

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

# COSTAZ DESKTOP

(Teacher's Assistant)



By:

Muhammad Ammar Khan

BSCS-F19-M-63

Muneeb Ahmed

BSCS-F19-M-78

A Project Submitted

In partial fulfillment of the requirements for the degree of  
BACHELOR'S IN COMPUTER SCIENCE

University of Punjab Jhelum Campus  
Department of Information Technology

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## **Copyright 2023 by PUJC Students**

All rights reserved. Reproduction in the whole or in the form requires prior written permission of Muhammad Ammar Khan and Muneeb Ahmed.

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## **DEDICATION**

*“Dedicated to our Parents and Teachers, without their wholehearted support, encouragement, and guidance, it would have been impossible for us to make this project”*

## CERTIFICATE OF APPROVAL

This is to clarify that following students have successfully completed the final year project named as “Costaz Desktop” at University of the Punjab Jhelum Campus, to fulfill the partial requirements of the degree of Bachelor of Computer Science.

Sr.	Student's Name	Student's Roll no
1	Muhammad Ammar Khan	BSCS-F19-M-63
2	Muneeb Ahmed	BSCS-F19-M-78

**Supervisor** -----

Ms. Aminah Ali

Department of Information Technology

University of the Punjab, Jhelum Campus

**Project Office Coordinator** -----

Ms. Aminah Ali

Department of Information Technology

University of the Punjab, Jhelum Campus

**Head of Department** -----

Ms. Nadia Mumtaz

Department of Information Technology

University of the Punjab, Jhelum Campus

**Administrator** -----

Dr. Muhammad Mudasar Ghafoor

Department of Commerce

University of the Punjab, Jhelum Campus

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## ACKNOWLEDGENTS

All glory to the Almighty **ALLAH (S.W.T)** the creator of the universe, who leads us in the sea of darkness and enables us to overcome adversity in difficult situations, Allah (S.W.T) has power overall. We lift up, ask for his help, mercy and strength in resisting our evil and corrupt practices. I want to thank **ALLAH (S.W.T)** for making this project possible with so much kindness and compassion.

All the respect and love of **HAZRAT MUHAMMAD (PUBH)** that enables us to see the philosophy of life.

The project is the result of a partnership.

Many thanks to Ms. Aminah Ali Our Project Manager, her supervision helped us to produce this project in a timely and efficient manner.

Also, Thanks to our friends, for their support.

Date: 10 March 2023.

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## PROJECT COMPLETION CERTIFICATE

It is confirmed that the report entitled "Costaz Desktop" confirms that Muhammad Ammar Khan Roll# BSCS-F19-M-63, & Muneeb Ahmed Roll# BSCS-F19-M-78 Bachelor of Computer Sciences students from the Department of Information Technology, University of Punjab Jhelum Campus contains sufficient resources required to offer the above degree.

**Ms. Aminah Ali** (Supervisor)

Lecturer

Email: aminaali@pujc.edu.pk

Department of Information Technology  
University of the Punjab, Jhelum Campus

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## DECLARATION

It is declared that this is an original piece of our work, expect where acknowledgement in text and references. This work has not been submitted in any form for other degree or diploma at any university or other institution for tertiary education and shall not be submitted by me in future for obtaining any degree from this or any other university.

Muhammad Ammar Khan

BSCS-F19-M-63

\_\_\_\_\_

Muneeb Ahmed

BSCS-F19-M-78

\_\_\_\_\_

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## ABSTRACT

The Application called “COSTAZ DESKTOP” is being developed to help teachers and manage the attendance and academic records of their students. The application is designed to improve the efficiency and effectiveness of teachers by allowing them to generate attendance sheets for individual students as well as for specific course. Costaz Desktop can also store data in Google drive, making it accessible from anywhere at any time. The name “COSTAZ” is a combination of words “CO” which represents an Assistant and “OSTAZ” (استاذ) which means teacher in Arabic.



## Table of Contents

<b>1. INTRODUCTION .....</b>	<b>11</b>
1.1 Project Title .....	11
1.2 Project Overview Statement.....	11
1.3 Project Overview Statement Template .....	11
1.4 Project Goals & Objectives .....	12
1.5 High-level system components .....	13
1.6 List of optional functional units .....	13
1.7 Exclusions .....	13
1.8 Application Architecture .....	14
1.9 Software and Hardware Specification .....	15
1.10 Gantt chart.....	15
1.11 Tools and technologies used with reasoning.....	15
<b>2. PROJECT/PRODUCT FEASIBILITY REPORT.....</b>	<b>16</b>
2.1 Feasibilities: .....	16
2.2 Project/Product Scope .....	19
2.3 Project/Product Costing .....	19
2.4 Critical Path Method .....	20
2.5 Gantt chart.....	22
2.6 Introduction to Team members and their skillset.....	23
2.7 Tools and Technology with reasoning .....	23
2.8 Vision Document.....	24
2.9 Risk List.....	24
<b>3 REQUIREMENTS ENGINEERING.....</b>	<b>25</b>
3.1 Systems Specifications .....	25
3.2 Organizational Chart .....	26
3.3 Scope of the System .....	27
3.4 Context Level Data Flow Diagram.....	29
3.5 Capture "shall" Statements.....	30
3.6 Allocate Requirements.....	30
3.7 Prioritize Requirements.....	33
3.8 Requirements Trace-ability Matrix.....	36
3.9 High-level Use case Diagram.....	40
<b>4 USE CASES .....</b>	<b>41</b>
4.1 Use case Description.....	41
4.2 Use-case Diagram.....	65
4.3 Data Model.....	66
<b>5 DOMAIN MODEL .....</b>	<b>67</b>
<b>6. SEQUENCE DIAGRAM .....</b>	<b>68</b>
<b>7. COLLEBARATION DIAGRAM.....</b>	<b>70</b>
<b>8. OPERATION CONTRACT .....</b>	<b>70</b>
<b>9. DESIGN CLASS DIAGRAM.....</b>	<b>73</b>
<b>10.MANUALS .....</b>	<b>74</b>

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

<i>10.1 Home Page.....</i>	<i>74</i>
<i>10.2 Create New Class.....</i>	<i>75</i>
<i>10.2.1 Context Menu.....</i>	<i>75</i>
<i>10.2.2 Editing Class.....</i>	<i>76</i>
<i>10.2.3 Deleting Class.....</i>	<i>76</i>
<i>10.3 Student Page .....</i>	<i>77</i>
<i>10.3.1 Context Menu.....</i>	<i>77</i>
<i>10.3.2 Adding Student Details.....</i>	<i>78</i>
<i>10.3.3 Editing Student Details .....</i>	<i>78</i>
<i>10.4 Session Management.....</i>	<i>79</i>
<i>10.4.1 Delete Session .....</i>	<i>79</i>
<i>10.5 Section Tile.....</i>	<i>80</i>
<i>10.5.1 Creating New Section.....</i>	<i>80</i>
<i>10.5.2 Context Menu .....</i>	<i>81</i>
<i>10.5.3 Editing Section .....</i>	<i>81</i>
<i>10.5.4 Delete Section .....</i>	<i>81</i>
<i>10.6 Settings Page.....</i>	<i>82</i>
<i>10.7 Adjustable Sidebar .....</i>	<i>82</i>
<i>10.8 Theme Modes .....</i>	<i>83</i>
<i>10.9 Records Preview .....</i>	<i>84</i>
<b>10 TESTING .....</b>	<b>85</b>
<i>10.1 Test case Class creation.....</i>	<i>85</i>
<i>10.2 Test case Editing class .....</i>	<i>85</i>
<i>10.3 Test case Create session.....</i>	<i>86</i>
<i>10.4 Test case Create section.....</i>	<i>86</i>
<i>10.5 Test case Editing section.....</i>	<i>86</i>
<i>10.6 Test case Adding students .....</i>	<i>87</i>
<i>10.7 Test case Editing student details .....</i>	<i>87</i>
<b>PROJECT COMPLETION CERTIFICATE .....</b>	<b>88</b>

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 1. Introduction

With the increasing number of students in schools, teachers are often found busy managing the attendance of their students. A teacher's assistant application is being developed, which will help teachers to manage the attendance sheet generation and reportcard generation. This application can be used by teachers to perform their daily duties more efficiently and effectively as they can now also be used for storing data in Google Drive, so that they can be accessed anywhere at any time. The name of this application is **Costaz**.

The word Costaz is derived from two words: “co-” which is used for the assistant agent, and “ostaz” (استاذ) which translates to “Teacher” in Arabic. **Costaz Desktop** is a teacher-friendly desktop application developed to manage academic records of their students. It allows for attendance sheets generation in Excel form, report card generation for individual students, and generation of marks sheets for all students, as well as for a specific course.

### 1.1 Project Title

**Costaz Desktop** – a Teacher’s Assistant Desktop Application

### 1.2 Project Overview Statement

The aim of the project is to develop a desktop application which can be used by teachers to manage the attendance of their students and also generate various reports in Excel sheet format. The application will use Google Sheets as its database and Google Drive to store the Excel sheets. The application will be developed using the Flutter framework.

### 1.3 Project Overview Statement Template

Project Title: <b>Costaz Desktop</b> – Teacher’s Assistant			
Group Leader: Muhammad Ammar Khan – BSCS-F19-M-63			
Project Members:			
Name	Registration #	Email Address	Signature
Muhammad Ammar Khan	BSCS-F19-M63	bscs.f19.m63@gmail.com	
Muneeb Ahmed	BSCS-F19-M78	bscs.f19.m.78@gmail.com	

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

Project Goal: To develop a desktop application that will be used by teachers to assist them in the management of student attendance, report cards and marks.	
Objectives:	
Sr.#	
1	Management of Attendance of Students.
2	Management of Report Cards of Students.
3	Generation of Records in Excel format.
4	Using Google Sheets as a lightweight Database.
5	Using Google Drive to store the Excel sheets.
6	Data Visualization
Project Success criteria:	
<ul style="list-style-type: none"> <li>- Attractive User Interface (UI), and great User Experience (UX)</li> <li>- Maximum Bug-free Experience.</li> <li>- Efficient usage of APIs.</li> <li>- Minimal backend, only as required.</li> </ul>	
Assumptions, Risks, and Obstacles:	
<ul style="list-style-type: none"> <li>- We need to use Google's API for many purposes, it may get a little tedious to work with.</li> <li>- Flutter is a relatively new technology.</li> <li>- Optimization can take time.</li> </ul>	
Organization Address (if any):	
Type of project:	<input type="checkbox"/> Research <input type="checkbox"/> Development
Target End users: Teachers	
Development Technology:	<input type="checkbox"/> Object-Oriented <input type="checkbox"/> Structured
Platform:	<input type="checkbox"/> Web based <input type="checkbox"/> Distributed
<input type="checkbox"/> Desktop based	<input type="checkbox"/> Setup Configurations <input type="checkbox"/> Other
Suggested Project Supervisor: Miss. Amina	
Approved By:	
Date: 01 September 2022	

### 1.4 Project Goals & Objectives

The main goal of the Teacher's Assistant Application is to make the work of the teachers easier by providing them with a desktop application through which they can manage the attendance and report cards of their students easily.

The objectives of the application are as follows:

- To provide an easy-to-use interface for the teachers to manage their students' attendance and report cards.
- To generate the sheets in Excel format so that they can be easily shared and stored.
- To use the Google Account of the teacher to allow access to the Google Sheets, and Google Drive.
- To provide data visualization.

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## ***1.5 High-level system components***

- **Authentication:** The Teacher will be authenticated by their Google Account, to keep perfect integration with other Google Services.
- **User Interface:** The interface will be developed using Flutter, which will give the application a Material Design' look and feel.
- **Business Logic:** The logical section will be developed in Dart Language in Flutter framework, to make the application partially platform independent.
- **Database:** The database will be created using Google Sheets, to keep and manage the data of the students.
- **Data Storage:** The data of the students in the form of Excel Sheets will be stored on Google Drive.
- **Data Visualization:** Pie Charts, and Bar Graphs will be used as Data Visualization. The teacher can switch between the two if possible.
- **Compatibility:** Costaz Desktop is being initially developed for desktop only, but will be made cross-platform in the future. Costaz Desktop will be compatible with Costaz Android as well.

