

DIU Class Schedule System

BY

Md. Abdur Rahaman Zihad

ID: 171-15-9097

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Nusrat Jahan

Sr. Lecturer

Department of Computer Science and Engineering
Daffodil International University

Co-Supervised By

Subhenur Latif

Assistant Professor

Department of Computer Science and Engineering
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

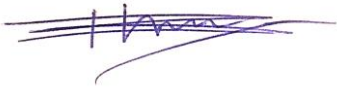
APRIL 2021

APPROVAL

This Project/internship titled “**DIU Class Schedule System**”, submitted by Md. Abdur Rahaman Zihad, ID No: 171-15-9097 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on Wednesday, June 02, 2021.

BOARD OF EXAMINERS

Chairman



Dr. Touhid Bhuiyan

Professor and Head

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University



Internal Examiner

Subhenur Latif

Assistant Professor

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University



Internal Examiner

Md. Abbas Ali Khan

Senior Lecturer

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University



Shah Md. Imran

Industry Promotion Expert

LICT Project, ICT Division, Bangladesh

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Nusrat Jahan, Sr. Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Nusrat Jahan
Sr. Lecturer
Department of CSE
Daffodil International University

Co-Supervised by:



Subhenur Latif
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:



Md. Abdur Rahaman Zihad
ID: 171-15-9097
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project successfully.

We really grateful and wish our profound our indebtedness to **Nusrat Jahan**, Sr. Lecturer, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*web development*” to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Professor **Dr. Touhid Bhuiyan**, Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

DIU Class Schedule as a web application and it's primarily showing class routine individually according to personal class schedule. It is also can be used for check available empty rooms, booking rooms for extra class etc. There are basically three types of users. This application is mainly focused for students and teacher's needs. Students can see their class schedules according to their taken courses. Teachers can see their class schedules and they can use some other useful features. Teachers can book available empty rooms to take extra classes. They can cancel their bookings too by following some conditions. Admins are the most powerful users. Admin can upload new routine using this application. They can book rooms for teachers and cancel if needed. They can add new teacher profiles if it is not in the database. This web application is secured enough that no users can access any functionality that they have no permission to access. Users can get their class schedule and do everything in just two or three clicks. I believe this application will ease the routine management and class booking hassle of students and teachers.

TABLE OF CONTENTS

Contents	Page
Board of examiners	ii
Declaration	iii
Acknowledgements	iv
Abstract	v
CHAPTER	
CHAPTER 1: Introduction	1-2
1.1 Introduction	1
1.2 Motivation	1
1.3 Objectives	1
1.4 Expected Outcomes	2
1.5 Project Management and Finance	2
1.6 Report Layout	2
CHAPTER 2: Background	3-4
2.1 Preliminaries	3
2.2 Related Works	3
2.3 Comparative Analysis	3
2.4 Scope of the Problem	4
2.5 Challenges	4
CHAPTER 3: Requirement Specification	5-10
3.1 Business Process Modeling	5
3.2 Requirement Collection and Analysis	5
3.3 Use Case Modeling and Description	8
3.4 Logical Data Model	9

3.5 Design Requirement	10
CHAPTER 4: Design Specification	11-13
4.1 Front-end Design	11
4.2 Back-end Design	11
4.3 Interaction Design and User Experience (UX)	13
4.4 Implementation Requirements	13
CHAPTER 5: Design Specification	14-25
5.1 Implementation of Database	14
5.2 Implementation of Front-end Design	14
5.3 Testing Implementation	21
5.4 Test Results and Reports	25
CHAPTER 6: Impact on Society, Environment and Sustainability	26-27
6.1 Impact on Society	26
6.2 Impact on Environment	26
6.3 Ethical Aspects	26
6.4 Sustainability Plan	27
Chapter 7: Conclusion and Future Scope	28-29
7.1 Discussion and Conclusion	28
7.2 Scope for Further Developments	28
Reference	30
Appendices	44
Originality Report	45

LIST OF FIGURES

FIGURE	PAGE
Figure 3.1.1: Business process modeling	5
Figure 3.2.1: Software development life cycle (agile)	6
Figure 3.2.2: Flow Chart	7
Figure 3.2.3: Data Flow Diagram for system	7
Figure 3.2.4: E-R Diagram of the system	8
Figure 3.3.1: Use case diagram	9
Figure 3.4.1: Logical Data Model	9
Figure 3.5.1: Architecture Design	10
Figure 5.2.1: Login Page	15
Figure 5.2.2: Routine	15
Figure 5.2.3: Profile of student	16
Figure 5.2.4: Profile of teacher	16
Figure 5.2.5: Add new course	17
Figure 5.2.6: Search empty rooms	17
Figure 5.2.7: Search empty room to book	18
Figure 5.2.8: Book an empty room	18
Figure 5.2.9: List of booked room	19
Figure 5.2.10: Teachers list	19
Figure 5.2.11: Parse Routine	20
Figure 5.2.12: Routine CSV	20
Figure 5.2.13: Standard form of evening routine	21

LIST OF TABLES

TABLE	PAGE
Table 4.2.1.1: NodeJS Latest Updates	11
Table 4.2.2.1: Express Framework Releases	12
Table 4.2.2.2: NodeJS Releases	12
Table 4.2.2.3: MySQL Releases	12
Table 5.3.1: Different tests	22

CHAPTER 1

INTRODUCTION

1.1 Introduction

It have been few months when Daffodil International University digitalized routine and booking management. Before then it was fully manual analog system. Students and teacher had to download pdf file filled with the routine of full CSE department then search for their course schedule inside the pdf. It was a huge waste of time. Again if teacher wanted to book room for extra class then he/she had to discuss schedule with students and then go to the department office to register a room. Sometimes two or more teachers used to book the same room on the same day which causes overlap and other issues. Class Schedule DIU application solved these problem a lot but it raised another problem. As it was an android application, non-android users were unable to use it. In this case, DIU Class Schedule web application will come handy. As it is a web application, users can use it on any platform they have. Also Class Schedule DIU and DIU Class Schedule are synchronized to each other. So users won't have to worry about which app they are using. Also a new technology has been used in this web application which is called PWA (Progressive Web Application). Using this technology, users will able to install this web application in their device like a native application. They won't need to use browser or stay connected to internet to use this application. We believe this will ease the life of the students and teachers of our university.

1.2 Motivation

As there is no system for managing routine and room booking our main aim is to make a scalable and dynamic system for the Computer Science and Engineering department for both Main and Permanent campus so that students and teachers can easily view and manages their classes and admin can manage the routine system. As it is a web based app, so any student or teacher with internet connection will be able to use it regardless of whatever platform they are using. We integrated our app with the highly secured ERP & Student Portal login system so that our user is secured from any security vulnerability and they don't have to use a new login system.

1.3 Objectives

Our main objective is to make a system where students can view their routine with other features and the teachers can book rooms with ease. On the other hand, we will be building a controllable and secured admin panel inside which will be used to manage the routine of the whole department. Because of having our app on two platforms, we have used rest API which is accessible from any type of platform and gives a secure environment. Our project is capable of handling a lot of user in real-time and give a robust performance to the users. Also our project is platform independent, so everyone will be able to use it on any platform (Android, IOS, Windows, etc.).

1.4 Expected Outcome

The expected outcome of our project is straight and forward. It helps the student to view their routine and let them know if their class teacher has booked any rooms for an extra class for them. For teachers, it helps them to book at a specific time if necessary. It also overcomes the limitation of the similar existing project for class management.

1.5 Project Management and Finance

This project is planned to be made open source when we complete almost all the tasks. So, if any further feature is needed then students of our university can always contribute and add any new feature. As our app is dependent on a backend server financial expense is needed to bear the cost. Although the cost will be pretty low as the app is designed in an efficient way so that remote dependency is at its minimum. So, the finance of this app will be handled by Daffodil International University itself.

1.6 Report Layout

In this project, a full overview of our system and related work and terminologies are given gradually. We recently made a survey on this similar work and try to what is more scope to develop this existing project. In chapter 2 we described the challenges and problems we faced which made it difficult for us. Another chapter 3 we describe the three-stage of background. We also describe the requirement specification and try to disclose users' demands. In, chapter 4 and 5 we disclose how we solve the problem and what we use to implement the project. Finally, in chapter 6 we remark some concluding and suggestions for future works.

CHAPTER 2

BACKGROUND

2.1 Preliminaries

Our educational institute Daffodil International University publishes routine and manages room booking manually. Which is very inefficient and causes a lot of problems. Students had to search through a large list in a pdf file to find their routine. After finding the routine they had to write it somewhere and then if someone loses it, he/she had to scrap the routine again from the pdf and rewrite it. On the other hand, when a teacher books a room manually sometimes it causes overlap with other teachers and various other issues. We will be planning toward solving all those problems and make a study on other related works to get a broad vision on the whole problem and make sure we can provide the best out of our project.

2.2 Related Works

There is a lot of routine related project available on the internet. But the problem is that they do not fit to fix the issues for our university. If one has a functionality it lacks another feature. And there was only one project which was made specifically for our university students which match in some aspect with our project. But the problem is that it lacks so many functionalities. So, our goal is to eliminate those issues and create a new application for students and teachers to help them out.

2.3 Comparative Studies

The other related project which was made for the students of our university lack functionalities such as it only supports displaying the routine. Moreover, the routine is only available for Main Campus students who are on the day shift. Students of the permanent campus and the students of the main campus who are on the evening shift do not get any benefit from the app. Teachers can also only view their routines. There is no option for room booking and other features. So, our application will take care of all these issues that lack in other projects. Sometimes application does not support all users. That is why we have developed our system as a web application to make our app accessible to every user. So that everyone can use our application.

