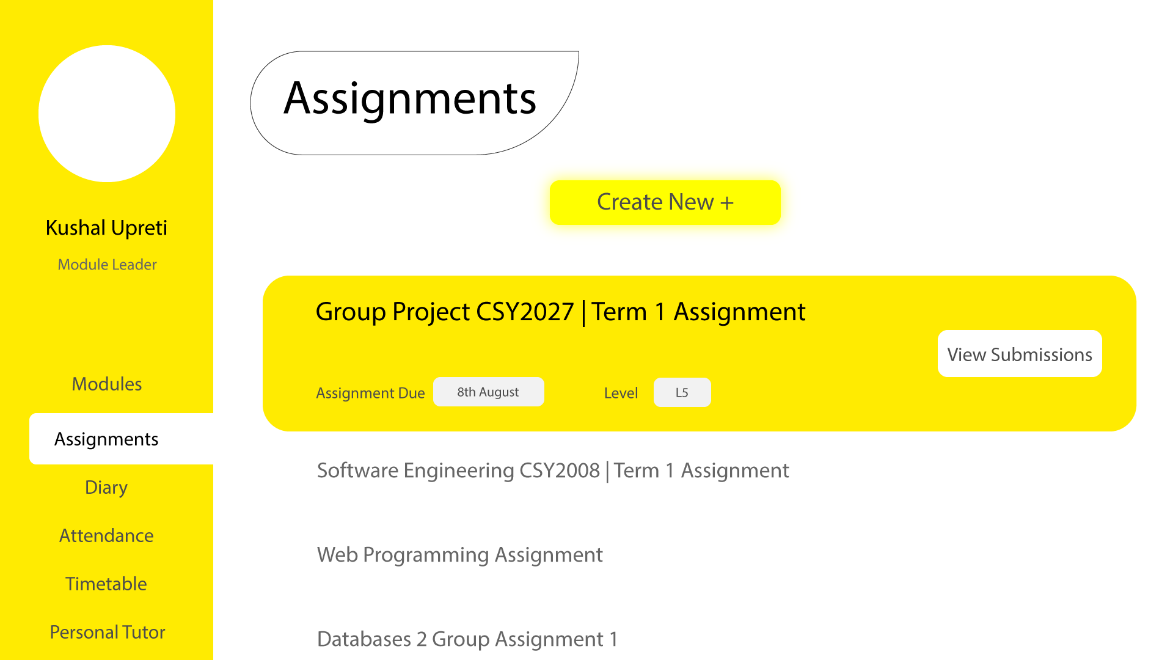
#### Module Leader’s View

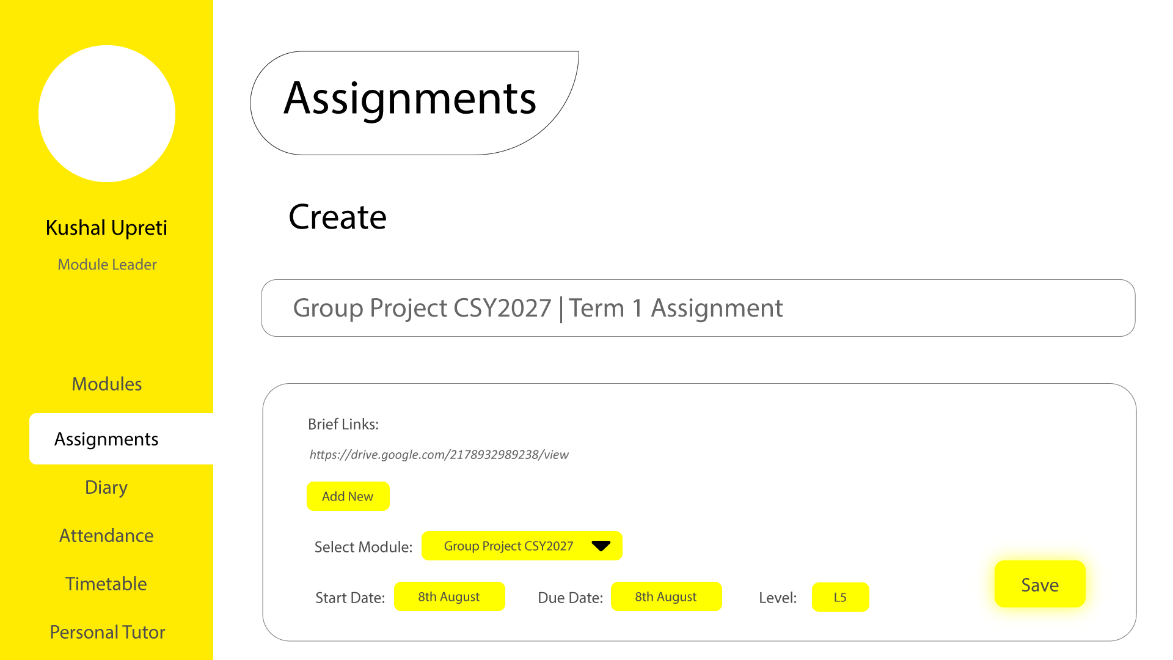
##### Modules View -Done



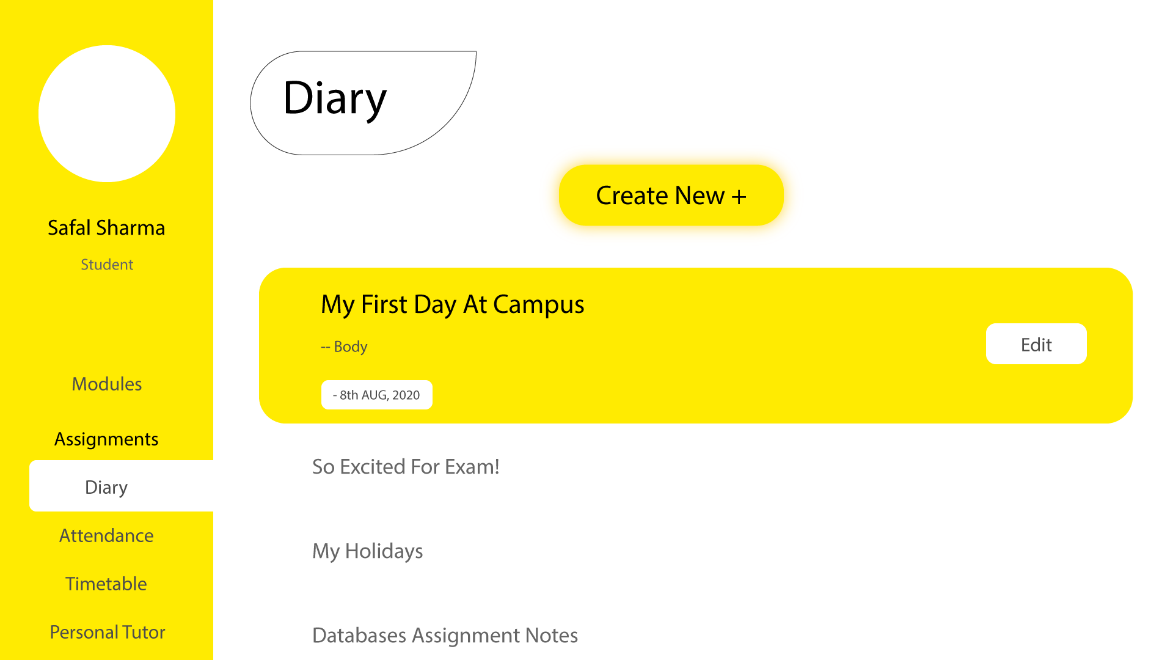


##### Assignments – Done(--TODO Tutor’s Feedback)

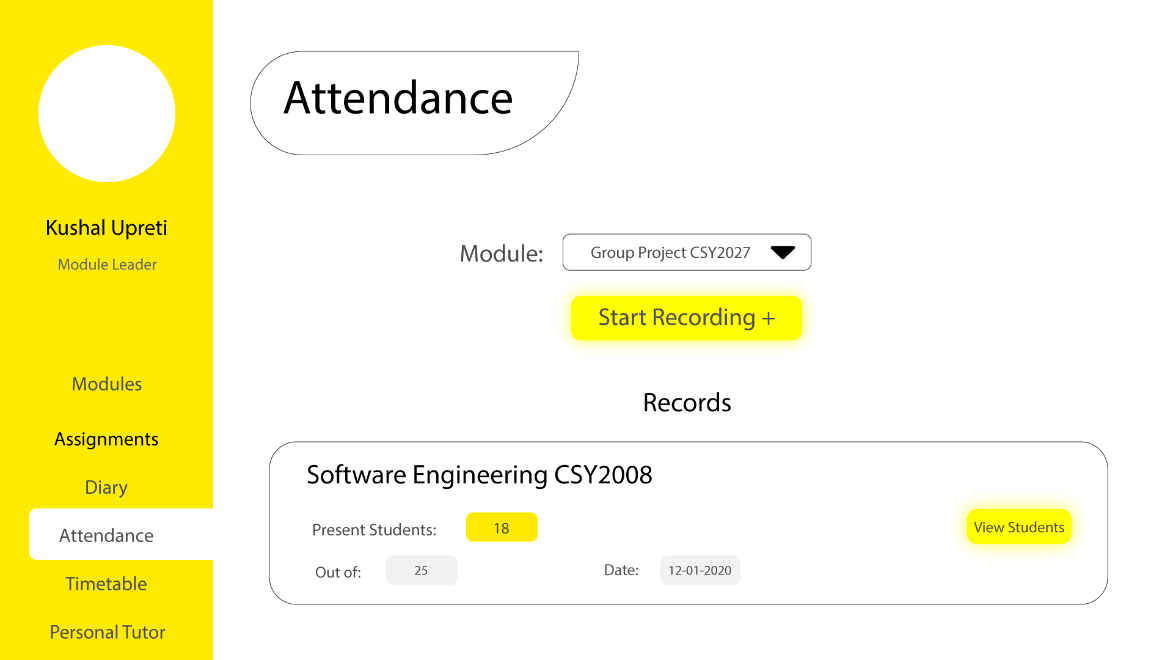


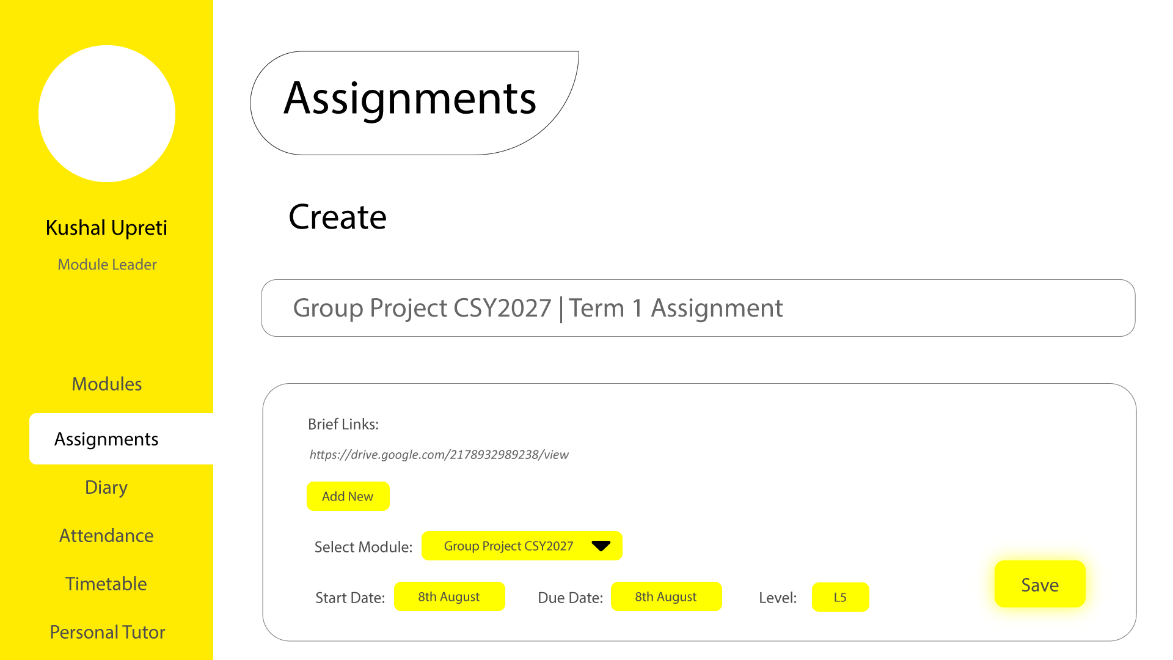


##### Diary – Done



##### Attendance -Done





##### Timetable -TODO

##### Personal Tutor -TODO