## ***1.6 List of optional functional units***

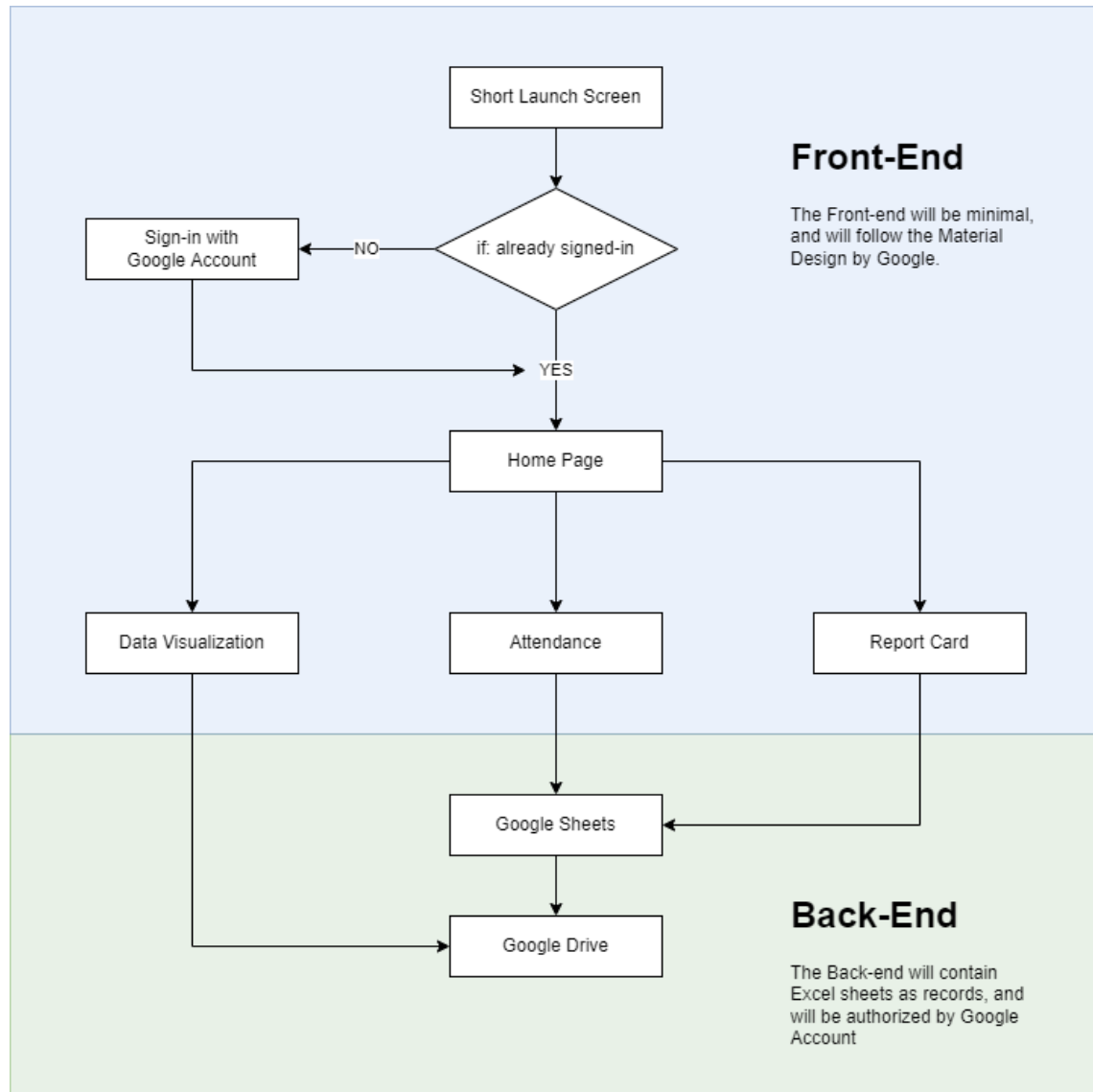
- **Expressive Sheets:** The rows of Attendance Sheets will be colored based on student's attendance.
- **Time Table:** Weekly Time Table scheduling, and reminders of upcoming events. Google Calendar can be used for alerts and reminders.
- **Student's Inquiry Portal:** Google Form integration for this purpose.
- **Online Class Arrangement:** Google Meet integration, and the invitation links will be shared to all the students of the particular class, and incoming students will be verified by their email (listed on Google Sheets), and will be automatically entered.
- **Inbox:** Only the mails of the students will be shown, and can be viewed in this section.
- **Personal Pocket:** A folder to store miscellaneous data. Data will be stored on a reserved place in Google Drive.

## ***1.7 Exclusions***

- **Notes Taking:** Notes taking features could also be possible, but it would require a system to handle whiteboard, which would require extra resources.
- **Post System:** A system like Facebook to post announcements or other data, could also be possible, but it would require a dedicated system to handle it.
- **Scheduling:** To make Costaz Desktop more independent, and lightweight, the scheduling tasks (i.e., room allocation etc.) are currently not supported.

## 1.8 Application Architecture

Costaz Desktop will use the **Client-Server** Architecture, in the following way;

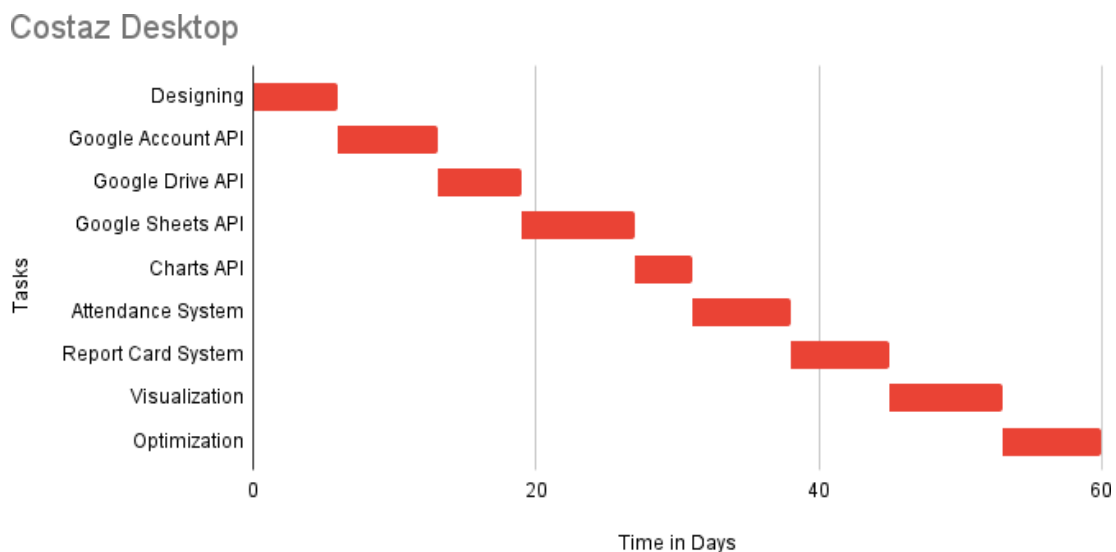


## 1.9 Software and Hardware Specification

Costaz Desktop is a Desktop application, and it has the following requirements;

- Windows 7/10/11 (latest build).
- 4 GB of RAM at least.
- Ethernet Connection.
- Google Account (personal/work).

### 1.10 Gantt chart



### 1.11 Tools and technologies used with reasoning

- **Flutter**  
Flutter framework is chosen due to the following reasons:
  - Dart compiles to C++ for Windows builds, which can be compiled to native binaries by a desired compiler (GCC/MSVC/Clang).
  - Plugin support.
  - Access to Google's Material Design and Library.
- **Google Account**  
The teacher's Google account will be used to access the Google Services API.
- **Google Sheets**  
For better integration with other Google Services.
- **Google Drive**  
To ensure reliability, security and trust, Google Drive is the perfect choice for data storage.

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## ***2. Introduction***

**Costaz Desktop** is a teacher-friendly desktop application developed to manage academic records of their students. It allows for attendance sheets generation in Excel form, report card generation for individual students, and generation of marks sheets for all students, as well as for a specific course. This application can be used by teachers to perform their daily duties more efficiently and effectively as they can now also be used for storing data in Google Drive, so that they can be accessed anywhere, anytime.

### ***2.1 Project/Product Feasibility Report***

**Costaz Desktop** provides a system to efficiently manage the records of the students in regular basis, and to generate reports. The feasibility of the **Costaz Desktop** is described in detail below, with respect to the following feasibility criteria;

- Technical
- Operational
- Economic
- Schedule
- Specification
- Information
- Motivational
- Legal and Ethical

#### ***2.1.1 Technical Feasibility***

**Costaz Desktop** is being developed using Dart language in Flutter framework. To provide cloud-services, and accessibility features, Google Sheets is being used as a Database for students' records, and Google Drive for a convenient and reliable storage medium for Excel sheets. Furthermore, the plugin, "GoogleAPIs" is being used to access all the APIs provided by the Google, and for better integration with Flutter framework.

#### ***2.1.2 Operational Feasibility***

Costaz Desktop allows teachers to manage their students' attendance, report cards and other information in an intuitive manner. Costaz Desktop is designed to be easy to use and operationally feasible. Detailed operational feasibility is as follows;



PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

- **Attendance:** Attendance marking is being made very easier, by providing following features:
  - Attendance can be traditionally marked one-by-one.
  - It can be marked for specific students as well, for which an additional view is being provided, and the performance of that student in that particular subject will be shown and visualized as well.
  - The Feature of “Select All” is being provided to mark all the (selected) students as “Present/Absent/Leave”.
- **Report Card Generation:** The report cards of the students will be managed more reliably and easily by providing a dedicated feature for this purpose. An option for Report Card Generation is being provided to the context menus of every view. Report cards can be generated in the following ways;
  - Student wise list generation, and performance visualization.
  - Course wise list generation, for the teachers who are teaching multiple subjects to the same class.
  - Class wise list generation, with performance analysis.
- **Data Storage:** The following features will be introduced for this purpose;
  - Data will be stored online, to reduce the chance of data loss, and maximize the accessibility of the data.
  - Exports will be in Excel format to improve the portability, and maximize the application-independence of the generated data.
- **Data Visualization:** Every collection of records will be visually presented in the form of preferred graph as well.

### ***2.1.3 Economic Feasibility***

Since the Application is desktop based, there will be no cost for any web-hosting. Furthermore, as (Teacher’s) Google Drive is being used as Data Storage service, there will be no additional cost of cloud data storage. All the configurations and settings will also be stored on Google Drive, so there will be no need to pay for any external database. Hence, Costaz Desktop is Economically Feasible for the developers, as well as for the customers.

### ***2.1.4 Schedule Feasibility***

The project is being developed in modules (as incremental development), and these modules will be gradually linked together timely (according to the Gantt Chart). January 2023 is the estimated completion date of our project.

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

### 2.1.5 *Specification Feasibility*

The requirements of **Costaz Desktop** are being successfully met, which are as follows;

- Feasible Attendance management system
- Efficient Reports Generation, at each level
- Data Visualization
- High-Performant framework: Flutter
- Maximizing the use of Open-Sourced modules: Flutter Plugins
- Using no-cost solutions for data handling
  - Google Sheets as Database
  - Google Drive for Data Storage, and Database

Hence, Costaz Desktop is feasible to develop, and satisfies all the requirements.

### 2.1.6 *Information Feasibility*

**Costaz Desktop** is a Desktop Application, which uses Google Account for Authentication, to gain access to Google Sheets and Google Drive linked to that Account. Google Services ensure security, reliability, and trust. Similarly, the Flutter framework is being used, which ensures maximum performance, maintainability, and portability.