2.4 Scope of the problem

Our main aim is to create a better system and eliminate the routine and booking related problem of our university. Initially, the project is meant for Computer Science and Engineering department. But the database will be designed in such a way that it can be modified easily and make it available for other departments. Moreover, if any other institution wants it can be easily converted to their need.

2.5 Challenges

Our challenge for making the Class Schedule application is to provide a secure and user-friendly environment to the users. Creating an interactive user interface is also a challenge. Moreover, as we are making this application platform independent, we must provide a consistent backend so our application can work fluently on every platforms. As this project will be dynamic, we must provide a feature-rich admin panel so that admin does not need to interact with the backend server.

CHAPTER 3

REQUIREMENT SPECIFICATION

Requirements specifications state what needs to be done by a system [1]. The requirements specification states what needs to be done in order for the organization to fulfill their purpose.

3.1 Business Processing Modeling

The following figure 3.1.1 shows Business Process Model of our application.

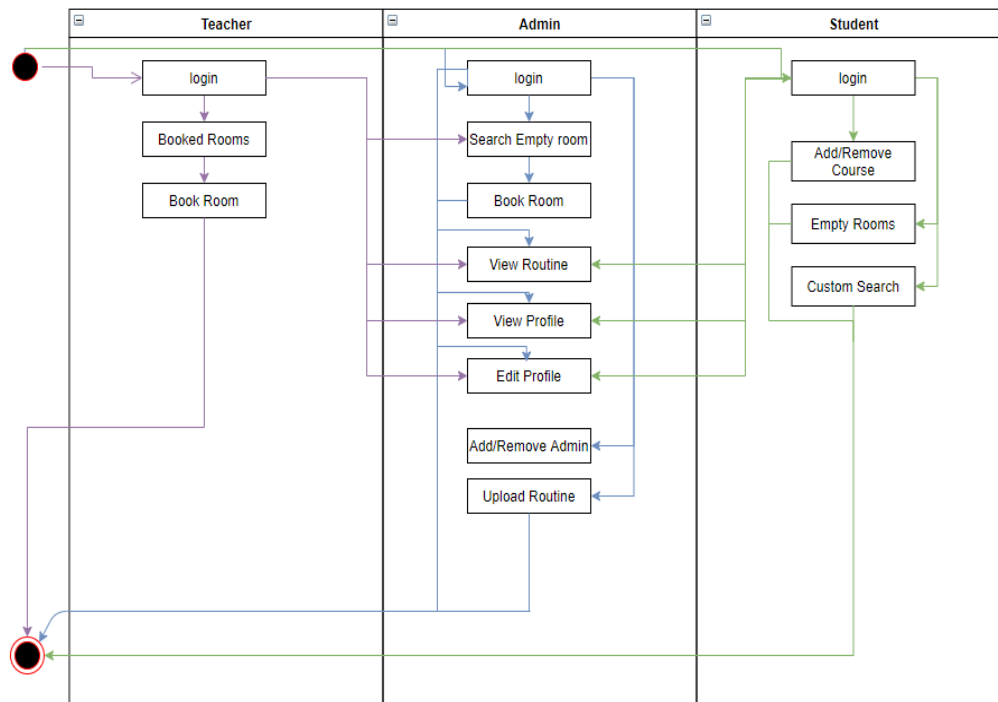


Figure 3.1.1: Business process modeling

3.2 Requirement Collection and Analysis

When it involves any kind of project, requirement collection plays a key role. Requirements collection is not only important for the project, but it's also important for the project management function. Requirement collection is the most vital step of a project. If the project team fails to capture all the required requirements for an answer, the project is running with a risk. This could result in many disputes and disagreements in the future. Therefore, take requirement collection as a key responsibility of the project team. So that we are collected our project requirement as soon as possible. Then we started our work.

Software Development Life Cycle (Agile)

The agile model is one of the most popular version of the systems development life cycle model for its linear sequential criteria, which suggests each phase must need to be totally completed before the following phase has start. At the tip of each phases, a review was taken to determine the project is on the correct path. Figure 3.2.1 illustrates the steps involved in agile model [2].

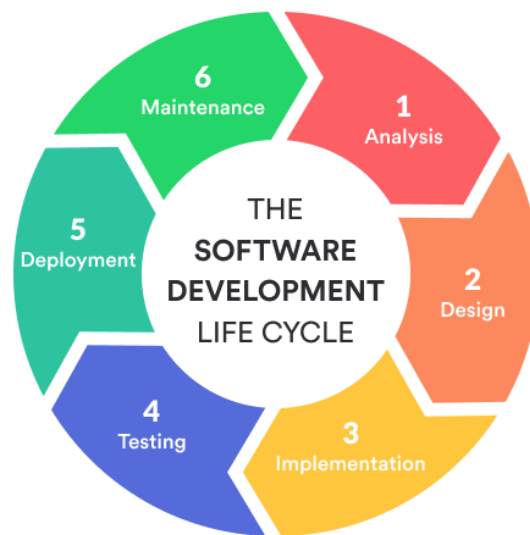


Figure 3.2.1: Software development life cycle (agile).

Flow Chart

The processes of our application Class Schedule is illustrated at figure 3.2.2.

Dataflow Diagram

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design). The below DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. Figure 3.2.3 illustrates Dataflow Diagram of our application.

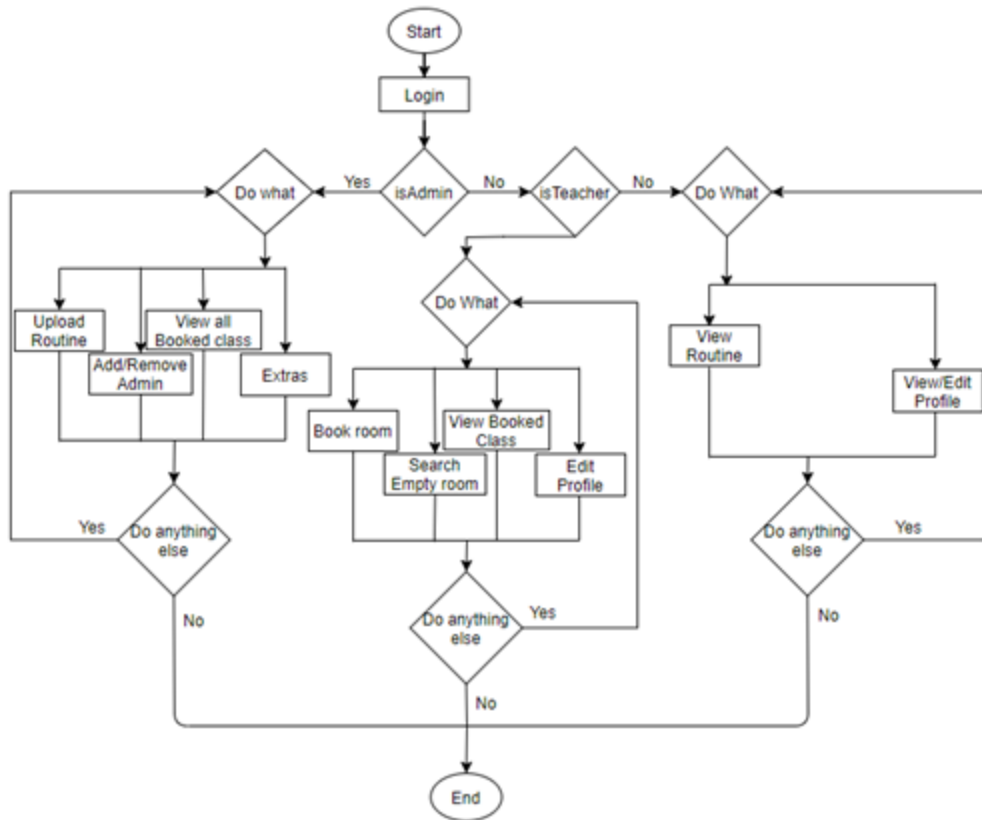


Figure 3.2.2: Flow Chart

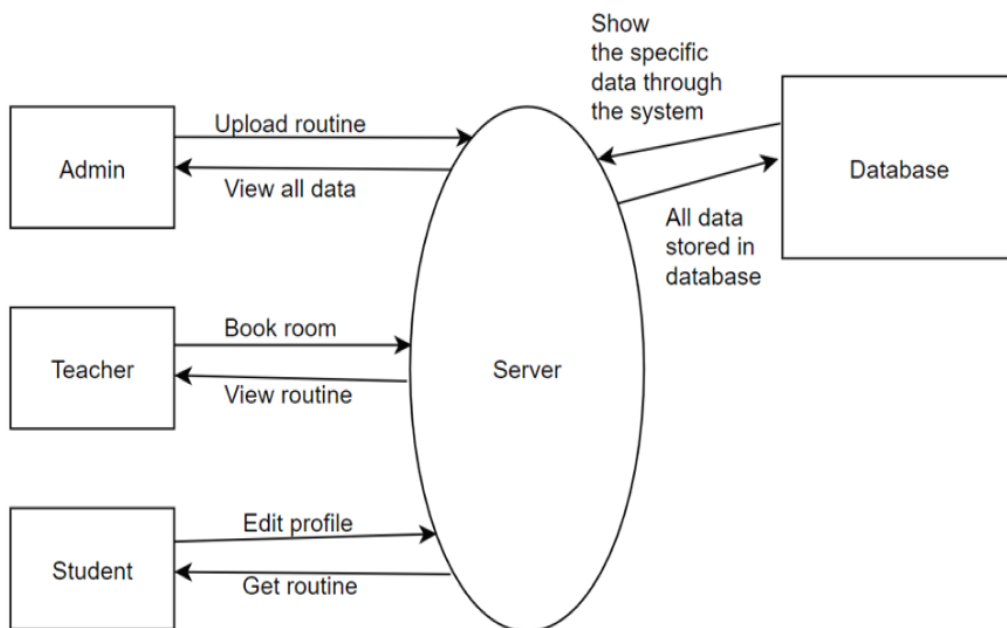


Figure 3.2.3: Data Flow Diagram for system

Entity Relationship Diagram

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database [3]. An ERD is conceptual and representational model of knowledge to represent the entity framework infrastructure. Figure 3.2.4 illustrates Entity Relationship Diagram of our application.