##### Tutor’s Feedback -TODO

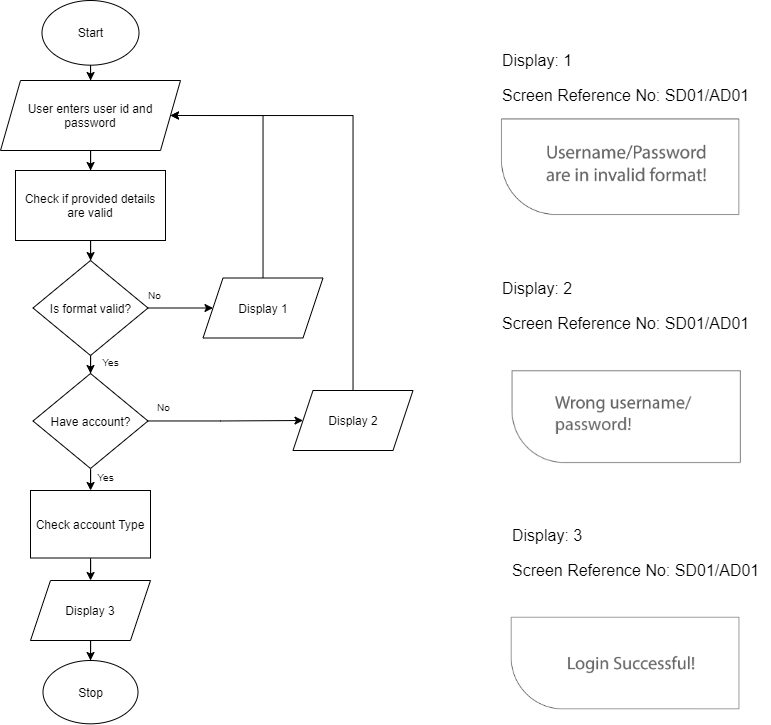
## System Activity Event Diagrams

This section of the report contains activity event diagrams of important eventsin the application.

#### Login

Activity Code: 001

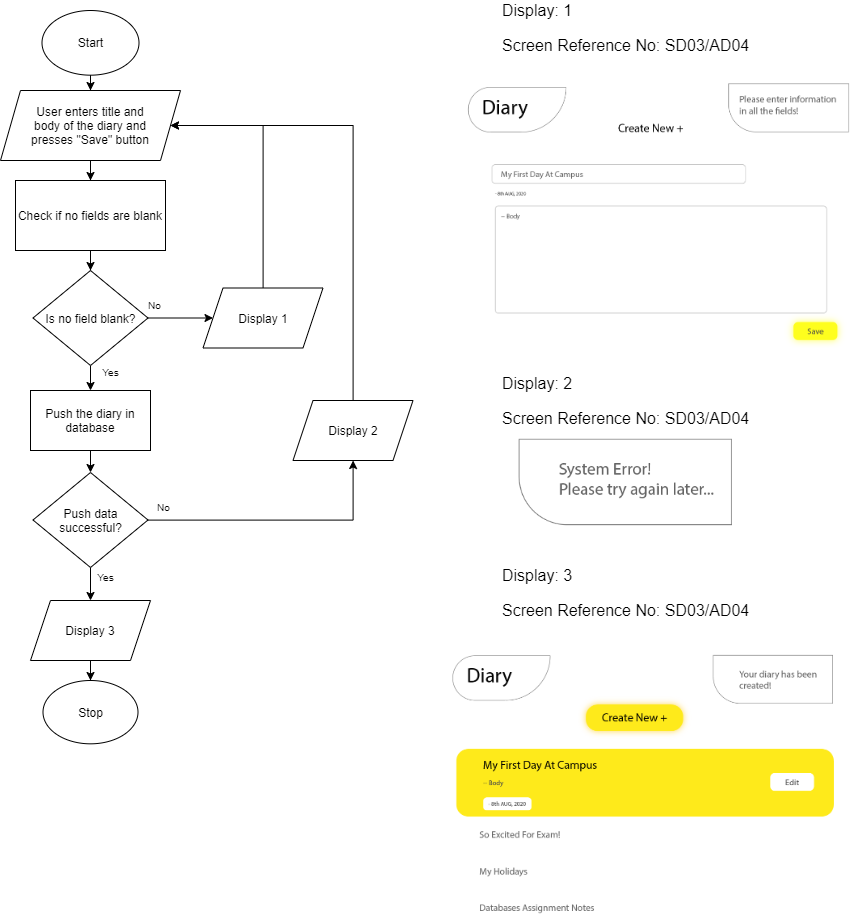
Activity Name: Login



#### Create Diary

Activity Name: Create Diary

Activity Code: 002



## Design Revisions

Various design revisions have been made through effective collaboration with the project stakeholders. Suggestions from visitors and non-associated personals were also welcomed. The key design changes during the development of the project are:

1. Removal of unnecessary elements and focus on minimalism.
2. Primary color was changed from Yellow to Green after several discussions with the team members and stakeholders.
3. [TODO]