### 2.1.7 *Motivational Feasibility*

Since our team members have all the tasks divided, and everyone is doing their work with keen interest, and new ideas with research are constantly being generated upon brainstorming. Moreover, the team also has technical development experience. Hence, the project is motivationally feasible.

### 2.1.8 *Legal & Ethical Feasibility*

Since Flutter framework and its plugins are being used which are all open-source, and Google Services which are being used are also free to use. Hence, **Costaz Desktop** is legally feasible.

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 2.2 Project/Product Scope

With the increasing number of students in schools, teachers are often found busy managing the attendance of their students. A teacher's assistant application is being developed, which will help teachers to manage the attendance sheet generation and reports generation. This application can be used by teachers to perform their daily duties more efficiently and effectively as they can now also be used for storing data in Google Drive, so that they can be accessed anywhere at any time. The name of this application is **Costaz**.

Costaz Desktop solves the basic Teachers' needs, and utilizes the most reliable, secure and trusted Google's Services. For instance, Google Sheets is being used as the database to store students' data and to generate reports easily, and Google Drive is being used as primary distributed storage for ease-of-access. Google Drive also eliminates the need for any external database, which could have to be explicitly managed. Google Account allows the use of all the Google Services.

## 2.3 Project/Product Costing

**Costaz Desktop** is being developed by a team of two developers only. The cost parameters include;

- 2.3.2 Labor Cost
- 2.3.3 Google Workspace cost
- 2.3.4 Cost of Google Sheets API
- 2.3.5 Licensing Cost

This sums up-to **~30k/-**

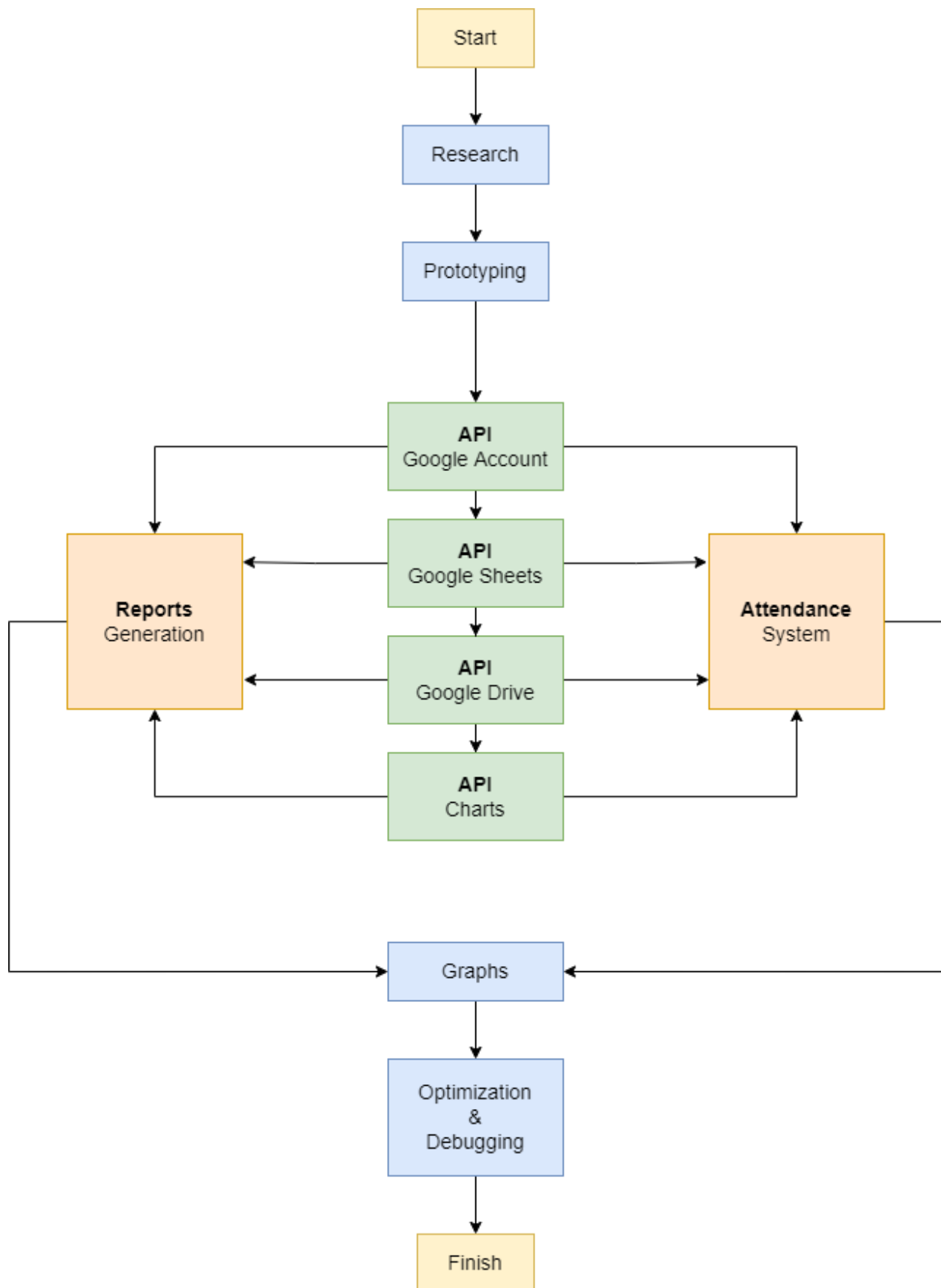
### 2.3.1 Project Cost Estimation by using Basic COCOMO'81

Type	Effort	Schedule	People Required
Organic	$PM = 2.4 \times (50)^{1.05}$	$TD = 2.5 \times (146)^{0.38}$	$PR = PM / TD$

Results		
PM	Person Month	145.9
kLOC	Lines Of Code (in thousands)	~50
TD	Total Duration (in months)	2.3
People	People Required	63

## 2.4 CPM – Critical Path Method

Activity	Description	Immediate Predecessor	Duration (Weeks)
A	Research on the Idea	-	6
B	Prototyping	A	7
C	Managing Google Account API	B	6
D	Managing Google Sheets API	C	8
E	Managing Google Drive API	D	4
F	Managing Charts API from 'E'	E	7
G	Attendance System	B, C, D, E	7
H	Report Cards System	B, C, D, E	8
I	Graphs Representation	G, H	7
J	Optimization & Debugging	I	7



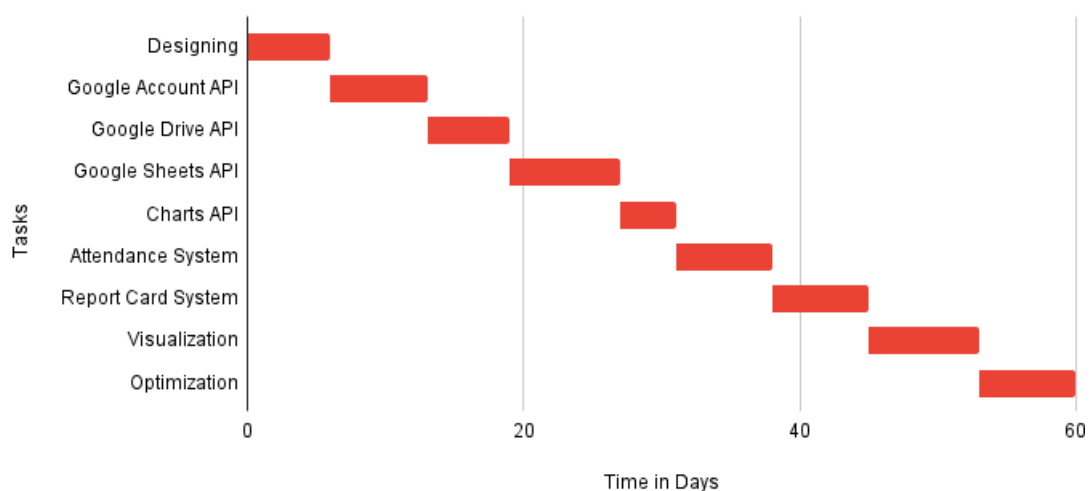
PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

Activity	Duration	ES	EF	LS	LF	TS	FS
A	6	0	6	20	26	20	0
B	7	6	13	26	33	20	0
C	6	13	19	33	39	20	0
D	8	19	27	31	39	12	0
E	4	27	31	19	39	-8	0
F	7	31	38	32	39	1	0
G	7	38	45	39	46	1	1
H	8	38	46	38	46	0	0
I	7	46	53	46	53	0	0
J	7	53	60	53	60	0	0

Critical Path
A → B → C → D → E → F → H → I → J

## 2.5 Gantt chart

### Costaz Desktop



PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 2.6 Introduction to Team members and their skill set

**Costaz Desktop** consists of a team of 2 members, with the following tasks and responsibilities assigned;

- **Muhammad Ammar Khan**

The Group leader, assigned to the following responsibilities;

- Development
- Brainstorming
- Logic Designing

- **Muneeb Ahmed**

The Researcher, assigned to the following responsibilities;

- Research on features, and ideas
- Research on Open-Source modules
- Research on Google Services APIs
- Critical Analysis
- Screens Designing

## 2.7 Tools and Technology with reasoning

- **Flutter**

Flutter framework is chosen due to the following reasons:

- Dart compiles to C++ for Windows builds, which can be compiled to native binaries by a desired compiler (GCC/MSVC/Clang).
- Plugin support.
- Access to Google's Material Design and Library.

- **Google Authentication Service**

The teacher's Google account will be used to access the Google Services API.

- **Google Sheets**

For better integration with other Google Services.

- **Google Drive**

To ensure reliability, security and trust, Google Drive is the perfect choice for data storage.

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 2.8 Vision Document

**Costaz Desktop** is a free and easy-to-use solution for the teachers. It provides a unified environment where they can work on their students' attendance, grades, and report cards. Additionally, the software stores all of their content on their Google Drive that can be accessed from any computer.

**Costaz Desktop** will be available only on Desktop. Users will be able to seamlessly access their data from any device, where Costaz Desktop will be installed. A user's Google Account will be used as a gateway to access Google Services, like Google Sheets to create the database, and their Google Drive to store all the data. The app works only in Online mode so that users can work on the go, and changes will be saved instantly, as the user enters the data, without worrying about the data-loss.

## 2.9 Risk List

**Costaz Desktop** is being developed by remaining on the safe-zone at its best, still there are some risks to this project, which are as follows;

- Google APIs**  
 Since the project is heavily dependent on Google Services, any change in API call structure will require an immediate refactor of **Costaz Desktop Engine**.
- Time**  
 Since the structure of Costaz Desktop is entirely unique, it will require custom widget-engineering in Flutter, which can take significant amount of time.
- Compactness, and Simplicity**  
 Costaz Desktop shall be maximum compact in binary and code size, and simplistic in design. Due to its featureful nature, maintaining the code quality and code size is a big challenge.
- Engine**  
 The Core of Costaz Desktop will be initially developed, which will contain the API call management, and all the common functions. Any failure in the Engine can be catastrophic.



PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

\*\*\*\*\*REQUIREMENTS ENGINEERING\*\*\*\*\*

## 3.1 Systems Specifications

### Introduction

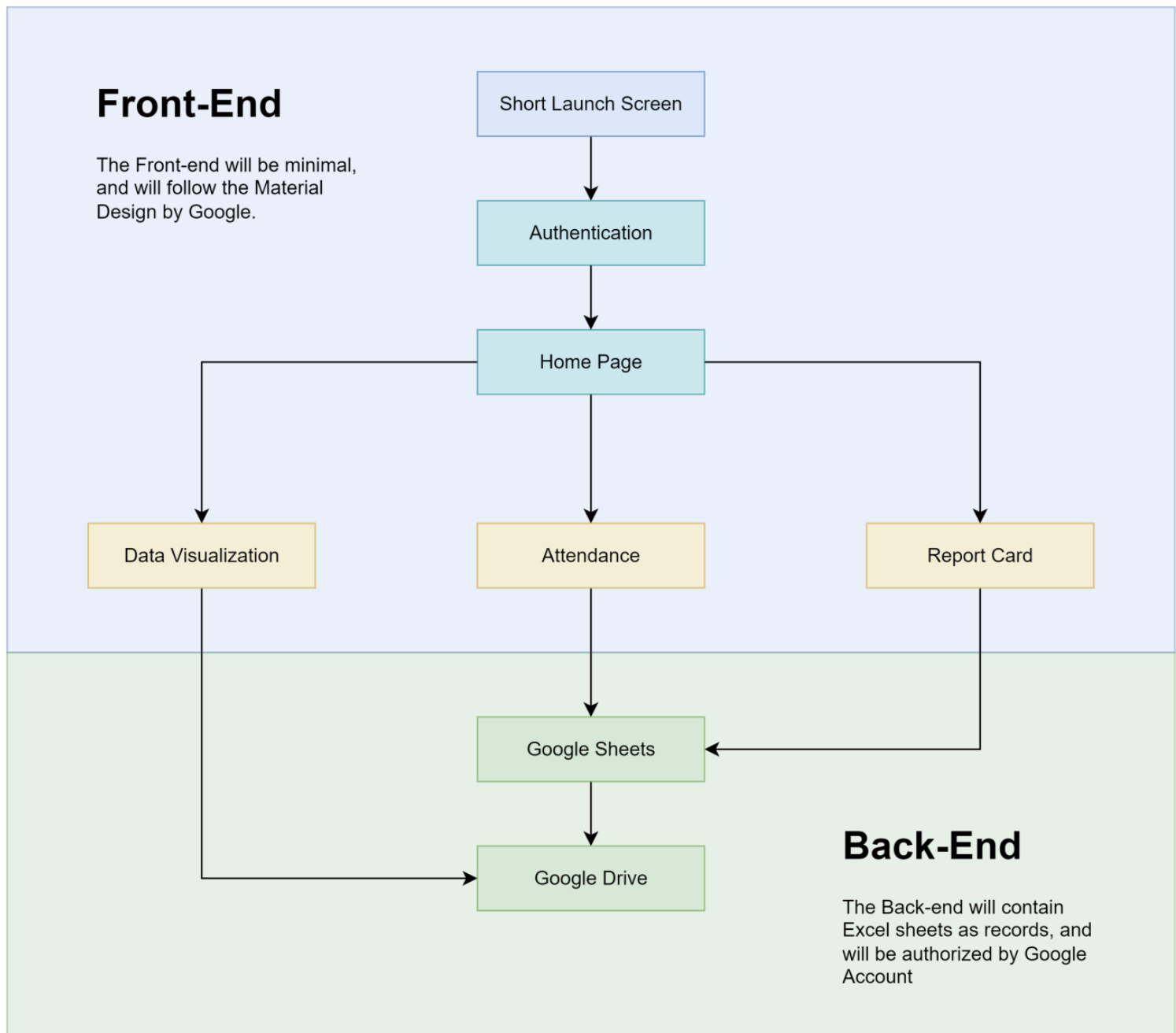
**Costaz Desktop** is a teacher-friendly desktop application developed to manage academic records of their students. It allows management of students' attendance and reports generation, and also helps visualize the data.

**Costaz Desktop** is being developed by a team of two developers, residents of Jhelum, Pakistan. The application was initially developed for universities, but now it can be used by any organization that needs to track the progress of its employees or students.

### Existing System

**Costaz Desktop** allows teachers to automate existing traditional attendance systems, by enabling Google Services. It allows for attendance sheets generation in Excel form, report card generation for individual students, and generation of marks sheets for all students, as well as for a specific course — all by using Google Sheets API. This application can be used by teachers to perform their daily duties more efficiently and effectively as they can now also be used for storing data in Google Drive, so that they can be accessed anywhere, anytime.

## 3.2 Organizational Chart



PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

### 3.3 Scope of the System:

The scope of the **Costaz** application is divided into four phases:

#### Phase 01:

This phase involves the authentication of the user, which includes:

- Launch Short Screen
- Sign in/up with Google Account
- Email Authentication
- Homepage

#### Phase 02:

This phase involves the data visualization of **Costaz Desktop**, which includes:

- Homepage
- Data Visualization
- Google Drive

#### Phase 03:

This phase involves the attendance system of the **Costaz Desktop**, which includes:

- Homepage
- Attendance
- Google Sheets
- Google Drive

#### Phase 04:

This phase involves the attendance system of the **Costaz Desktop**, which includes:

- Homepage
- Report Card
- Google Sheets
- Google Drive

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## Summary of Requirements (Initial Requirements)

**Costaz Desktop** considers the following requirements;

- **Attendance System**

A full-fledge, easy-to-use, non-redundant attendance system should be provided, which can replace the traditional attendance marking — still following the similar standards.

- **Reports Generation**

Report Cards should not have to be separately created, but should be generated by the information available, and provided by the user (teacher). The reports can also contain visualizations, like graphs, and charts, indicating students' performance. Reports should be easy to generate and shared to the students.

- **Online Services**

**Costaz Desktop** aims to opt-for modern computations, and feasibility standards, but still wants to follow traditional layout and schemes of the educational-institutes. To cope with the situation, Google Services are being used to store, compute, and visualize the data.

### 3.3.1 Identifying External Entities

#### Over Specifying Entities

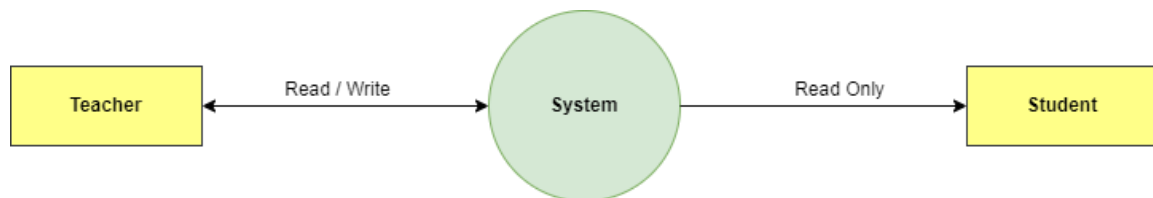
- Teacher
- Student
- Google Services
  - Google Authentication Service
  - Google Drive
  - Google Sheets

#### Performing Refinements

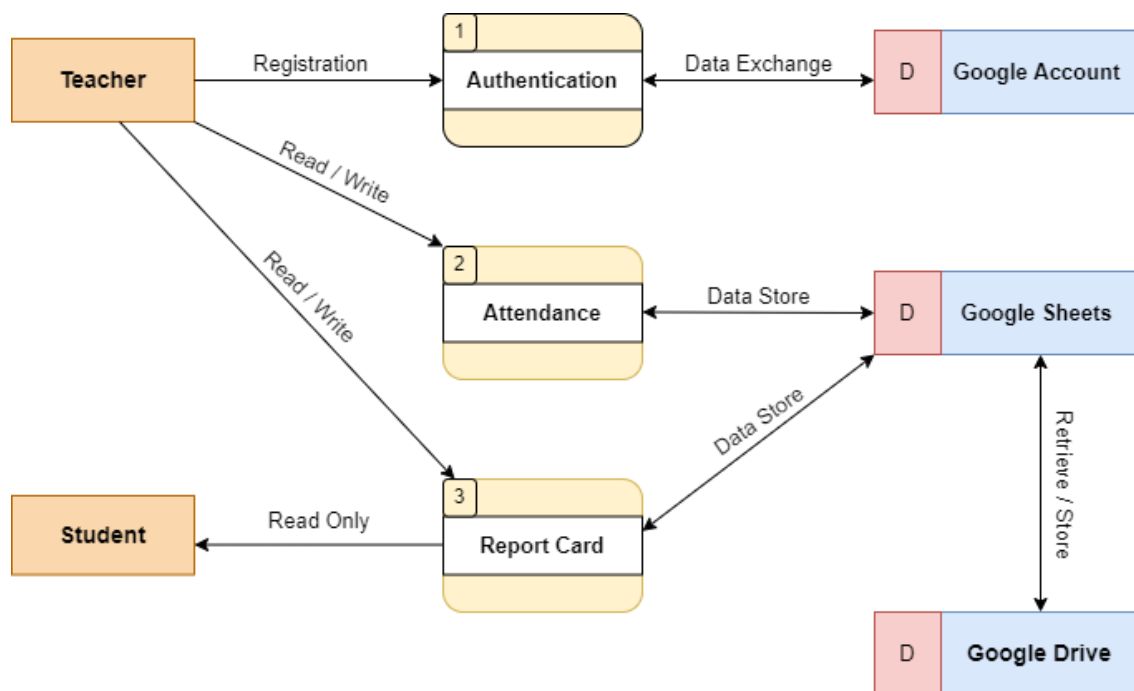
- Teacher
- Google Services

### 3.4 Context Level Data Flow Diagram

#### Level 0:



#### Level 1:



### 3.5 Capture “shall” Statements

Para	External Entity	Initial Requirements
1.0	Teacher	A teacher “shall” open the application.
1.0		A teacher “shall” provide their Google Account for Authentication Key.
1.0		A teacher “shall” go-to the main menu bar to access more features.
1.0		A teacher “shall” create the class card or course card.
1.0		A teacher “shall” right-click on the card for the context menu.
1.0		A teacher “shall” provide all the related details of a student to an application
1.0		A teacher “shall” mark a student’s status (present/absent) in a class.
1.0		A teacher “shall” access the attendance sheet in their Google Drive
1.0		A teacher “shall” enter each student's marks separately.
1.0		A teacher “shall” generate the report card.
2.0	Google	Google “shall” authenticate the user.
2.0		Google APIs “shall” allow users to interact with the application.
2.0		Google Sheets “shall” be used as a database to keep student information in it.
2.0		Google Drive “shall” store the student stuff and personnel files in separate folders so that teachers can be more organized with their work.
2.0		Google Drive “shall” remove barriers with allowing sharing

### 3.6 Allocate Requirements

Para	Initial Requirements	Use Case Name
1.1	The teacher “will” sign up with their Google account to get verified to use Google Drive and Google Sheets.	UC_Authentication
1.2	A teacher “will” customize their experience by configuring the settings.	UC_Settings
1.3	The Navigation bar “will” be displayed on the left side of the screen.	UC_Navigation_Bar
1.3.1	A teacher “will” open the default page where all the performance of classes and courses will be available in the pie charts.	UC_Home
1.3.2	A teacher “shall” login to the system.	UC_Main
1.3.2.1	A teacher “shall” add the class of students on the homepage of the application.	UC_Class_Add

1.3.2.2	A teacher “will” create multiple class cards according to the schedule.	UC_Class_Card
1.3.2.2.1	A teacher “will” move to the next screen by clicking on the Class Card.	UC_Class_Primary_Action
1.3.2.2.2	A teacher “will” visualize an additional context menu on the screen with just a right click.	UC_Context_Menu
1.3.2.2.2.1	A teacher “shall” generate the report card of students in a group.	UC_Report_Generation
1.3.2.2.2.2	A teacher “will” delete a record file after it has been generated.	UC_Delete
1.3.2.2.2.3	A teacher “will” sometimes rename a report file after it has been generated.	UC_Rename
1.3.2.2.2.4	A teacher “will” visualize a pie-chart/bar-graph of how their students are behaving and performing.	UC_Performance
1.3.2.3	A teacher “shall” add the course for students in the Class Section.	UC_Course_Add
1.3.2.4	A teacher “will” set limits for the course, like total marks, marks for assignments, quizzes etc.	UC_Thresholds
1.3.2.5	A teacher “will” be able to create multiple course cards according to the schedule.	UC_Course_Card
1.3.2.5.1	A teacher “will” move to the next screen by clicking on Course Card.	UC_Course_Primary_Action
1.3.2.5.2	A teacher “will” be able to display a context menu on the screen by just right-clicking.	UC_Context_Menu
1.3.2.5.2.1	A teacher “will” generate a report for their students which will include the student's name, roll number, all the marks they obtained in that course, and the GPA of those marks.	UC_Report_Generation
1.3.2.5.2.2	A teacher “will” delete a record file after it has been generated.	UC_Delete
1.3.2.5.2.3	A teacher “will” be able to rename the selected cell to provide edited information of the student.	UC_Rename
1.3.2.5.2.4	A teacher “will” visualize a pie chart of their student's performance. In this visualization, each slice of the pie is one metric.	UC_Performance
1.3.2.5.2.5	A teacher “will” change the limits for both the grades of individual students and the number of courses.	UC_Edit_Thresholds
1.3.2.6	A teacher “will” select the class based on the available sections or shifts (mor/eve).	UC_Dropdown_Partition
1.3.2.6.1	A teacher “will” move to the next screen, where all the selected students' records are available by clicking the mouse over to the right.	UC_Dropdown_Primary_Action