The elements of an ERD are:

- Entities
- Relationship
- Attribute

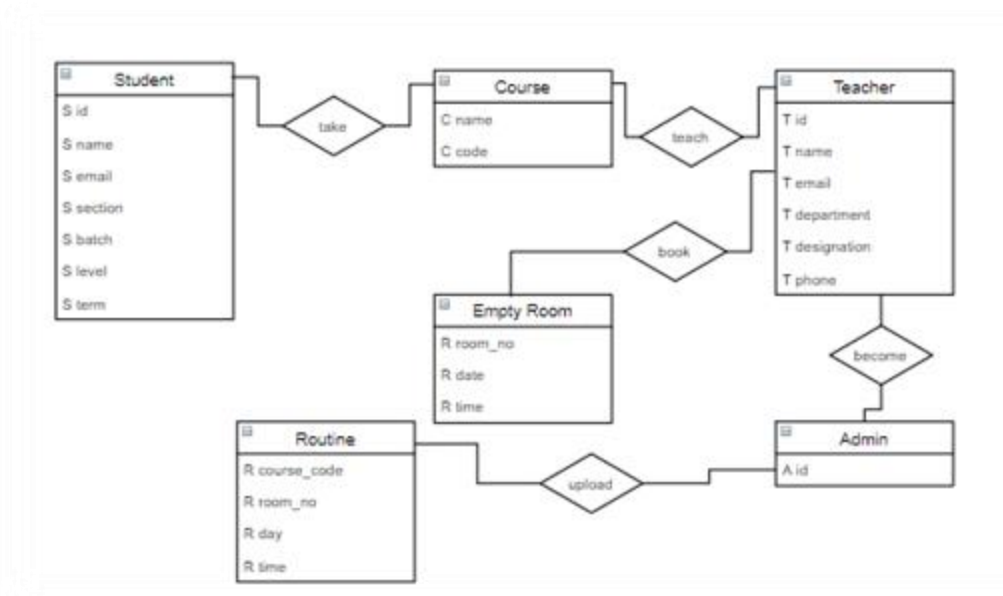


Figure 3.2.4: E-R Diagram of the system

3.3 Use Case Modeling

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved [4]. A use case diagram can identify the several types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. Use case diagram that identify and explain the features in our app is described in figure 3.3.1.

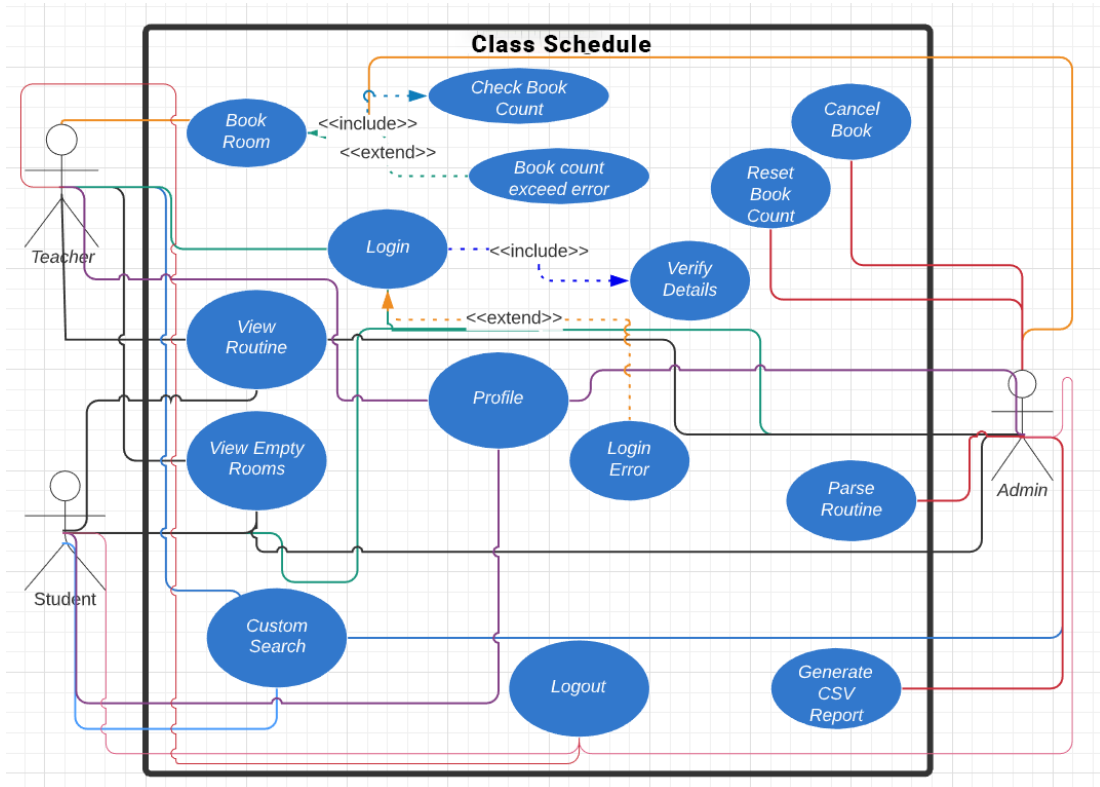


Figure 3.3.1: Use case diagram

3.4 Logical Data Model

Our application’s database is designed in a consistent way that querying within database will be smooth and it will give proper performance in the server side. A logical data models represent the abstract structure of a domain of information [5]. A logical data model for Class Schedule is depicted in Figure 3.4.1.

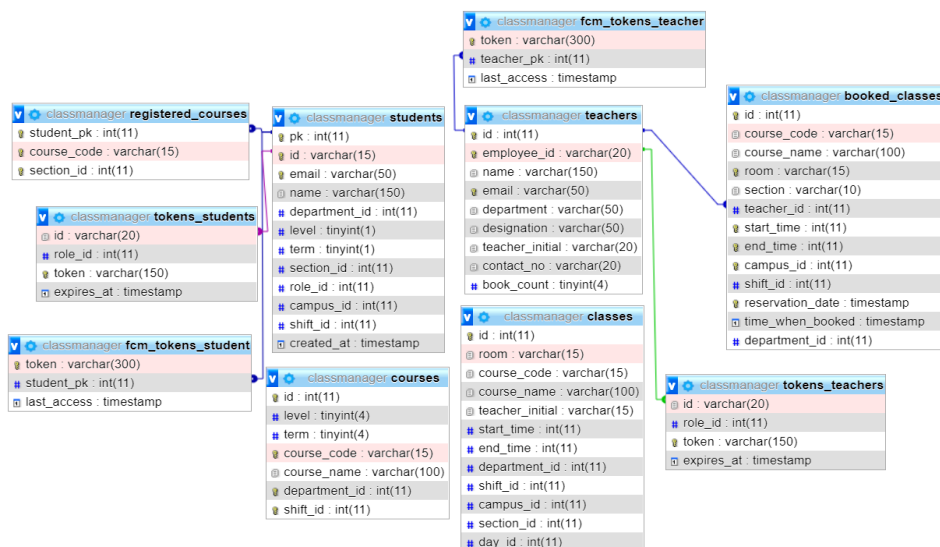


Figure 3.4.1: Logical Data Model

3.5 Design Requirement

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. During this chapter overall system design of our application has been shown, where architectural design, use case diagram, flow chart, and data multidimensional language included. The whole design of this project is user friendly. Modern and updated design tools are used for this project. Also, new concepts are considered to create them user-friendly. In the future, any quiet edit is allowed as time permits.

Architecture Design

Our app is built upon a simple but powerful architecture in which our application fetches data from an online web server via REST API and the show it in the user's device. The reason REST API is used because it is secure powerful and can be used in all platform such as Web, Android, iOS, Windows etc. A basic structure of our architecture design is figure 3.5.1 [6].