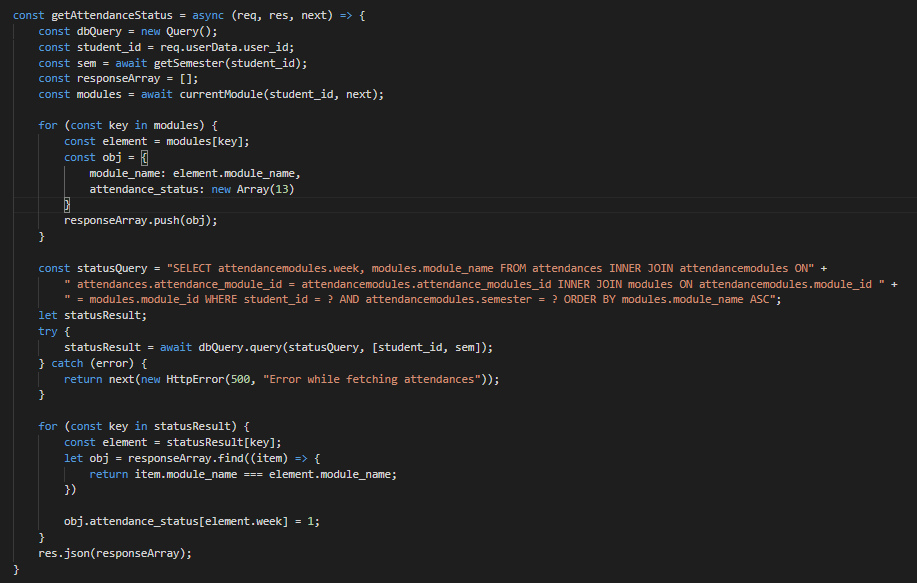
## Heuristic Evaluation

[TODO]

# 5. **System Build and Technical Notes**

## **5.1. Technically Difficult Code Sections**

## 5.1.1 Student Attendance



**Functions**

* Opening attendance form that students can click to attend.
* Displaying attendance status for current modules for a student.
* Viewing all the present students for a week for a given module by a tutor.

## 5.1.2. Assignment submission/grading



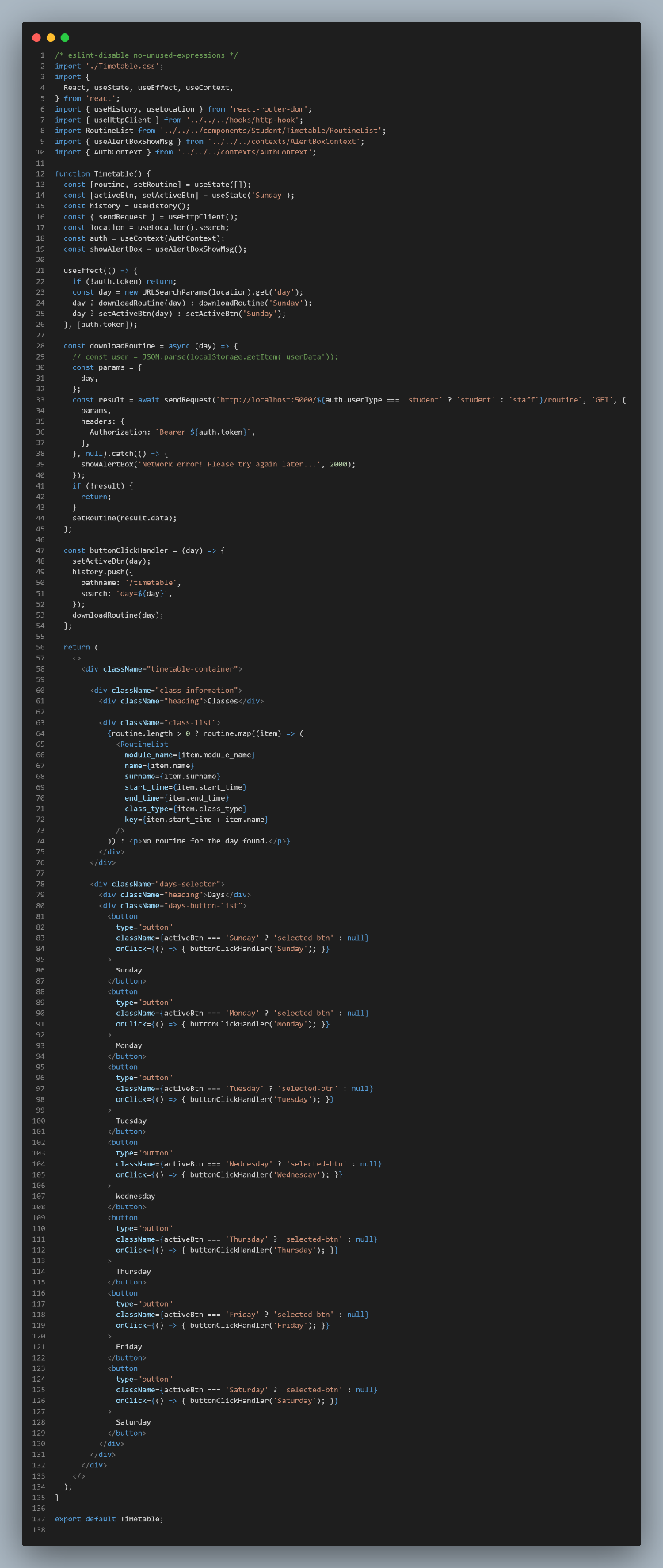
**Functions**

* Create assignment with a set deadline by the tutor.
* Submit assignments multiple times by the student before the deadline.
* Grade or provide feedbacks by the tutor for an assignment submission.

## 5.1.3. Timetable (Student/Tutor)

**Functions**

* Create student/tutor timetable by the admin.
* View the student timetable for the current semester.
* View the periods to be taught by a tutor.



# 7. **System evaluation**

## 7.1. Usability evaluation strategy

After completing the project, we evaluated it. Evaluation is the process of valuing thing using a certain method. For that we asked some people to use it. We asked them some question and their thoughts regarding our system. We recorded their genuine thoughts based on their experience. We asked them series of questions which are as follows:

1. How good is our school management system?
2. How fast was the response?
3. Did we implement our functionalities in a good way?
4. How easy is our system?
5. Did you find any backwards in our project?