1.3.2.6.2	A teacher “can” create a context menu with open and delete features by right-clicking.	UC_Context_Menu
1.3.2.6.2.1	A teacher “will” generate a report for their students that include the student's name and roll number, as well as all the marks they obtain in that class. The report will also include the GPA of those marks.	UC_Report_Generation
1.3.2.6.2.2	A teacher “will” delete the report file once it's generated.	UC_Delete
1.3.2.6.2.3	A teacher “will” rename the generated report and share it with the students.	UC_Rename
1.3.2.6.2.4	A teacher “will” visualize the data they receive from their records, so they can more easily spot the students who are performing exceptionally well, those who are doing OK, and those who need help.	UC_Performance
1.3.2.7	On click, the teacher “will” visualize the student's name, roll no, rank and CGPA.	UC_Student_Field
1.3.2.7.1	A teacher “will” move to the expanded screen of the record field by clicking left on their mouse.	UC_Student_Primary_Action
1.3.2.7.2	The teacher “can” visualize the attendance, and performance, create and generate reports, and reset and delete the field of the student by just clicking right on the mouse.	UC_Context_Menu
1.3.2.7.2.1	A teacher “will” open their student records individually in the field.	UC_Open
1.3.2.7.2.2	A teacher “will” create a report of students based on their attendance, performance, and grades.	UC_Create_Report
1.3.2.7.2.3	A teacher “can” generate student reports by clicking left on the Generate button on the screen.	UC_Generate_Report
1.3.2.7.2.4	A teacher “can” visualize how their students perform in assignments, presentations, and quizzes with pie charts – use them to quickly understand their strengths and weaknesses.	UC_Performance
1.3.2.7.2.5	A teacher “will” reset the record field in case if a student is missing or absent for extended periods of time.	UC_Reset
1.3.2.7.2.6	A teacher “can” move a student's field up or down by simply dragging it. This is helpful if your class list is in alphabetical order, and you want to rearrange students' fields for different activities.	UC_Drag_Drop
1.3.2.8	The teacher “will” either mark the attendance individually or by selecting all.	UC_Attendance_View
1.3.2.8.1	Each button will be colored differently. By default, students will be present, and the teacher just needs to mark the missing student’s attendance.	UC_Attendance_Button_PAL



1.3.2.9	A teacher “will” visualize the student’s report card, which will show a graphical representation of all the student's performances in the pie charts. The charts are made from attendance records, class activities and marks received	UC_Report_Card_View
1.3.2.9.1	A teacher “will” set limits for both the grades of individual students and the number of courses, on the fly.	UC_Add_OTF
1.3.2.9.2	A teacher “can” save the changes to their report card by pressing buttons below the generated report card.	UC_Generate_Report
1.3.3	A teacher “will” pin students who are doing better and put information in bookmarks for students who need more attention.	UC_Bookmarks
1.3.4	A teacher “will” usually put all the files in Google Drive.	UC_Files
1.3.5	A teacher “will” change the font size, theme, notification preferences, and more in the setting.	UC_Settings
1.3.6	A teacher “will” provide feedback on the application features or any bugs.	UC_Feedback
1.3.7	A teacher “will” get all the old, deleted material in the trash.	UC_Trash
1.3.8	A teacher “will” visualize our latest updates and will see the terms and conditions, privacy policy statement, and company location	UC_About

### 3.7 Prioritize Requirements

Para	Rank	Initial Requirements	Use Case ID	Use Case Name
1.0	Highest	The teacher “will” sign up with their Google account to get verified to use Google Drive and Google Sheets.	UC_1	UC_Authentication
1.0	Highest	A teacher “shall” add the class of student in the home page of application.	UC_2	UC_Class_Add
1.0	Highest	A teacher “shall” add the course in the top-level menu of the application.	UC_3	UC_Course_Add

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

1.0	Highest	The teacher “will” either mark the attendance individually or by selecting all.	UC_4	UC_Attendance_View,
1.0	Highest	A teacher “will” visualize the student report card which will show a graphical representation of the entire student's performance in the form of charts. The charts are made from attendance records, class activities and marks received.	UC_5	UC_Report_Card_View
2.0	Medium	The Navigation bar “will” be displayed on the left side of the screen.	UC_6	UC_Navigation_Bar
2.0	Medium	A teacher “will” open the default page where all the performance of classes and courses is available in the form of a graph.	UC_7	UC_Home
2.0	Medium	A teacher “shall” generate the report card of students individually or in a group.	UC_8	UC_Report_Generation
2.0	Medium	A teacher “will” visualize a pie chart of how their students are behaving and performing.	UC_9	UC_Performance
2.0	Medium	A teacher “will” set limits for both the grades of individual students and the number of courses	UC_10	UC_Tresholds
2.0	Medium	On click, Teacher “will” see the student's name, roll no, rank and CGPA.	UC_11	UC_Student_Field
2.0	Medium	A teacher “will” usually put all the files in Google Drive.	UC_12	UC_Files
2.0	Medium	A teacher “will” get all the old deleted material in the trash.	UC_13	UC_Trash
2.0	Medium	A teacher “can” save the changes to their report card by pressing buttons below the generated reportcard.	UC_14	UC_Generate_Report
2.0	Medium	Each button will be colored differently. By default, students will be present and the teacher just needs to mark the missing student’s attendance.	UC_15	UC_Attendance_Button_PAL
2.0	Medium	The teacher “can” visualize the attendance, and performance, create and generate reports, reset and delete the field of the student by just clicking right on the mouse.	UC_16	UC_Context_Menu
2.0	Medium	A teacher “will” select the class based on the available sections or shifts (mor/eve).	UC_17	UC_Dropdown_Partition
2.0	Medium	A teacher “will” pin students who are doing better and put information in bookmarks for students who need more attention.	UC_18	UC_Bookmarks

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

2.0	Medium	A teacher “will” change the font size, theme,notification preferences, and more in the setting.	UC_19	UC_Settings
3.0	Lowest	A teacher “will” create multiple class cards according to the schedule.	UC_20	UC_Class_Card
3.0	Lowest	A teacher “will” move to the next screen by clickingthe mouse over to the right.	UC_21	UC_Class_Primary_Action
3.0		A teacher “shall” generate the report card of students individually or in a group.	UC_22	UC_Report_Generation
3.0	Lowest	A teacher “will” delete report file information afterit has been generated.	UC_23	UC_Delete
3.0	Lowest	A teacher “will” sometimes rename a report fileafter it has been generated.	UC_24	UC_Rename
3.0	Lowest	A teacher “will” visualize a pie chart of how theirstudents are behaving and performing.	UC_25	UC_Performance
3.0		A teacher “will” create multiple course cardsaccording to the schedule.	UC_26	UC_Course_Card
3.0	Lowest	A teacher “will” move to the next screen by clickingthe mouse over to their left.	UC_26	UC_Course_Primary_Action
3.0	Lowest	A teacher “will” generate a report for their students that includes the student's name, roll number, all the marks they obtained in that course, and the GPA of those marks.	UC_27	UC_Report_Generation
3.0	Lowest	A teacher “will” delete the report file after it is generated.	UC_28	UC_Delete
3.0	Lowest	A teacher “will” rename the report file after it is generated and provide contents to the students.	UC_29	UC_Rename
3.0	Lowest	A teacher “will” visualize a pie chart of their student's performance. In this visualization, each slice of the pie is one particular metric.	UC_30	UC_Performance
3.0	Lowest	A teacher “will” change the limits for both the grades of individual students and the number of courses.	UC_31	UC_Edit_Thresholds
3.0	Lowest	A teacher “will” move to the next screen where all the selected students' records are available by clicking the mouse over to the right.	UC_32	UC_Dropdown_Primary_Action
3.0	Lowest	A teacher “will” generate a report for their students that include the student's name and roll number, as well as all the marks.	UC_33	UC_Report_Generation

		class. The report will also include the GPA of those marks.		
3.0	Lowest	A teacher “will” move to the expanded screen of the record field by clicking left on their mouse.	UC_34	UC_Student_Primary_Action
3.0	Lowest	A teacher “will” open their student records individually in the field.	UC_35	UC_Open
3.0	Lowest	A teacher “will” create a report of students based on their attendance, performance, and grades.	UC_36	UC_Create_Report
3.0	Lowest	A teacher “can” generate student reports by clicking left on the Generate button on the screen.	UC_37	UC_Generate_Report
3.0	Lowest	A teacher “will” reset the record field in case if a student is missing or absent for extended periods of time.	UC_38	UC_Reset
3.0	Lowest	A teacher “can” move a student's field up or down by simply dragging it. This is helpful if your class list is in alphabetical order, and you want to rearrange students' fields for different activities.	UC_39	UC_Drag_Drop
3.0	Lowest	A teacher “will” provide feedback on the application features or any bugs.	UC_40	UC_Feedback
3.0	Lowest	A teacher “will” visualize our latest updates and will see the terms and conditions, privacy policy statement, and company location	UC_41	UC_About

### 3.8 Requirements Trace-ability Matrix

Sr#	Para	System Specification Text	Build	Use Case Name	Category
1	1.0	The teacher “will” sign up with their Google account to get verified to use Google Drive and Google sheets.	B1	UC_Authentication	Services
2	1.0	A teacher “will” customize their experience by configuring the settings.	B1	UC_Settings	Services
3	1.0	The Navigation bar “will” be displayed on the leftside of the screen.	B1	UC_Navigation_Bar	Services
4	1.0	A teacher “will” open the default page where all the performance of classes and courses are available in the pie charts.	B1	UC_Home	Services
5	1.0	A customer “shall” log in to the system and can change his password.	B1	UC_Main	Services