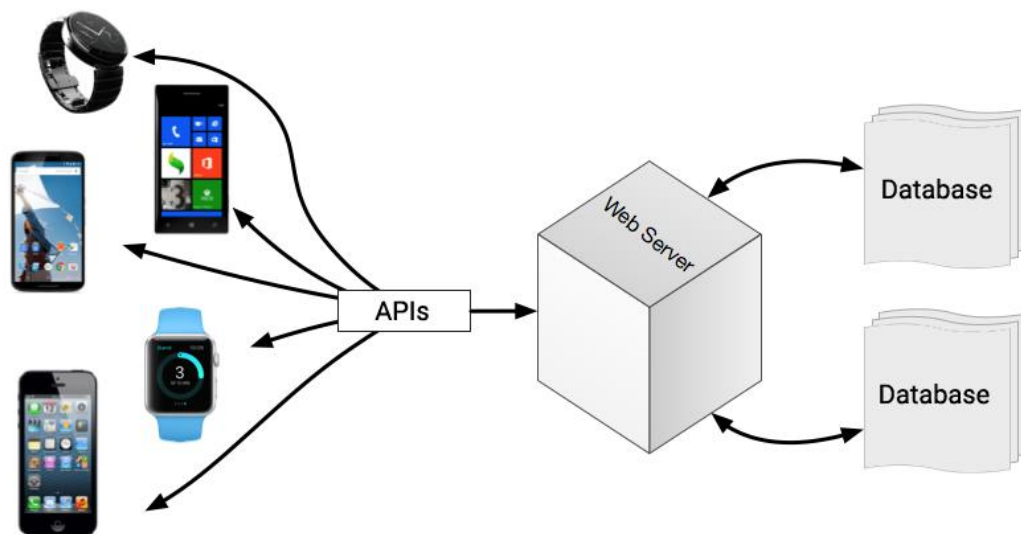


Figure 3.5.1: Architecture Design.

CHAPTER 4

DESIGN SPECIFICATION

4.1 Front-end Design

For designing frontend of a web application there are many options. HTML and CSS are must for designing a web application. In addition, we used a Javascript based frontend framework called VueJS. This framework is now very popular for building SPA (Single Page Application). We also used Vuetify library for designing our frontend.

4.2 Back-end Design

Back-end design can be classified into 2 categories for our application. First one is the application back-end and second one is server side back-end. We used Javascript for both backends. For server side, we used ExpressJS and for our application backend, we used NodeJS.

4.2.1 Application Back-end Design

There are quite a few options are available for designing web application back-end. As our focus is the performance for our application, we have used NodeJS to handle our backend. NodeJS is very fast, scalable and lightweight as a development platform.

Table 4.2.1.1: NodeJS Latest Updates

Version	Release Date
NodeJS 14.15 (Fermium)	23 February 2021
NodeJS 12.21 (Erbium)	23 February 2021
NodeJS 10.24 (Dubnium)	23 February 2021

4.2.2 Server Back-end Design

As our application will be for both Android and Web, we need a REST API for communicating with each other. Therefore, is a lot of option available for server backend. We have used Express framework for our REST API. Express is a web

framework for NodeJS. It has been called the de facto standard server framework for NodeJS. Node.js provides web application with high performance and high scalability. That means our application will provide high performance and we can scale it as our necessary. For database MySQL is being used. It is fast, secure and most used database management system in current time.

Table 4.2.2.1: Express Framework Releases

Version	Release Date
5.0.0-alpha.8	March 26, 2020
4.17.1	May 26, 2019
4.17.0	May 17, 2019
4.16.4	Oct 11, 2018
4.16.3	Mar 12, 2018

Table 4.2.2.2: NodeJS Releases

Version	Release Date
Node.js 14.9.0	2020-08-26
Node.js 14.8.0	2020-08-11
Node.js 14.7.0	2020-07-29
Node.js 14.6.0	2020-07-15
Node.js 14.5.0	2020-06-30

Table 4.2.2.3: MySQL Releases

Version	Release Date
8.0	19 April 2018s
5.7	21 October 2015

5.6	5 February 2013
5.5	3 December 2010
5.1	14 November 2008

4.3 Interaction Design and User Experience

Design of this application is user friendly. We also followed the material design guidelines provided by google to make our application more beautiful and interactive. We have also assured great user experience. Design of the views in our app is designed with the Vuetify library. Some elements were handled programmatically to ensure better user experience.

4.4 Implementation Requirements

To implement our design into life specific environment is required. As our application depends on the data from remote server a hosting service and database is required to communicate from our app to the server. Our backend REST API is built upon NodeJS framework so the server must support environment form running NodeJS applications.

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Implementation of Database

Our database is implemented in such a way that it is able to handle multiple concurrent requests simultaneously. We have used MySQL which is currently the most popular relational database management system. Moreover, it is open source it can be modified to our need if necessary. Administration of our database can be securely handled by phpMyAdmin which is available in mostly all platform. All our application data will be stored here including class information and user details.

5.2 Implementation of Front-end Design

UI of our application includes routine, user information, admin panel, search activity etc. are explained below:

Login: This is the first screen that appears when a user opens their app. It is connected to the login system of our university. That is why a user does not need to create an account to use this app. Teachers can login into this app using their employee id and password and on the other hand students can login using their student id and password of their student portal. So, they have to insert these information and then they will be redirected to complete their remaining information or if someone has already inserted them then they will be redirected to the home screen of the app. Figure 5.2.1 shows Login page.

Routine: After login, user will be redirected to the main screen of this app. Here users will able to view all their class schedules sorted in day by day basis. User will also able to turn on notification for a scheduled class. If notification is turned on then user will be notified via a push notification a few minutes before the specific class schedule. Figure 5.2.2 shows Routine page.

Profile : Every user have a profile area where they will able to view their profile information. Teacher and students will have different types of information in their profile area. Student will have Name, Email, ID, Department, Campus, Shift, Section, Level, Term, and Courses in their profile area while Teachers will have Name, Department, Email, Designation, and Teacher Initial. User's routines will be shown

accordingly their profile informations. Figure 5.2.3 shows profile of students and 5.2.4 show profile of a teacher.

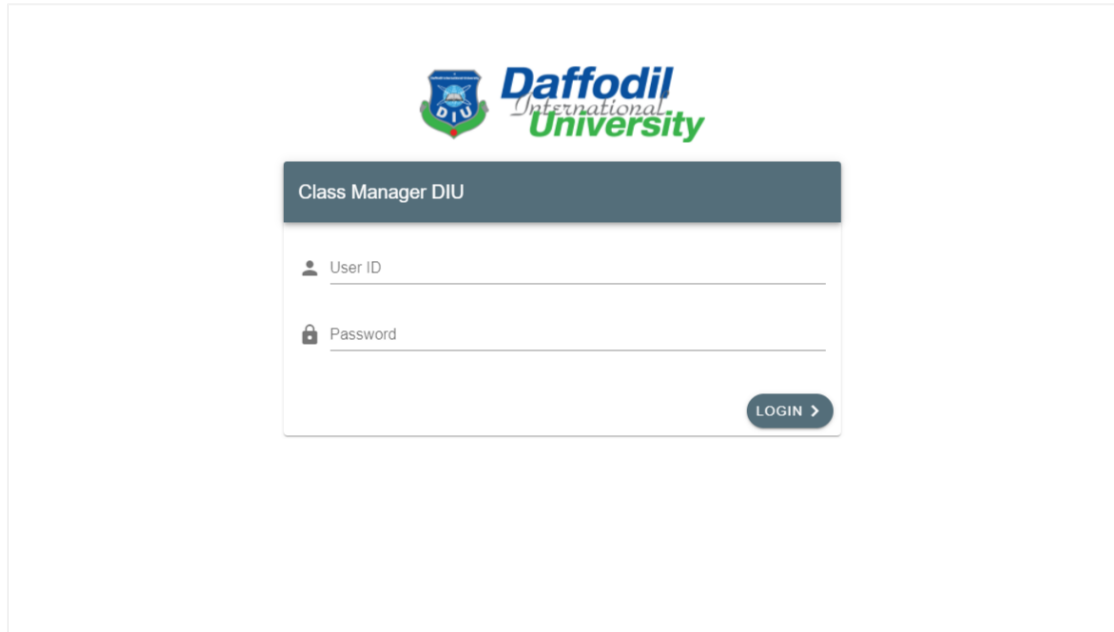


Figure 5.2.1: Login Page

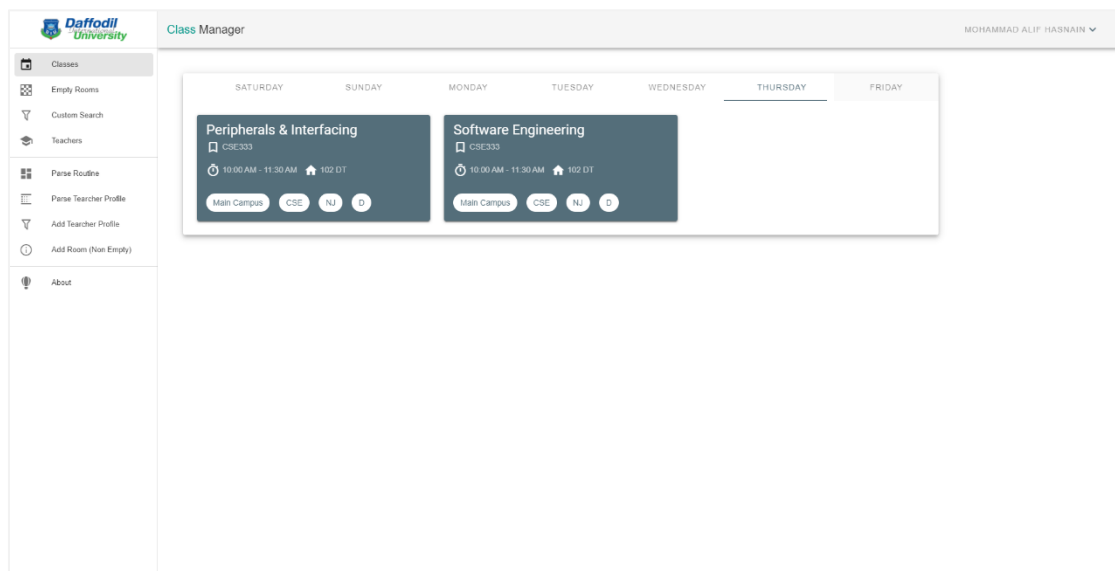


Figure 5.2.2: Routine

Add New Course (Student only): In profile page, students will be able to see their registered course list. There will be an option “Add Course” under the course list. Students will be able to add new course to their profile and track that course schedule. Figure 5.2.5 shows Add new page.