## 7.2. System plot trails

**Person 1:**

How good is our school management system?

|  |  |
| --- | --- |
| A+ |  |
| A | ✔ |
| B+ |  |
| B |  |
| C |  |
| D |  |

How fast was the response?

|  |  |
| --- | --- |
| A+ | ✔ |
| A |  |
| B+ |  |
| B |  |
| C |  |
| D |  |

Did we implement our functionalities in a good way?

|  |  |
| --- | --- |
| Yes | ✔ |
| No |  |
| Could be much better |  |

How easy is our system?

|  |  |
| --- | --- |
| Super Easy | ✔ |
| Easy |  |
| Hard |  |
| Super Hard |  |

Did you find any backwards in our project?

|  |  |
| --- | --- |
| Yes | ✔ |
| No |  |

**Person 2:**

How good is our school management system?

|  |  |
| --- | --- |
| A+ | ✔ |
| A |  |
| B+ |  |
| B |  |
| C |  |
| D |  |

How fast was the response?

|  |  |
| --- | --- |
| A+ | ✔ |
| A |  |
| B+ |  |
| B |  |
| C |  |
| D |  |

Did we implement our functionalities in a good way?

|  |  |
| --- | --- |
| Yes | ✔ |
| No |  |
| Could be much better |  |

How easy is our system?

|  |  |
| --- | --- |
| Super Easy |  |
| Easy | ✔ |
| Hard |  |
| Super Hard |  |

Did you find any backwards in our project?

|  |  |
| --- | --- |
| Yes |  |
| No | ✔ |

**Person 3:**

How good is our school management system?

|  |  |
| --- | --- |
| A+ |  |
| A |  |
| B+ | ✔ |
| B |  |
| C |  |
| D |  |

How fast was the response?

|  |  |
| --- | --- |
| A+ |  |
| A | ✔ |
| B+ |  |
| B |  |
| C |  |
| D |  |

Did we implement our functionalities in a good way?

|  |  |
| --- | --- |
| Yes | ✔ |
| No |  |
| Could be much better |  |

How easy is our system?

|  |  |
| --- | --- |
| Super Easy | ✔ |
| Easy |  |
| Hard |  |
| Super Hard |  |

Did you find any backwards in our project?

|  |  |
| --- | --- |
| Yes |  |
| No | ✔ |

**Person 4:**

How good is our school management system?

|  |  |
| --- | --- |
| A+ | ✔ |
| A |  |
| B+ |  |
| B |  |
| C |  |
| D |  |

How fast was the response?

|  |  |
| --- | --- |
| A+ | ✔ |
| A |  |
| B+ |  |
| B |  |
| C |  |
| D |  |

Did we implement our functionalities in a good way?

|  |  |
| --- | --- |
| Yes |  |
| No |  |
| Could be much better | ✔ |

How easy is our system?

|  |  |
| --- | --- |
| Super Easy |  |
| Easy | ✔ |
| Hard |  |
| Super Hard |  |

Did you find any backwards in our project?

|  |  |
| --- | --- |
| Yes |  |
| No | ✔ |

## 7.3 System Trial Results

|  |  |  |
| --- | --- | --- |
| S. No | Respective System Evaluation | Average Results |
| 1. | Usability | A |
| 2. | Functionalities | A |
| 3. | Experience | A |
| 4. | Speed | A |
| 5. | Backwards | D |
| 6. | Quality | A |
| 7. | Implementation | B |

## 7.4 Project Work Log

s

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task | Team Member’s Contribution | | | |
| Introduction | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| Project Background | 50% |  |  | 50% |
| Project Aims and Objectives |  | 50% | 50% |  |
| Project Development Methodology |  | 100% |  |  |
| Requirements Engineering | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| Interview Plans | 25% | 25% | 25% | 25% |
| Interview findings | 50% |  |  | 50% |
| Other Problem Domain Research |  | 60% |  | 40% |
| Comparable Software System Review |  | 50% | 50% |  |
| Development Relevant Legislation | 50% |  |  | 50% |
| Academic Literature Review |  | 50% | 50% |  |
| Any other relevant problem domain investigation data | 50% |  |  | 50% |
| Requirements Specification | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| Problem Domain Description |  | 50% | 50% |  |
| Existing Business Operation | 50% |  |  | 50% |
| Summary of existing business limitations requiring resolution |  | 100% |  |  |
| Functional Requirements |  |  |  | 100% |
| Performance Requirements | 100% |  |  |  |
| Design Constraints |  |  | 100% |  |
| Commercial Constraints |  | 50% |  | 50% |
| Acceptance Tests |  | 100% |  |  |
| System Analysis and Design | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| Preliminary Design Stages | 50% |  |  | 50% |
| Textual Analysis |  | 50% | 50% |  |
| Significant Event Analysis | 50% |  |  | 50% |
| Commands Queries and Constraints |  | 50% | 50% |  |
| Detailed Static System Designs | 25% | 25% | 25% | 25% |
| Detailed Dynamic System Designs | 25% | 25% | 25% | 25% |
| System Database Design | 50% |  | 50% |  |
| System Interface Design | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| Draft Interface Designs | 25% | 25% | 25% | 25% |
| Design Revisions |  |  | 100% |  |
| Heuristic Evaluation |  | 50% |  | 50% |
| System Build and Technical Notes | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| Technically Difficult Code Sections | 100% |  |  |  |
| Final System Interface Displays |  |  |  | 100% |
| Test Strategy | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| Overview of Test Strategy |  | 50% | 50% |  |
| Sample Test Results |  | 50% | 50% |  |
| System Evaluation | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| System Pilot Trials | 50% |  |  | 50% |
| System Trial Results | 50% |  |  | 50% |
| Project Conclusions | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| Project Management | 100% |  |  |  |
| Project Gantt Chat |  | 100% |  |  |
| Project Meeting Minutes |  |  | 100% |  |
| Project Quality Plan |  |  |  | 100% |
| Project Work Log | 25% | 25% | 25% | 25% |
| References | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |
| Appendix | Salina Khadka | Safal Sharma | Kushal Upreti | Prawesh Gautam |

# 8. Project Conclusion

The project of developing Course Management System for Woodlands University College was a lot of work. The project took almost 22 weeks to complete. We began with interview session which helped us n gathering the data required for the project. We have tried our best to cover all the deliverables for this project. As per the requirement we focused more on the features and functionalities of the CMS rather than ease of implementation.

There are basically three users for whom we needed to design and develop the system. The three users are **Administrators, Students, and Module Leaders.** For each of these users we needed to make separate functionalities. The functionalities includes CRUD some additional operations. Before beginning the development work we figured out the use cases and we designed the prototype for the system. We created wireframes and mockup designs for every modules required. Then we decided to go for the development/implementation part of the project. We mainly used react and node js for the project. We created rest APIs for every modules and the testing of the rest APIs were done with the help of postman tool.