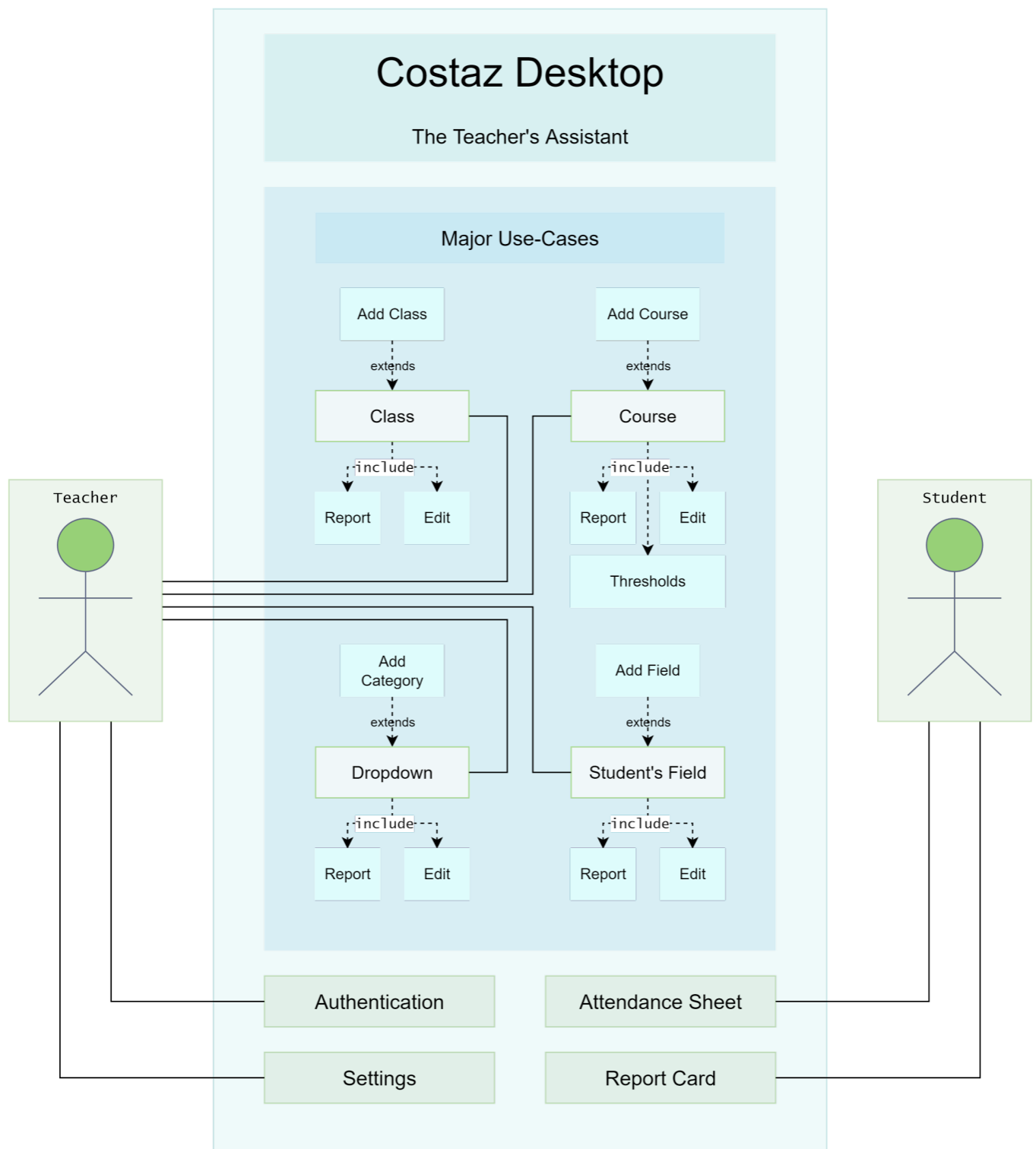
PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

6	1.0	A teacher shall add the class of students on the homepage of the application.	B1	UC_Class_Add	Services
7	1.0	A teacher “will” create multiple class cards according to the schedule.	B1	UC_Class_Card	Services
8	1.0	A teacher “will” move to the next screen by clicking the mouse over to the right.	B1	UC_Class_Primary_Action	Services
9	1.0	A teacher “will” visualize an additional context menu on the screen with just a right click.	B1	UC_Context_Menu	Services
10	2.0	A teacher “shall” generate the report card of students individually or in a group.	B1	UC_Report_Generation	Services
11	2.0	A teacher “will” delete report file information after it has been generated.	B1	UC_Delete	Services
12	1.0	A teacher “will” sometimes rename a report file after it has been generated.	B1	UC_Rename	Services
13	1.0	A teacher “will” visualize a pie chart of how their students are behaving and performing.	B1	UC_Performance	Services
14	1.0	A teacher “shall” add the course to the top-level menu of the application.	B1	UC_Course_Add	Services
15	1.0	A teacher “will” set limits for both the grades of individual students and the number of courses	B1	UC_Tresholds	Services
16	1.0	A teacher “will” create multiple course cards according to the schedule.	B1	UC_Course_Card	Services
17	1.0	A teacher “will” move to the next screen by clicking the mouse over to their left.	B1	UC_Course_Primary_Action	Services
18	1.0	A teacher would be able to display a context menu on the screen by just right-clicking.	B1	UC_Context_Menu	Services
19	1.0	A teacher “will” generate a report for their students that includes the student's name, roll number, all the marks they obtained in that course, and the GPA of those marks.	B1	UC_Report_Generation	Services
20	1.0	A teacher “will” delete the report file after it is generated.	B1	UC_Delete	Services
21	2.0	A teacher “will” rename the report file after it is generated and provide contents to the students.	B1	UC_Rename	Services
22	2.0	A teacher “will” visualize a pie chart of their student's performance. In this visualization, each slice of the pie is one particular metric.	B1	UC_Performance	Services

23	1.0	A teacher “will” change the limits for both the grades of individual students and the number of courses.	B1	UC_Edit_Thresholds	Services
24	1.0	A teacher “will” select the class based on the available sections or shifts (mor/eve).	B1	UC_Dropdown_Partition	Services
25	1.0	A teacher “will” move to the next screen, where all the selected students' records are available by clicking the mouse over to the right.	B1	UC_Dropdown_Primary_Action	Services
26	1.0	A teacher “can” create a context menu with open and delete features by right-clicking.	B1	UC_Context_Menu	Services
26	1.0	A teacher “will” generate a report for their students that includes the student's name and roll number, as well as all the marks they obtain in that class. The report will also include the GPA of those marks.	B1	UC_Report_Generation	Services
27	1.0	A teacher “will” delete the report file once it's generated.	B1	UC_Delete	Services
28	1.0	A teacher “will” rename the generated report and share it with the students.	B1	UC_Rename	Services
29	1.0	A teacher “will” visualize the data they receive from their records, so they can more easily spot the students who are performing exceptionally well, those who are doing OK, and those who need help.	B1	UC_Performance	Services
30	1.0	On click, the teacher “will” visualize the student's name, roll no, rank and CGPA.	B1	UC_Student_Field	Services
31	2.0	A teacher “will” move to the expanded screen of the record field by clicking left on their mouse.	B1	UC_Student_Primary_Action	Services
32	2.0	The teacher “can” visualize the attendance, and performance, create and generate reports, and reset and delete the field of the student by just clicking right on the mouse.	B1	UC_Context_Menu	Services
33	1.0	A teacher “will” open their student records individually in the field.	B1	UC_Open	Services
34	1.0	A teacher “will” create a report of students based on their attendance, performance, and grades.	B1	UC_Create_Report	Services
35	1.0	A teacher “can” generate student reports by clicking left on the Generate button on the screen.	B1	UC_Generate_Report	Services
36	1.0	A teacher “can” visualize how their students perform in assignments, presentations, and quizzes with pie charts – use them to quickly understand their strengths and weaknesses.	B1	UC_Performance	Services

37	1.0	A teacher “will” reset the record field in case if a student is missing or absent for extended periods of time.	B1	UC_Reset	Services
38	1.0	A teacher “can” move a student's field up or down by simply dragging it. This is helpful if your class list is in alphabetical order, and you want to rearrange students' fields for different activities.	B1	UC_Drag_Drop	Services
39	1.0	The teacher “will” either mark the attendance individually or by selecting all.	B1	UC_Attendance_View	Services
40	1.0	Each button will be colored differently. By default, students will be present, and the teacher just needs to mark the missing student’s attendance.	B1	UC_Attendance_Button_PAL	Services
41	1.0	A teacher “will” visualize the student report card, which will show a graphical representation of all the student's performances in the pie charts. The charts are made from attendance records, class activities and marks received	B1	UC_Report_Card_View	Services
42	2.0	A teacher “will” set limits for both the grades of individual students and the number of courses, on the fly.	B1	UC_Add_OTF	Services
43	2.0	A teacher “can” save the changes to their report card by pressing buttons below the generated report card.	B1	UC_Generate_Report	Services
44	1.0	A teacher “will” pin students who are doing better and put information in bookmarks for students who need more attention.	B1	UC_Bookmarks	Services
45	1.0	A teacher “will” usually put all the files in GoogleDrive.	B1	UC_Files	Services
46	1.0	A teacher “will” change the font size, theme, notification preferences, and more in the setting.	B1	UC_Settings	Services
47	1.0	A teacher “will” provide feedback on the application features or any bugs.	B1	UC_Feedback	Services
48	1.0	A teacher “will” get all the old deleted material in the trash.	B1	UC_Trash	Services
49	1.0	A teacher “will” visualize our latest updates and will see the terms and conditions, privacy policy statement, and company location	B1	UC_About	Services

### 3.9 High-Level Use case Diagram





## 4 USE CASES

The team's goal is to create a software application that will make life easier for teachers. Teachers can use it to log their student records and attendance, as well as their assignments. With the increasing number of students in schools, teachers are often found busy managing the attendance of their students. A teacher's assistant application is being developed, which will help teachers to manage the attendance sheet generation and report card generation. This application can be used by teachers to perform their daily duties more efficiently and effectively as they can now also be used for storing data in Google Drive, so that they can be accessed anywhere at any time. The name of this application is Costaz. The word Costaz is derived from two words: “co-” which is used for the assistant agent, and “ostaz” (استاذ) which translates to “Teacher” in Arabic. Costaz Desktop is a teacher- friendly desktop application developed to manage academic records of their students. It allows for attendance sheets generation in Excel form, report card generation for individual students, and generation of marks sheets for all students, as well as for a specific course.

### 4.1 Use-Case Description

#### □ Teacher

It is a class because it satisfies all 6 characteristics.

##### Behavior:

1. Signup
2. Add Class
3. Add Course
4. Mark Attendance
5. Edit Records
6. Generate Reports
7. Share Reports

#### □ Google API

It is a class because it satisfies all 6 characteristics.

##### Behavior:

1. Authenticate User
2. Manage Google Sheets
3. Manage Google Drive

## Use Cases:

<b>Use Case #1.1</b>	UC_Authentication	
<b>Description</b>	The actor will sign up with their Google account to get verified to use Google Drive and Google Sheets.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Given Email must be valid.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Google account must be valid.</li> <li>Successfully registered in application with Google account.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Invalid credentials.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	When Actor clicks Signup button.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor will open the application.
	2	The credentials will be taken in a popup screen only when the user entered first time.
	3	Application account for the Actor will be created.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
		The credentials can be stored in an external database (Firebase).
<b>Variation</b>		<b>Branching Action</b>
	1	Actor can add a custom email in the given mail field if the Gmail account is not login into the browser or device.
	2	Feature of “Forgot Password” can be provided.
	3	Separate features for signup and sign-in can be provided.
<b>Exception</b>	1	When actor puts invalid credentials.
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.2</b>	UC_Settings	
<b>Description</b>	The actor will customize their experience by configuring the settings.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>The actor must be logged in.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Actor can successfully configure the file and exported in Google drive</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>External storage is write-protected.</li> <li>Due to unavailability of internet configuration file could not e uploadedto Google drive.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Configure file will automatically popup after sign in.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	When actor signup with an account, a theme selection window will be triggered on-screen.
	2	By default, the theme will be selected by light.
	3	Actor can select any between dark and light theme.
	4	Click on the change button and the theme will be changedeverywhere, accordingly.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
	1	We can swap setting window from use case 1.2.
<b>Exception</b>		External storage is write-protected.
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3</b>	UC_Navigation_Bar	
<b>Description</b>	The Navigation bar will be displayed on the left side of the screen	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>User must be logged in.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Main Menu items will be displayed upon clicking on navigation bar button.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Memory or processor is too much busy, so the menu bar won't be able to animate, hence shadowing the underlying items.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Click on the navigation bar icon.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor must sign-in.
	2	Settings will be configured.
	3	Homepage will appear and menu icon will be on the top left corner.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
		More items can be added in the navigation
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.1</b>	UC_Home	
<b>Description</b>	The Actor will open the default page where all the performance of classes and courses will be available in the pie charts.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Actor must be created class and course to see the performance overview of any class/course.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Main Menu features successfully pop up on the screen by clicking on the menu bar.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Failed to load main menu features due to slow internet or providing wrong information in the login section.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Click on the login/signup button.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor must authenticate by Google.
	2	After successful authentication, actor redirect on the home page.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.2</b>	UC_Main	
<b>Description</b>	The Actor will open the default page where all the performance of classes and courses will be available in the pie charts.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Actor must be authenticated by Google Services.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Successfully land on the main menu from navigation bar.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Failed to load main menu features due to slow internet</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Click on the main menu button in the navigation bar.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor must authenticate by Google.
	2	Navigation bar will open in the home page.
	3	Click on the Menu Icon in navigation bar.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.2.1</b>	UC_Class_Add	
<b>Description</b>	The Actor will add the class of students on the homepage of the application.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Actor should be Authenticate.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Class Create when Actor click</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Google Drive Storage full.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	When the teacher clicks on the Class Add in the application.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor will authenticate by Google API.
	2	After that actor move to the Main Menu from the Navigation Bar
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.2.2</b>	UC_Class	
<b>Description</b>	The Actor will create multiple class cards according to the schedule.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Class should be instantiated.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>When Class will be added in the List.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Google Drive Storage Full.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	When Class will be added by UC_Class_Add	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor will authenticate by Google API.
	2	After that actor move to the Main Menu from the Navigation Bar
	3	Class Should be instantiated.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
		<ul style="list-style-type: none"> <li>Class Performance will be display on Screen.</li> <li>Total Number of students will display on screen.</li> </ul>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		



PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

<b>Use Case #1.3.2.3</b>	UC_Class_Report	
<b>Description</b>	The Actor will generate the report card of students in a group.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>• Class must have Entities.</li> <li>• Entitles must have their Records.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• Report will create successfully.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>• Class does not have Entities.</li> <li>• Entitles does not have Records.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Trigger by Actor via Context Menu.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on Generate Report in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
		<ul style="list-style-type: none"> <li>• Report can be generated for Course Only.</li> <li>• Report can be generated for individual student only.</li> </ul>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

<b>Use Case #1.3.2.4</b>	UC_Class_Edit	
<b>Description</b>	The Actor will delete a record file after it has been generated and sometimes rename a report file.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Class must be Exist.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Class Successfully Edit via Context Menu.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Class will disconnect by Internet.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	It will trigger via Context Menu by right click.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on Edit in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

<b>Use Case #1.3.2.1.1</b>	UC_Course_Add	
<b>Description</b>	The Actor will add the course for students in the Class Section.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Actor should be Authenticate.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Course Create when Actor click on the Course Add</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Google Drive Storage full.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	When the teacher clicks on the Class Add in the application.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor will authenticate by Google API.
	2	After that actor move to the Main Menu from the Navigation Bar.
	3	Create a Class by UC_Class_Add.
	4	Add a Course in the Created Class.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.2.1.2</b>	UC_Course	
<b>Description</b>	The Actor will be able to create multiple course cards according to the schedule.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Course should be instantiated.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>When Course will be added in the List.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Google Drive Storage Full.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	When Class will be added by UC_Course_Add	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor will authenticate by Google API.
	2	After that actor move to the Main Menu from the Navigation Bar
	3	Actor will create the class by UC_Class_Add.
	4	Course will be added by UC_Course_Add.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
	1	<ul style="list-style-type: none"> <li>Course Performance will be display on Screen.</li> <li>Total Number of students will display on screen.</li> </ul>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.2.1.3</b>	UC_Course_Report	
<b>Description</b>	The actor will generate the report card of students in a group.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Course must have Entities.</li> <li>Entitles must have their Records.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Report will create successfully.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Course does not have Entities.</li> <li>Entitles does not have Records.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Trigger by Actor via Context Menu.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on Generate Report in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
		<ul style="list-style-type: none"> <li>Report can be generated for Class Only.</li> <li>Report can be generated for individual student only.</li> </ul>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

<b>Use Case #1.3.2.1.4</b>	UC_Course_Edit	
<b>Description</b>	The actor will delete a record file after it has been generated, and sometimes rename a report file after it has been generated.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Course must be created in the Class.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Course Successfully Edit via Context Menu.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Course will disconnect by Internet.</li> <li>Attached Class with Course Discard by Actor.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	It will trigger via Context Menu by right click.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on Edit in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.2.1.5</b>	UC_Thresholds	
<b>Description</b>	The actor will change the limits for both the grades of individual students and the number of courses, and also set limits for the course, like total marks, marks for assignments, quizzes etc.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>• Parent Class must exist.</li> <li>• Student must enroll in the Course.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• Threshold successfully saved.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>• Limitation Conflict in the thresholds.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	<ul style="list-style-type: none"> <li>• Thresholds will trigger via context menu by right click on course.</li> </ul>	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on thresholds in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
	1	<ul style="list-style-type: none"> <li>• Maximum Student Mark will set during records. (Edit)</li> </ul>
<b>Exception</b>		<ul style="list-style-type: none"> <li>• Thresholds page will available for more Fields.</li> </ul>
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

<b>Use Case #1.3.2.1.6</b>	UC_Dropdown	
<b>Description</b>	The Actor will select the class based on the available sections or shifts (mor/eve).	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Actor should be in the student field.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Successfully create via context menu in the student field.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Course will terminate in the class.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	When the Actor click on the context menu.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on dropdown in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		



<b>Use Case #1.3.2.1.7</b>	UC_Dropdown_Report	
<b>Description</b>	The Actor will generate a report for their students that include the student's name and roll number, as well as all the marks they obtain in that class. The report will also include the GPA of those marks.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>• Dropdown must have Entities.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• Report will create successfully.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>• Dropdown does not have Entities.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Trigger by Actor via Context Menu.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on Generate Report in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

<b>Use Case #1.3.2.1.8</b>	UC_Dropdown_Edit	
<b>Description</b>	The Actor will delete the report file once it's generated and rename the generated report and share it with the students.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Course must be created in the Class.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Course Successfully Edit via Context Menu.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Disconnect by Internet.</li> <li>Attached Class with Course Discard by Actor.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Actor via Context Menu.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on Edit in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.2.1.1.1</b>	UC_Add_Field	
<b>Description</b>	The Actor will add the multiple fields according to their entities records.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Course should be instantiated.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Successfully field added by actor.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Field related class discarded by actor.</li> <li>Field related course discarded by the actor.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Actor on add while being in the course section.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor will create the class.
	2	Course will add in the created class.
	3	Click on the plus button for create field.
	4	Enter the details of every student individually.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
		<ul style="list-style-type: none"> <li>Performance will also display on screen.</li> </ul>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.2.1.1.2</b>	UC_Student_Feild	
<b>Description</b>	The Actor will visualize the student's name, roll no, rank and CGPA.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>It is evaluated by add field.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>When Course will add in the List.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Lost Student Records.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>		
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Create the Class by Add Class.
	2	Add the Specific Course related to the created class.
	3	Student field appeared on the screen.
	4	Enter the student detail individually in fields.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

<b>Use Case #1.3.2.1.1.3</b>	UC_Student_Field_Report	
<b>Description</b>	Actor will create a report of students based on their attendance, performance, and grades and generate student reports by clicking left on the Generate button on the screen.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Student Field must have records.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Report will create successfully.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Dropdown does not have Entities.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Trigger by Actor via Context Menu.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on Generate Report in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

<b>Use Case #1.3.2.1.1.4</b>	UC_Student_Field_Edit	
<b>Description</b>	The Actor will visualize the attendance and performance, create and generate reports, and reset and delete the field of the student by just clicking right on the mouse.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Record must be created in the fields.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Field Successfully Edit via Context Menu.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Student Field will disconnect by Internet.</li> <li>Attached Class with Course Discard by Actor.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Trigger by Actor via Context Menu.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click.
	2	Click on Edit in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

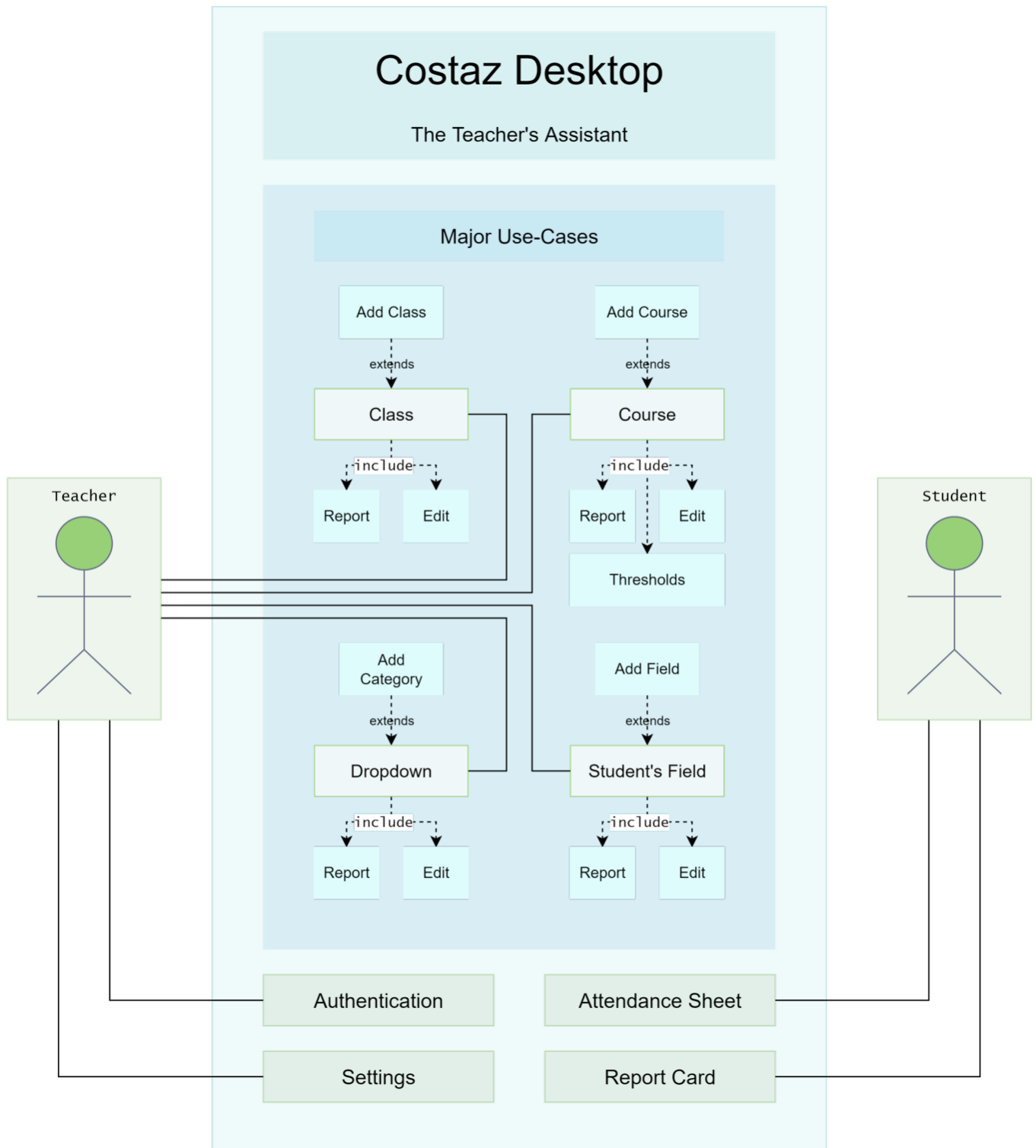
<b>Use Case #1.3.2.1.1.6</b>	UC_Attendance_View	
<b>Description</b>	The Actor will either mark the attendance individually or by selecting all	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>Student Entitles must be exist.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Actor successfully marks the Attendance.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Internet Disconnected.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Primary Actor on Student Field. (click)	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Actor click on the student field in the Google sheet.
	2	Dropdown Menu will appear on Screen.
	3	Attendance will mark each student individually.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
		<ul style="list-style-type: none"> <li>Performance of each student show on the screen.</li> </ul>
<b>Variation</b>		<b>Branching Action</b>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