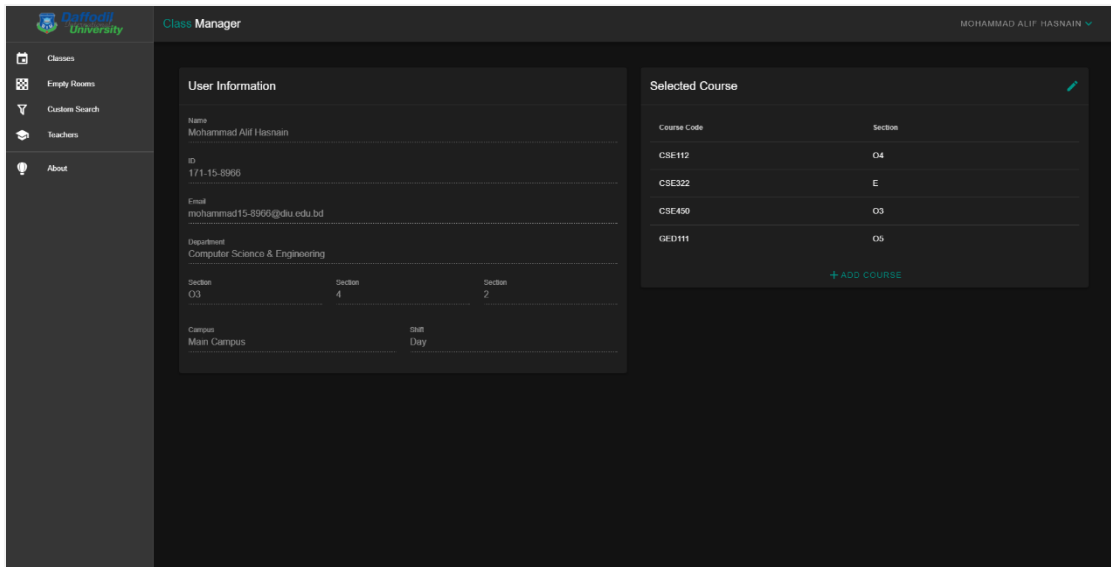


Figure 5.2.3: Profile of student

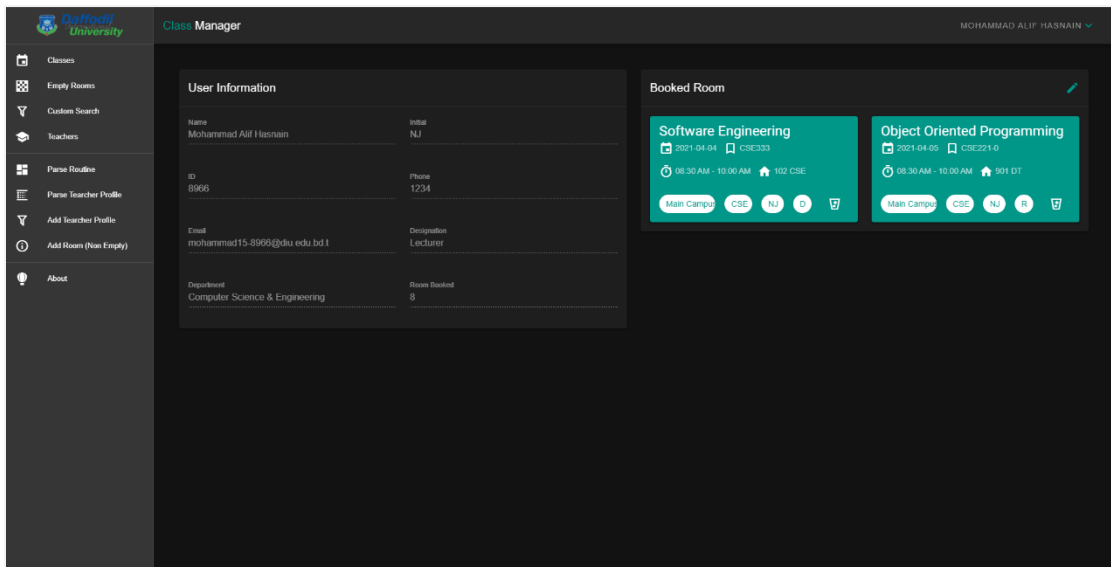


Figure 5.2.4: Profile of teacher

Search Empty Room : Selecting “Empty Rooms” form navigation drawer will enable users to see the empty rooms that are available in a specific time. Here users can search for all empty room for a specific date and time. User will have to select the day of the week, Time period, and campus. The list of all the rooms which will be empty on that selected time will be shown. Figure 5.2.8 shows UI for displaying empty rooms.

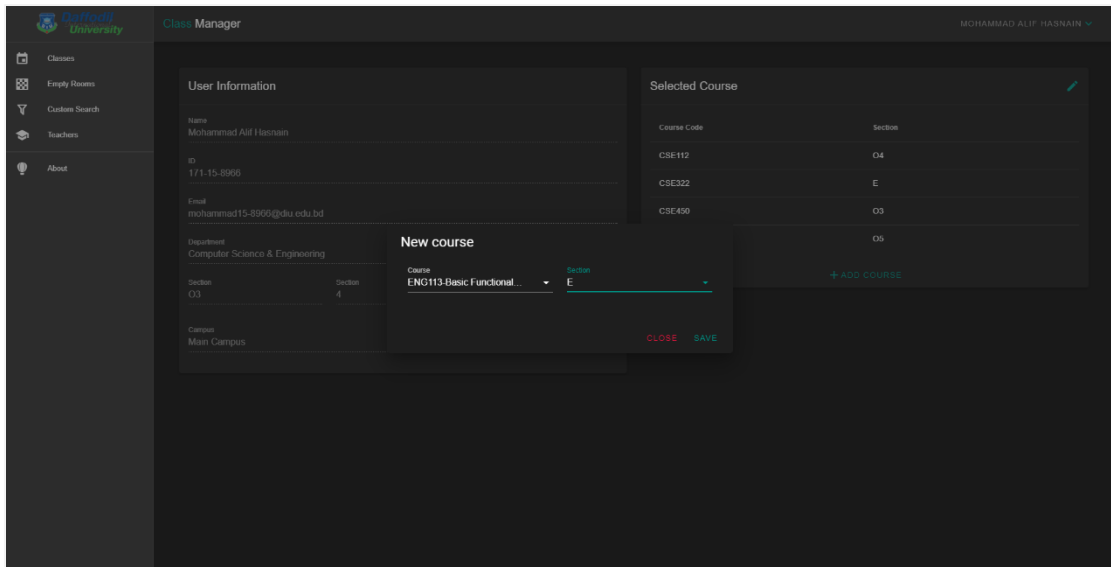


Figure 5.2.5: Add new course

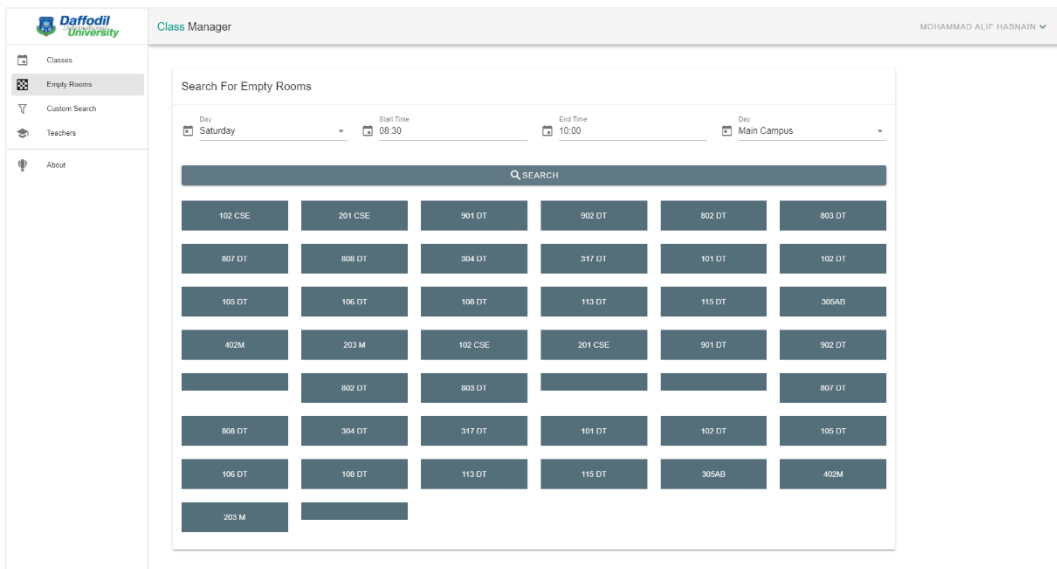


Figure 5.2.6: Search empty room

Book Room (Teachers Only) : Selecting “Empty Room” option from the navigation drawer teachers will go to empty room page where they will get an extra option “Book Room” to book a room in a specific day and time. A user will have to put the date, time period, and campus for searching rooms in a specific time. Then it will show all empty rooms on that schedule. User will have to click on the room number. This will show a new dialog where user will have to provide the course details, section, shift, and department name to book that room for the specific date and time. This feature is only available for teachers. Figure 5.2.7 and figure 5.2.8 shows UI for room booking by teachers.