The course management system consists of several other management systems for different users. Each of the user needs to login with their credentials to access the modules. We have the following management system for the students.

* + Module Management
  + Timetable
  + Assignment Management
  + Diary Management
  + Attendance Records
  + Personal Tutor Management

We have the following management system for the administrator

* + Course Management
  + Module Management
  + Assignment Management
  + Diary Management
  + Attendance Records
  + Personal Tutor Management
  + Time Table Management
  + Student Records

We have the following management system for the module leaders

* + Module content management
  + PAT management
  + Assignment Management
  + Grade Management

We have tried to complete all the requirements for the project. Overall, it was a great learning experience. We got to experience working in a group and moreover working in a real time project.

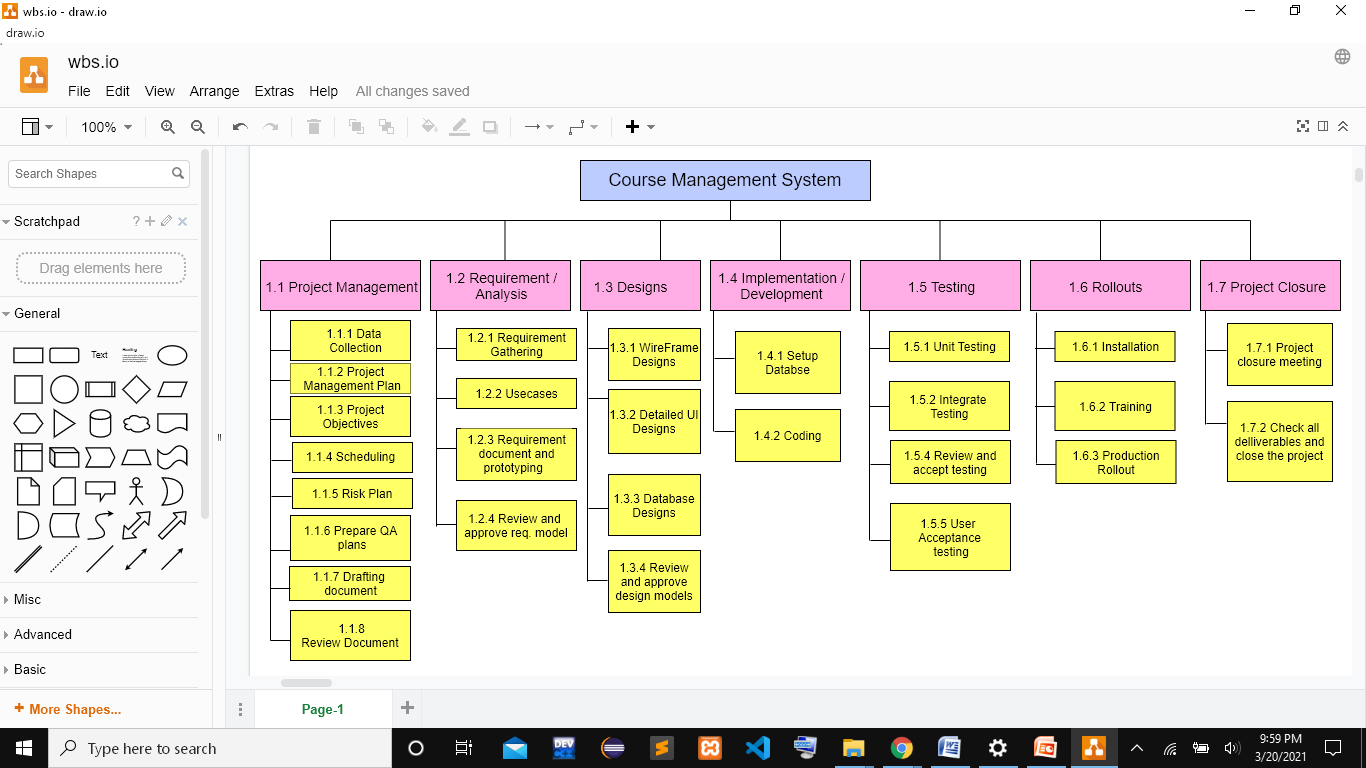
# 9. Project Management

In order for a project to be successful, project management is a must. Project management comprises of planning and leading projects. This section consists of some diagrams that were important for project management which includes Work breakdown structure, PERT chart, activity table and Gantt chart.

## 9.1. Work Breakdown Structure (WBS)

Work Breakdown Structure is a chart that illustrates tasks or activities for a project. It breaks down tasks into different categories and further into sub tasks until we end up at the lowest level of hierarchy with individual tasks which be recognized as a single coherent task.

The WBS should be **Mutually Exclusive Completely Exhaustive** which means each task should be different from another task and it should include all the tasks.