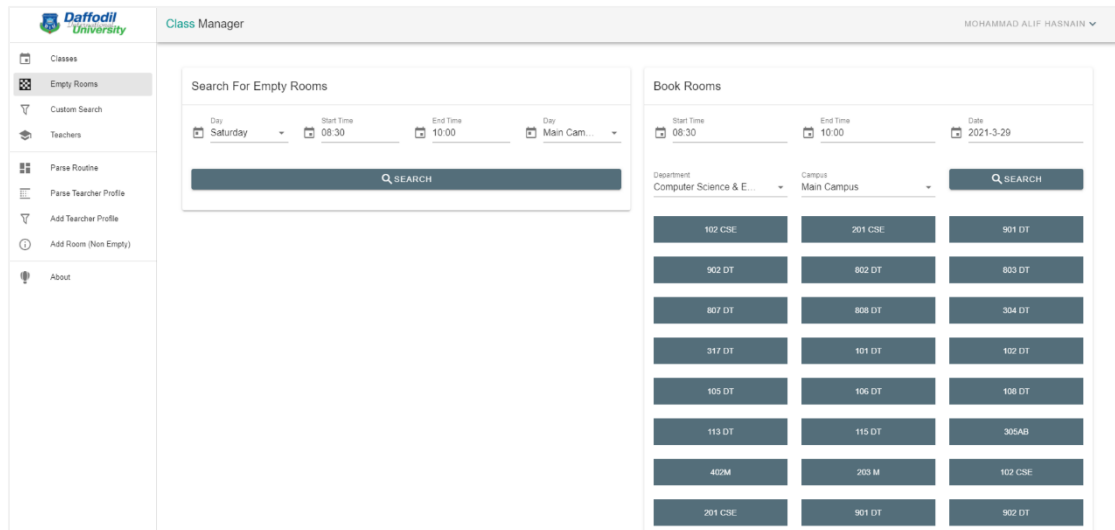


Figure 5.2.7: Search empty room to book

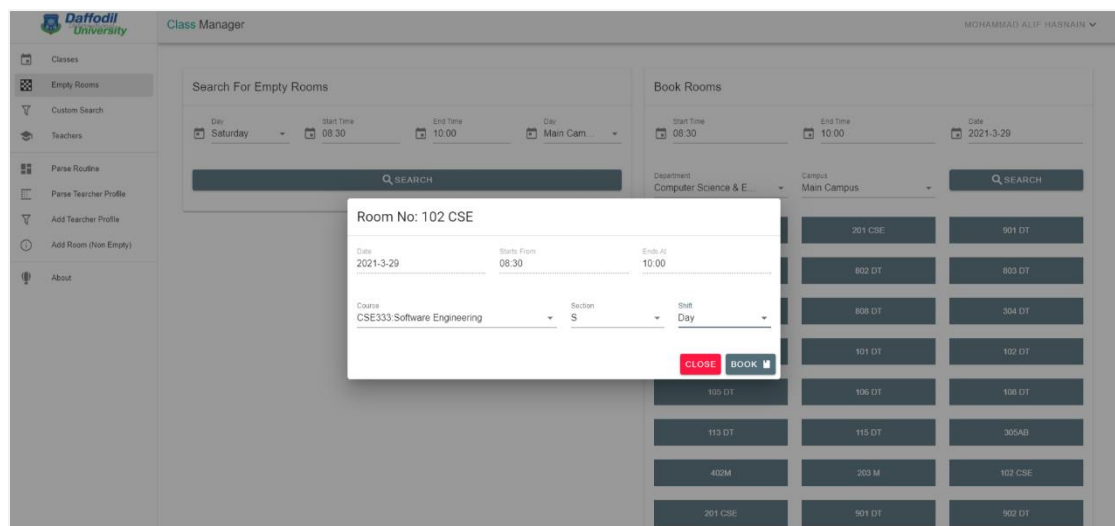


Figure 5.2.8: Book an empty room

List of booked room: Selecting “Profile” from the dropdown menu will show users profile. In teacher’s profile page there will be a section called booked room where a teacher will be able to see the list of remaining extra room that he/she has booked. Figure 5.2.9 shows the list of remaining booked room.

Upload Routine (Admin Only) : To upload a routine file admin have to select “Parse Routine” from navigation drawer. Here admin will upload new routine CSV file and select department, shift, and campus. Routine will automatically update for every users. After uploading routine admin must increase routine version selecting “Miscellaneous” from navigation drawer. Figure 5.2.15 shows UI for parsing a routine.

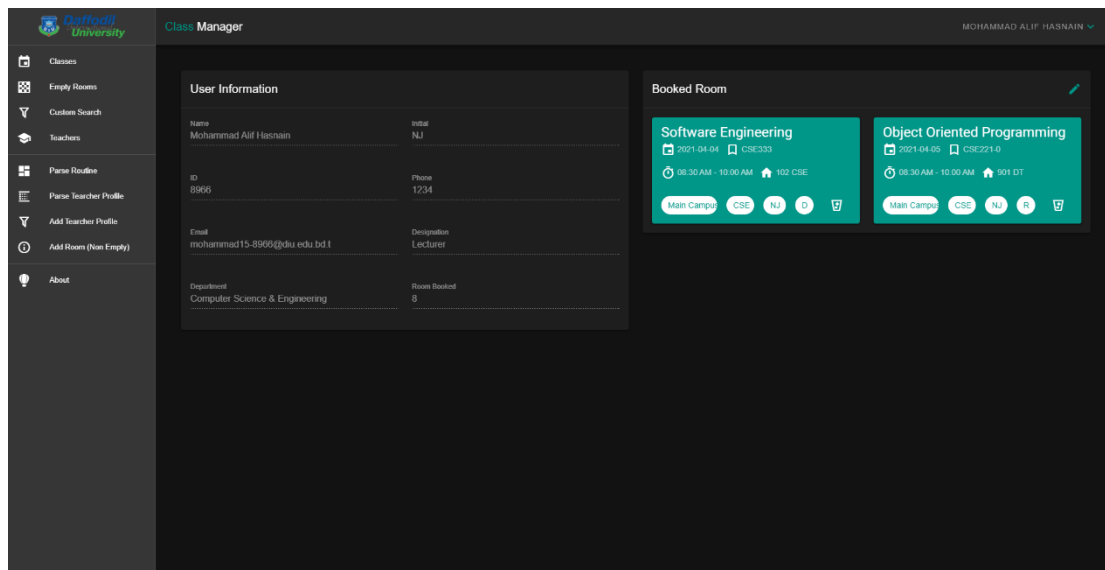


Figure 5.2.9: List of booked room

Teacher’s List : Every users can see every teacher’s details from “Teachers” page. Users will have to select “Teachers” from the navigation drawer. Users can also search for any teacher by their name or teacher initials. Figure 5.2.10 show the list of teachers.

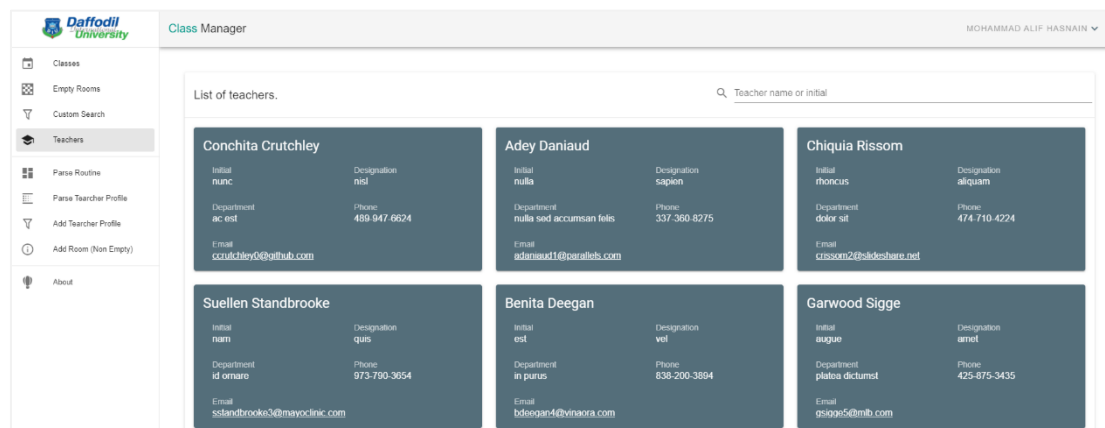


Figure 5.2.10: Teachers list

Upload Routine (Admin Only) : To upload a routine file admin have to select “Parse Routine” from navigation drawer. Here admin will upload new routine CSV file and select department, shift, and campus. Routine will automatically update for every users. After uploading routine admin must increase routine version selecting “Miscellaneous” from navigation drawer. Figure 5.2.11 shows UI for parsing a routine.