*Fig: Workbreak down structure of project (CMS)*

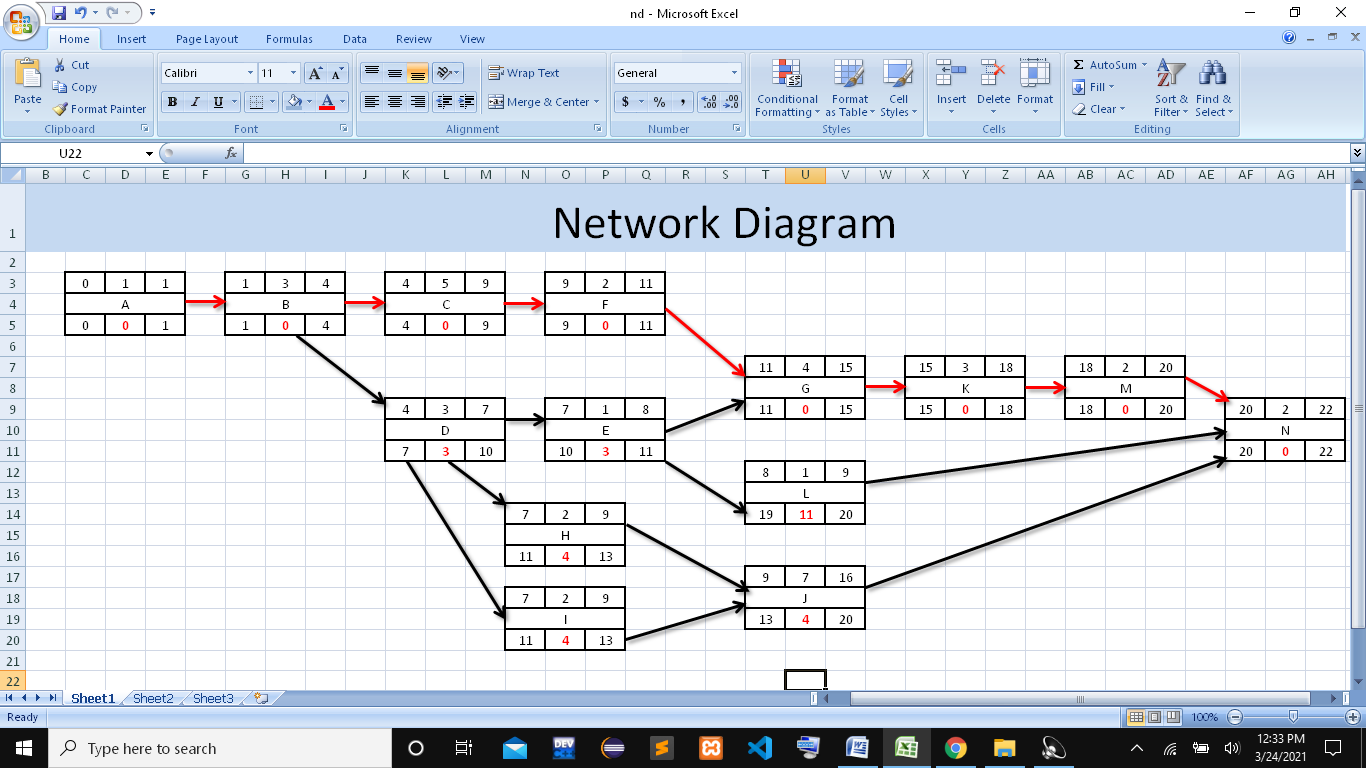
## 9.2. PERT (Program Evaluation and Review Technique) chart

PERT chart is a statistical tool used to project management in order to provide graphical representation of project’s timeline.

The table below is dependency table which illustrates which tasks are dependent on each other. It consists of the milestones that we need to complete during certain time period.

|  |  |  |  |
| --- | --- | --- | --- |
| Activity | Description | Preceding Activity | Duration(Weeks) |
| A | Data collection | - | 1 |
| B | Scheduling | A | 3 |
| C | Preparing QA plans | B | 5 |
| D | Reviewing documents | B | 3 |
| E | Requirement gathering | D | 1 |
| F | Finalizing use cases | C | 2 |
| G | Prototyping | E,F | 4 |
| H | Designing | D | 2 |
| I | Setting up database | D | 2 |
| J | Coding | H,I | 7 |
| K | Testing | G | 3 |
| L | Installation | E | 1 |
| M | Training | K | 2 |
| N | Production rollout | L,J,M | 2 |

*Fig: Table illustrating task dependency*

**

*Fig: Network diagram to calculate critical path and slack*

In the above network path, A-B-C-F-G-K-M-N is the **critical path** meaning those activities cannot be delayed even by a single day. If those activities are delayed even by one day then the whole project will be delayed.

The duration in red is the **slack** for each activity. Slack is the time period that can be delayed to complete an activity. In our case it is the number of weeks that can delay to complete the activity but it will not delay the ending of project.

## 9.3. Gantt Chart

Gant chart is a horizontal bar which provides graphical representation of project schedule that helps to plan, coordinate and track specific task in the project.

We have created the below Gantt chart according to the above network diagram.



*Fig : Gantt Chart*

## 9.4. Project Meetings

The following gives details on every meeting that we held for project discussion.

**Meeting 1:**

**Venue:**Google meet

**Time:**7:00 am

**Date:**11thDecember , 2020

**Person in Attendance:**Team members and Mr.Suresh Gautam

**Work Completed:**None

**Meeting agenda:** Assignment Briefing

**Activities done:** Teams divided

**Plannings:** Documentation work division

**Meeting 2:**

**Venue:** Google meet

**Time:** 10:00 am

**Date:** 20thDecember, 2020

**Person in Attendance:** Team members

**Work Completed:** Introduction part of documentation

**Meeting agenda:** Update in work completion

**Activities done:** Made changes in documentation

**Plannings:** Development work scheduling

**Meeting 3:**

**Venue:** Google meet

**Time:** 2:00 pm

**Date:** 30thDecember, 2020

**Person in Attendance:** Team members

**Work Completed:**

* Section 1 of documentation
* Mockups and wireframes for login page

**Meeting agenda:** Update on work

**Activities done:**

* Made changes in mockup designs according to the briefing

**Plannings:** Interview Planning and scheduling

**Meeting 4:**

**Venue:** Google meet

**Time:** 12:00 pm

**Date:** 5th January, 2021

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** Section 1 and 2 of documentation

**Meeting agenda:** Interview session with Mr. Suresh Gautam

**Activities done:** Interview session done

**Plannings:** Further working on development and coding part

**Meeting 5:**

**Venue:** viaviber

**Time:**9:00 pm

**Date:** 16thJanuary, 2021

**Person in Attendance:** Team members

**Work Completed:** Section 1 and 2 of documentation and login page frontend completed

**Meeting agenda:** Checking on work progress

**Activities done:** Made some changes in documentation part

**Plannings:** Working on development

**Meeting 6:**

**Venue:** Google meet

**Time:** 2:00 pm

**Date:** 30thJanuary, 2021

**Person in Attendance:** Team members

**Work Completed:** Section 1,2,3,4 of documentation completed, development of student and admin login page, database design completed

**Meeting agenda:** Checking on work progress

**Activities done:** Made some changes in database design

**Plannings:** Working on development part

**Meeting 7:**

**Venue:** Nami college

**Time:** 8:00 am

**Date:** 3rdFeburary, 2021

**Person in Attendance:** Team members

**Work Completed:** Documentation up to 4 completed, development of module leader app

**Meeting agenda:** Discussing the progress

**Activities done:** Worked on few modules of student app

**Plannings:** Working on documentation part

**Meeting 8:**

**Venue:** Nami college

**Time:** 8:00 am

**Date:** 10thFeburary, 2020

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** Documentation part up to 4 completed

**Meeting agenda:** Discussing on Rest APIs

**Activities done:** Researched about Rest APIs

**Plannings:** Create Rest APIs for all modules

**Meeting 9:**

**Venue:** Nami College

**Time:** 8:00 am

**Date:** 18thFeburary, 2021

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** Documentation up to 4 and development of some modules of student app, Rest APIs made for all modules

**Meeting agenda:** Discuss about agile methodology of software development

**Activities done:** Discussed about scrum and agile methodology

**Plannings:** Working on development as well as documentation

**Meeting 10:**

**Venue:**Nami college

**Time:** 10:00 am

**Date:** 3rdMarch, 2021

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** Development work of student app almost completed, gantt chart made

**Meeting agenda:** Discuss on unit testing, connection pooling

**Activities done:** Demonstrated our work to Mr. Suresh Gautam

**Plannings:** Work on unit testing and connection pooling

**Meeting 11:**

**Venue:**Nami college

**Time:** 12:00 pm

**Date:** 10thMarch, 2021

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** Section 1, 2,3,4,9completed, development work of student app completed

**Meeting agenda:** Discussion about creating a githubrepo

**Activities done:** Created a github repo, demonstrated work to Mr. Suresh Gautam and receiving feedbacks

**Plannings:** Working in the repository in github

**Meeting 12:**

**Venue:** Nami College

**Time:** 1:00 pm

**Date:** 17thMarch, 2021

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** Documentation completed upto 4 and section 9 and 7 also completed, development of student app and few modules of admin completed

**Meeting agenda:** Discussing about how to write project methodologies and DFD (Data Flow Diagram) in documentation

**Activities done:** Researched on DFD

**Plannings:** Adding more information in the documentation

**Meeting 13:**

**Venue:** Nami College

**Time:** 12:00 pm

**Date:** 26thMarch, 2021

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** Documentation 1,2,3,4,7,8,9 completed, all modules of admin completed

**Meeting agenda:** Discussing on test cases

**Activities done:** Made some test cases

**Plannings:** Working on test cases

**Meeting 14:**

**Venue:** Nami College

**Time:** 12:00 pm

**Date:** 30thMarch, 2021

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** Documentation 1,2,3,4,7,8,9 completed, all modules of admin completed, test cases of few modules done

**Meeting agenda:** Discussing on test cases

**Activities done:** Made some more test cases

**Plannings:** Working on test cases and development part

**Meeting 14:**

**Venue:** Nami College

**Time:** 12:00 pm

**Date:** 10thApril, 2021

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** Documentation 1,2,3,4,7,8,9 completed, all modules of admin completed, few modules of module leader app completed, and test cases work on progress

**Meeting agenda:** Discuss on error handling

**Activities done:** Handled some error in the system

**Plannings:** Working on error handling

**Meeting 15:**

**Venue:** Nami College

**Time:** 12:00 pm

**Date:** 20thApril, 2021

**Person in Attendance:** Team members and Mr.Suresh Gautam

**Work Completed:** All documentation and development work completed

**Meeting agenda:** Checking on project

**Activities done:** Demonstrated the whole project to Mr. Suresh Gautam

**Plannings:** Making demo video

## 9.5. Project Quality Plan

Project Quality Plan includes details on the deliverables of the project. It specifies the standard of the project. Clients will have some expectations regarding the project and it is the responsibility of ours to meet their expectations by providing them a quality product. The table below provides a transparent overview on the deliverables of our product.

|  |  |  |  |
| --- | --- | --- | --- |
| **S N** | **Functionality** | **Importance Level** | **Completion Status** |
| Student | | | |
| 1 | Submitting assignments |  |  |
| 2 | Displaying grades |  |  |
| 3 | Sending email to PAT |  |  |
| 4 | Maintaining a diary |  |  |
| 5 | Viewing modules and timetables |  |  |
| Administrator | | | |
| 6 | Admission process |  |  |
| 7 | Assigning PAT to students |  |  |
| 8 | Adding admin, module leaders to the system |  |  |
| 9 | Assigning course to the students |  |  |
| 10 | Assigning module to module leader |  |  |
| 11 | Adding announcements |  |  |
| 12 | Changing the student’s status |  |  |
| 13 | Adding new modules and courses |  |  |
| 14 | Editing, deleting any added students or courses or module leaders |  |  |
| 15 | Adding personal notes |  |  |
| Module Leaders | | | |
| 16 | Adding attendances | High | Completed |
| 17 | Adding module contents |  |  |
| 18 | Adding assignments |  |  |
| 19 | Add grades |  |  |

# 10. References

1. Legislation.gov.uk.(2021). *Data Protection Act 2018*. [online] Available from: <https://www.legislation.gov.uk/ukpga/2018/12/contents/enacted>[Accessed 13 March 2021].

2. Legislation.gov.uk.(2021). *Computer Misuse Act 1990*. [online] Available from: <https://www.legislation.gov.uk/ukpga/1990/18/contents>[Accessed 13 March 2021].

3. Figma.(2021). *Figma: the collaborative interface design tool.* [online] Available at: <https://www.figma.com/>[Accessed 10 March 2021].

4. Legislation.gov.uk.(2021). *Copyright, Designs and Patents Act 1988*. [online] Available from: <https://www.legislation.gov.uk/ukpga/1988/48/part/I/chapter/III/crossheading/computer-programs-lawful-users>[Accessed 13 March 2021].

5. Engineer4Free.(2014).Project Management.*Youtube*[online] Available from: <https://www.youtube.com/playlist?list=PLOAuB8dR35oeyKU0ojIxD8Muf6Mwc8ugW> [Accessed 24 March 2021].

6. Student Project Guidance &Development.(2021). *System Design Of Library Management System – Student Project Guidance & Development*. [online] Available from: <https://www.studentprojectguide.com/project-report/system-design/system-design-library-management-system/> [Accessed 24 March 2021].

7. Freeprojectz.com.(2021). *College Management System Activity UML Diagram | FreeProjectz*. [online] Available from: <https://www.freeprojectz.com/uml-diagram/college-management-system-activity-diagram>[Accessed 24 March 2021].