Routine Parsing Procedure : Admin must follow some steps to make the routine CSV file available for parsing. This formatting must be done carefully or it will result in routine with mistakes. Full procedure is given below –

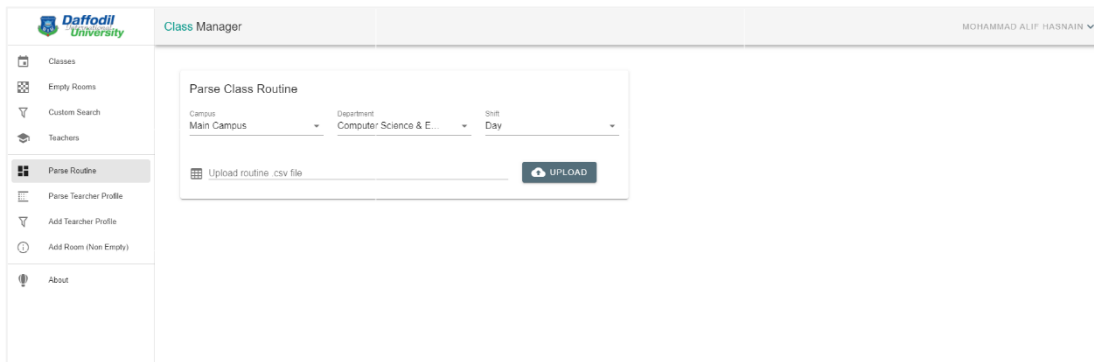


Figure 5.2.11: Parse Routine

- Day Shift CSE Main Campus** : Routine parser algorithm will parse routine day by day of the week. So that means it will first search for ‘Saturday’ keyword in the csv file then it will continue to parse before it reached the ‘Sunday’ keyword. The total number of columns will be from ‘A’ to ‘R’. Please note that there is no empty columns it the beginning. Figure 5.2.16 shows a demo of the csv file.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
Effective From: October 02, 2020													Prepared by: Routine Committee				
Schedule																	
08:30-10:00			10:00-11:30			11:30-01:00			01:00-02:30			02:30-04:00			04:00-05:30		
Room	Course	Teacher	Room	Course	Teacher	Room	Course	Teacher	Room	Course	Teacher	Room	Course	Teacher	Room	Course	Teacher
L-901 CSE	CSE334(O6)	NIS	L-901 CSE	CSE332(O6)	ANM	L-901 CSE	CSE332(O6)	ANM	L-901 CSE	CSE324(O4)	JAP	L-901 CSE	CSE324(O4)	JAP	L-901 CSE	CSE334(O6)	NIS
L-902 CSE			L-902 CSE			L-902 CSE	CSE312(O3)	FA	L-902 CSE	CSE312(O3)	FA	L-902 CSE	CSE334(O6)	NIS	L-902 CSE	CSE334(O6)	NIS
L-903 CSE	CSE332(O4)	UD	L-903 CSE	CSE332(O4)	UD	L-903 CSE	CSE155(O4)	MIR	L-903 CSE	CSE155(O4)	MIR	L-903 CSE	CSE332(O1)	ANMM	L-903 CSE	CSE332(O1)	ANMM
L-901 CSE	CSE135(O14)	SNP	L-901 CSE	CSE135(O14)	SNP	L-901 CSE	CSE332(O12)	TRA	L-901 CSE	CSE332(O12)	TRA	L-901 CSE	CSE215(R1)	FRS	L-901 CSE	CSE215(R1)	FRS
L-902 CSE	CSE135(O6)	ZZ	L-902 CSE	CSE135(O6)	ZZ	L-902 CSE	CSE135(O7)	MZR	L-902 CSE	CSE135(O7)	MZR	L-902 CSE			L-902 CSE		

Figure 5.2.12 : Routine CSV

Section concatenated with ‘+’ (plus sign) is allowed and can be parsed by the parser. But if teacher initial concatenated with ‘+’ (plus sign) is not allowed. Example is :

Banquet Hall	GED321(O1+O2+O3)	SMR+SH
--------------	------------------	--------

Here, sections with the course code is allowed but the initial “SMR+SH” is not allowed. So we have to modify this and write it like below :

Banquet Hall	GED321(O1+O2+O3)	SMR
Banquet Hall	GED321(O1+O2+O3)	SH

Electrical lab class schedule different than normal classes. So, it also needs some modification manually. As we can see the cell named “Lab (Electrical Circuit, Digital Electronics, Electronic Devices and Circuits)” has to be

renamed to “Electrical Lab” for all seven days of week. Then it has to be formatted like below:

A	B	C	D	E	F	G	H	I
Electrical Lab								
	09:00-11:00		11:00-01:00		01:00-03:00		03:00-05:00	
L-202 M L-506A M	CSE133(01)	SD			CSE224(05) <u>LJN</u>		CSE133(013)	SM

That is all information related to parsing routine of CSE day shift main campus routine.

- Evening Shift CSE Main Campus :** Unlike routine from CSE day shift main campus the routine of evening shift does not need a lot of modification. Please remember that parsing in this case does not support section and teacher initial concatenate with ‘+’ (plus sign). So, if there is any field like this then they must be separated. Section is taken from the row which includes course code. Note that there is no empty row at the beginning of the table. Parsing algorithm will read columns from ‘A’ to ‘K’. Figure 5.2.17 shows a standard form of evening routine.

A	B	C	D	E	F	G	H	I	J	K
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING										
CLASS ROUTINE FOR EVENING PROGRAM										
Fall 2020										
Effective From: September 09, 2020						Prepared By: Routine Committee				
06:00-07:30						7:30-9:00				
Day	Room #	Course Code	Course	Batch	Teacher Initial	Room #	Course Code	Course	Batch	Teacher Initial
Saturday	L-902 CSE					L-902 CSE				
	L-801 CSE					L-801 CSE				
	L-802 CSE					L-802 CSE				
	L-701 CSE	CSE421L(01)	Computer Graphics Lab	36(01)	MBS	L-701 CSE	CSE421L(01)	Computer Graphics Lab	36(01)	MBS
	L-702 CSE					L-702 CSE				
	L-601 CSE	CSE331L(01)	Compiler Design Lab	37(01)	MAH	L-601 CSE	CSE331L(01)	Compiler Design Lab	37(01)	MAH
	L-602 CSE	CSE313L(01)	Computer Network Lab	38(01)	TBA	L-602 CSE	CSE313L(01)	Computer Network Lab	38(01)	TBA
	L-101CSE					L-101CSE				
	903 CSE					903 CSE				
	801 DT5					801 DT5				
	802 DT5					802 DT5				
	803 DT5					803 DT5				
	804 DT5					804 DT5				
	805 DT5					805 DT5				
	806 DT5					806 DT5				
	807 DT5					807 DT5				

Figure 5.2.17: Standard form of evening routine

5.3 Testing Implementation

Testing is a method to test whether the product matches expected requirements and to confirm that product is Defect free. Several types of tests were added to check if everything is working as expected. Different test cases and their outcome is given in Table 5.3.1.

Table 5.3.1: Different tests

Test Case	Test Input	Expected Outcome	Obtained Outcome	Pass/Fail
1. Student Login	Student ID and Password	Successfully login	Successfully login	Pass
2. Student Login	Incorrect input	Login Failed	Login Failed	Pass
3. Teacher Login	Teacher ID and Password	Successfully login	Successfully login	Pass
4. Teacher Login	Incorrect input	Login Failed	Login Failed	Pass
5. View Profile	Click on Profile Menu	View Profile information	View Profile information	Pass
6. Update Profile	Give all the values	Update successful	Update successful	Pass
7. Search Empty Room	Give necessary value	Show search result	Show search result	Pass
8. Book Room	Give necessary values	Room booking successfull	Room booking successfull	Pass
9. Upload Routine	Select routine file and necessary value	Upload successfull	Upload successfull	Pass
10. Update Routine	Edit values and click save	Edit successful	Edit successfull	Pass
11. Logout	Click on "Logout"	Logged out	Logged out	Pass

Following are the test cases for the Students of DIU Class Schedule app.

- **Test Case 1.1:**

Precondition: Device must be connected to the internet.

Assumption: User has opened the application and is on login page.

Input: User pressed “Login” button.

Result: User is logged in to the system.

- **Test Case 1.2:**

Precondition: Device must be connected to the internet.

Assumption: User is logged in and viewing the “necessary info” page.

Input: User’s necessary info is entered and save button pressed.

Result: User is viewing the routine which was taken by him.

- **Test Case 1.3:**

Precondition: Device must be connected to the internet.

Assumption: User pressed “Empty Room” button to look for rooms that are available.

Input: Device must be connected to the internet.

Result: User is viewing the empty rooms that are available.

- **Test Case 1.7:**

Precondition: Device must be connected to the internet.

Assumption: User wants to see his profile info.

Input: User pressed “Profile” button.

Result: User is viewing his info.

- **Test Case 1.8:**

Precondition: Device must be connected to the internet.

Assumption: User wants to add course and pressed “Add new course” button.

Input: User entered desired section and subject and pressed confirmed button.

Result: New course added.

- **Test Case 1.12:**

Precondition: Device must be connected to the internet.

Assumption: User wants to see the app info.

Input: User pressed “About” button.

Result: User is viewing the info about the app.

- **Test Case 1.13:**

Precondition: Device must be connected to the internet.

Assumption: User wants to log out.

Input: User pressed “log out” button.

Result: User is logged out.

Following are the test cases for the Teachers of DIU Class Schedule app.

- **Test Case 1.15:**

Precondition: Device must be connected to the internet.

Assumption: User has opened the application and on login page.

Input: User credentials are entered and “Login” button is pressed.

Result: User is logged in to the system and viewing the routine.

- **Test Case 1.16:**

Precondition: Device must be connected to the internet.

Assumption: User wants to see his profile info.

Input: User pressed “Profile” button.

Result: User is viewing his info.

- **Test Case 1.18:**

Precondition: Device must be connected to the internet.

Assumption: User pressed “Empty Room” button to look for rooms that are available.

Input: Desired day and time is entered and search button pressed.

Result: User is viewing the empty rooms that are available.

- **Test Case 1.19:**

Precondition: Device must be connected to the internet.

Assumption: User pressed “Empty Room” button to look for rooms that are available.

Input: Desired day and time is entered and search button pressed.

Result: User is viewing the empty rooms that are available.

- **Test Case 1.22:**

Precondition: Device must be connected to the internet.

Assumption: User wants to book a classroom and pressed “Book classroom” button.

Input: Date,time is entered and search button pressed.

Result: User is viewing available room.

- **Test Case 1.23:**

Precondition: Device must be connected to the internet.

Assumption: User wants to see the room booked by him.

Input: User pressed “Booked Classrooms” button.

Result: User is viewing the room booked by him.

- **Test Case 1.27:**

Precondition: Device must be connected to the internet.

Assumption: User wants to see the app info.

Input: User pressed “About” button.

Result: User is viewing the info about the app.

- **Test Case 1.28:**

Precondition: Device must be connected to the internet.

Assumption: User wants to log out.

Input: User pressed “log out” button.

Result: User is logged out.

5.4 Test Results and Reports

Test result was successful also there can be some minor bug that can't always be exploited using tests. A simple Unit test report summary is given at Figure 5.4.1.

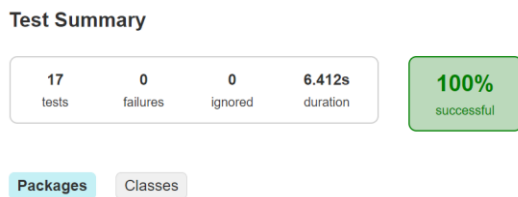


Figure 5.4.1 : Unit Test report

CHAPTER 6

IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

6.1 Impact On Society

From our childhood, we have seen that routine management was manual. Students had to carry a piece of paper or put the paper on the wall to see the routine. But our science and technology have improved so much that has made gadget like smartphone very cheap to afford. So almost everyone has access to smartphones. That is where our routine comes in handy. The app can be accessed from student's devices, so they will not have to carry the extra paper which decreases the usage of paper socially. Our society is more backward using technology compared to the people of the west. The Internet can be a way to reduce this problem. As our app requires the internet to log in and to update the routine, students will have to use the internet sometimes. As a result, they will have to learn to buy data packages, device configurations, etc. In other words, they will at ease solving the basic problems that people face sometimes.

6.2 Impact On Environment

Our application is totally environment friendly. Although it can have some impact on the environment as well. As an example to use our application an smart device (Smart Phones/Laptops) is necessary and a mobile phone can cause environmental pollution if the e-garbage from the phone is not recycled correctly. So, if someone brought a mobile phone to use our app and does not recycle it that this can cause an impact on the environment. Moreover, we are also helping the environment through our application. This is because when there was no application for routine and room booking management people must use paper and pen for noting them down. To make papers tree is necessary. So, our application will help to save thousands of trees someday and make this world beautiful.

6.3 Ethical Aspects

If we consider the ethical aspect there is nothing to worry about. Because we are only using it for personal work for an institution so there will not be any unethical work regarding our application. Moreover, the application is also built open source so there is less chance of doing unethical or something that could hurt someone in any way.

6.4 Sustainability Plan

Web technologies update regularly and brings new features and security measures. The framework and technology we used in our project gets major updates after certain periods. We have build our application using the latest availble technology at this time. Again if any major update comes which may conflicts against our existing code then we will have to update our application and make required changes. This applications will be open source after a certain time and other web technology enthusiasts will able to make changes and add new features to our application.

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 Discussion and Conclusion

There was no application for room booking and viewing routine in our university once. Students and teachers must do these works manually. This work was consuming a lot of time for both students and teachers. This application is developed to make it easier for teachers and students to view their routine easily and to book a room at their desired time. We consider our “DIU Class Schedule” app will be beneficial to both the Teachers and the students. Before the app was developed students had to keep pictures of the routine on their phone. Teachers and students had to go to the department office to book a classroom which was time-consuming. Another problem was that teachers often overlapped their room booking schedule with other teachers. So, one of them have to cancel their class and this was hampering the education of the students and valuable times of our teacher. It was also hard to find information about some teachers on the internet. So, we have also taken in our consideration and added features for getting any teacher information. We believe our app will save the time and unimportant effort for teachers and students in these scenarios. Moreover, we decided to keep it free without any cost, and also, we did not use any types of Ads so that students and teachers can get a good user experience using our application. It is also made open source so if anyone wants, they can update or improve this app in the future. Which will help to many feature improvements and various bug fixes.

7.2 Scope for Further Developments

For simplicity, the system is originally developed for viewing routines, updating routines, and booking classrooms. Gradually we have added more and more options to the app like viewing teacher information. Although we already have a lot of features there is always room for more improvements. “Class Test” and “To-do” options are planned to be added so that students can set a reminder for their class test and homework. As previously we have discussed that our application depends on a REST API. So, this means this REST API can be used from any platform. So, it is possible to make an iOS version of our application. The advantage of making an iOS application will be that the developer will not need to work on the server backend again as it is

already ready. This was all that we can only think about. But we can that different people can have different fascinating ideas. So, to get ideas and new features from other students of our university we have made our application open source and make it available to everyone. As the application will be open source if someone has any other interesting ideas in there that can increase its feature their contribution will be welcomed. We have made this application with the latest and the best technology possible. It can be a great scope to the new learners who are willing to learn about the latest architecture and design patterns in the field of “Web Development”.

REFERENCE

- [1] What are System Requirements Specifications, available at <<<https://www.inflectra.com/ideas/topic/requirements-definition.aspx#:~:text=What%20is%20a%20System%20Requirements,a%20system%20or%20software%20application>>>, last accessed on 24-09-2020 at 12.30PM.
- [2] Agile SDLC available at <<<https://mlsdev.com/blog/agile-sdlc>>>, last accessed on 22-10-2020 at 12.35PM.
- [3] What is an Entity Relationship Diagram (ERD), available at <<<https://www.smartdraw.com/entity-relationship-diagram/>>>, last accessed on 23-10-2020 at 12.50PM.
- [4] Use case diagram, available at <<https://en.wikipedia.org/wiki/Use_case_diagram#:~:text=A%20use%20case%20diagram%20at,which%20the%20user%20is%20involved>>, last accessed on 26-11-2020 at 12.45PM.
- [5] Logical Schema, available at <<https://en.wikipedia.org/wiki/Logical_schema#:~:text=Logical%20data%20models%20represent%20the,they%20relate%20to%20one%20another>>, last accessed on 27-11-2020 at 12.30PM.
- [6] Using REST API in mobile app, available at <<<https://www.techicy.com/using-the-rest-api-in-mobile-app-development.html>>>, last accessed on 28-11-2020 at 03.30AM.

ORIGINALITY REPORT

4/30/2021

Turnitin

<h2>Turnitin Originality Report</h2>	
<p>Processed on: 30-Apr-2021 04:15 +06 ID: 1573801207 Word Count: 5957 Submitted: 1</p>	
<p>Similarity Index</p> <h1>15%</h1>	<p>Similarity by Source</p> <p>Internet Sources: 14% Publications: 1% Student Papers: 10%</p>
<p>Zihad By Nusrat Jahan</p>	

<p>7% match (Internet from 05-Mar-2020) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/3693/P13410%20%2828%25%29.pdf?isAllowed=v&sequence=1</p>
<p>2% match (student papers from 20-Jan-2021) Submitted to Daffodil International University on 2021-01-20</p>
<p>1% match (Internet from 20-Apr-2021) https://www.inettutor.com/source-code/mobile-nutrition-app-with-admin-panel-capstone-documentation/</p>
<p>< 1% match (Internet from 02-Apr-2021) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5355/162-15-8132%20%2811-%29.pdf?isAllowed=v&sequence=1</p>
<p>< 1% match (Internet from 27-Aug-2020) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/4173/P15398%20%2820-%29_.pdf?isAllowed=v&sequence=1</p>
<p>< 1% match (Internet from 07-Apr-2021) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5398/161-15-7475%20%2827-%29.pdf?isAllowed=v&sequence=1</p>
<p>< 1% match (Internet from 07-Apr-2021) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/4125/151-15-258%3d24_.pdf?isAllowed=v&sequence=1</p>
<p>< 1% match (Internet from 07-Apr-2021) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5207/162-15-7914%20%2825-%29.pdf?isAllowed=v&sequence=1</p>
<p>< 1% match (Internet from 23-Mar-2021) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/20.500.11948/2717/142-15-3634.pdf?isAllowed=v&sequence=2</p>
<p>< 1% match (Internet from 02-Apr-2021) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5320/162-15-8066%20%2823-%29.pdf?isAllowed=v&sequence=1</p>
<p>< 1% match (Internet from 07-Apr-2021) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5518/161-15-7028%20%2823-%29.pdf?isAllowed=v&sequence=1</p>
<p>< 1% match (student papers from 09-Apr-2018) Submitted to Daffodil International University on 2018-04-09</p>
<p>< 1% match (student papers from 12-Jan-2021) Submitted to Daffodil International University on 2021-01-12</p>
<p>< 1% match (student papers from 04-Apr-2018)</p>

https://www.turnitin.com/newreport_printview.asp?eq=1&eb=0&esm=0&oid=1573801207&sid=0&n=0&m=2&svr=46&r=33.06468131401967&lang=en... 1/15