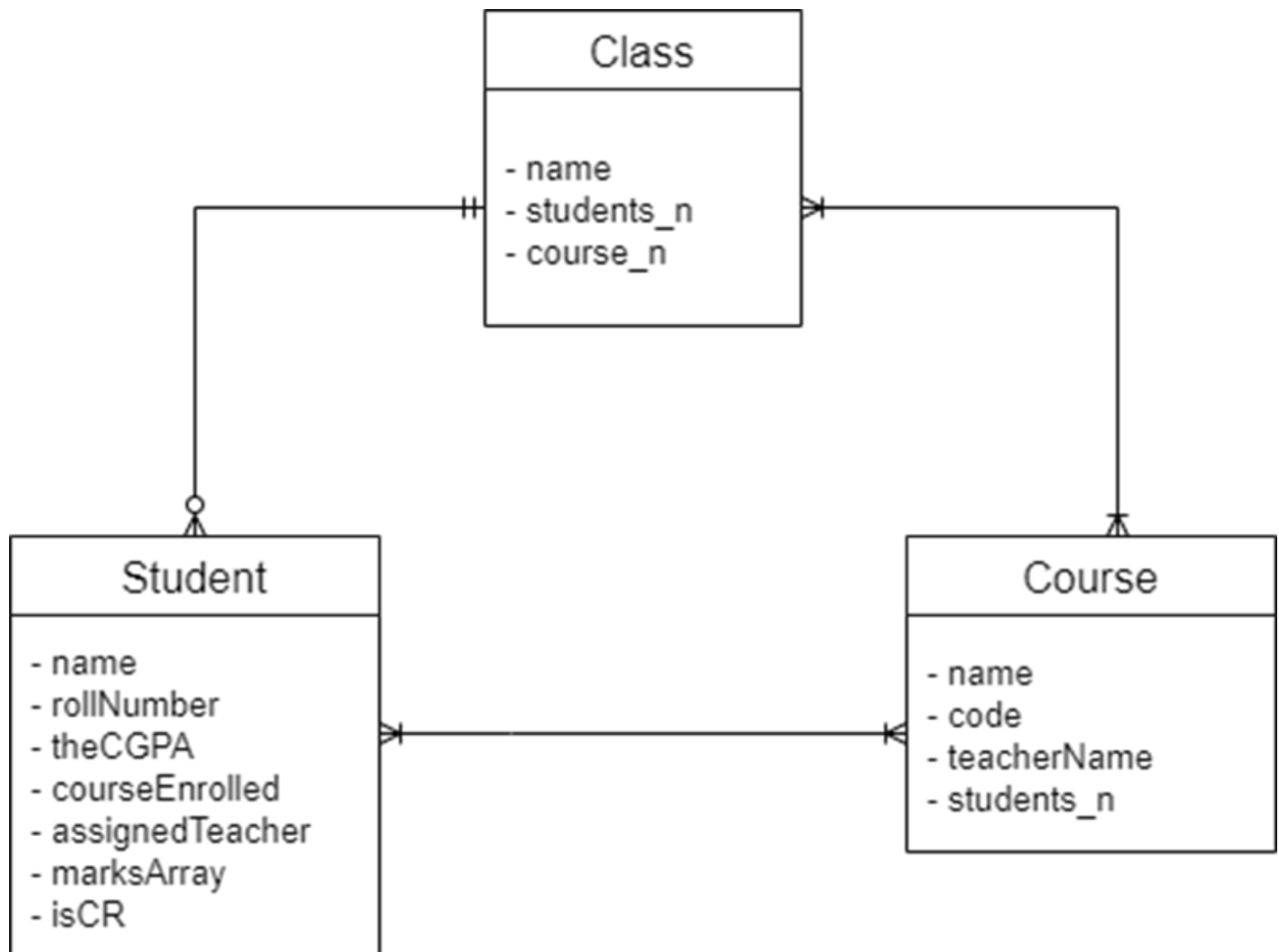
<b>Use Case #1.3.2.1.1.7</b>	UC_Report_Card_View	
<b>Description</b>	The Actor will visualize the student's report card, which will show a graphical representation of all the student's performances in the pie charts. The charts are made from attendance records, class activities and marks received.	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>• Parent entity of record card must exist.</li> <li>• Student record available.</li> <li>• Internet Available.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• Report Card Generated Successfully.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>• Student Record does not exist.</li> </ul>	
<b>Actors</b>	Teacher	
<b>Trigger</b>	Trigger by Actor via Context Menu in the student field.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Context Menu display by Right click in the student field.
	2	Click on Generate Report Card in Context Menu.
<b>Extension</b>	<b>Step</b>	<b>Branching Action</b>
<b>Variation</b>		<b>Branching Action</b>
		<ul style="list-style-type: none"> <li>• Actor can create specific report according to Class.</li> <li>• Actor can generate specific report according to course.</li> </ul>
<b>Exception</b>		
<b>Other Information</b>		
<b>Open Issues</b>		
<b>Due Date</b>		



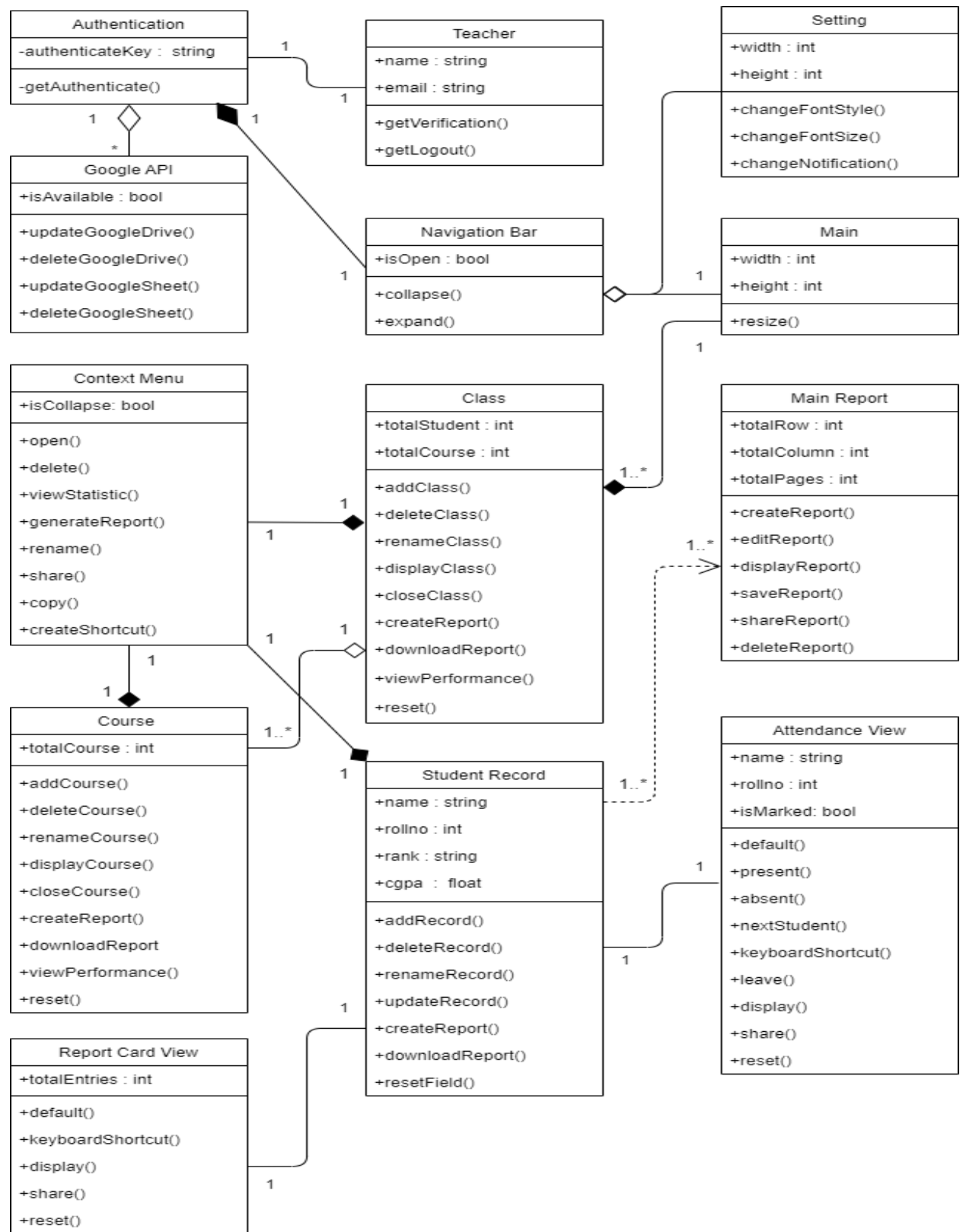
## 4.2 High-Level Use case Diagram



## 4.2 Data Model

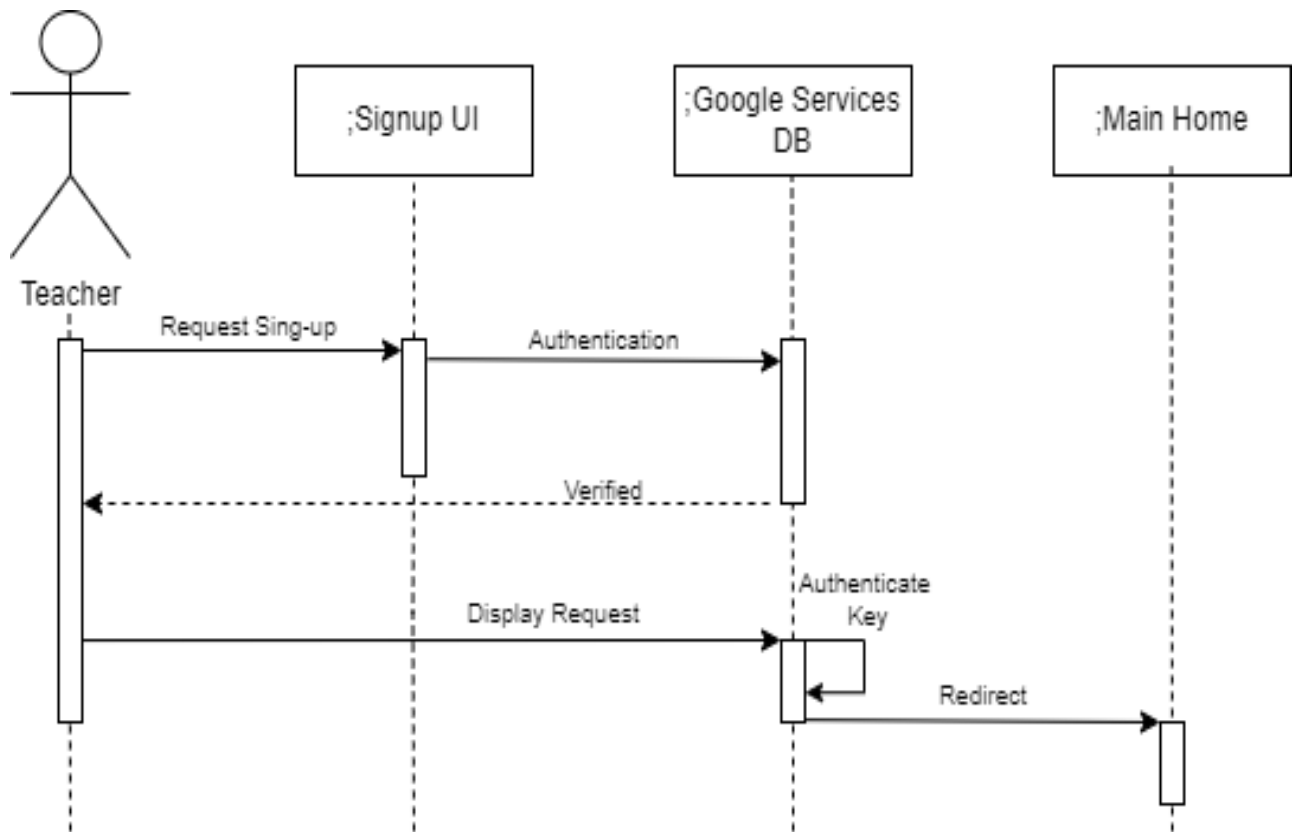


## 5.0 Domain Model

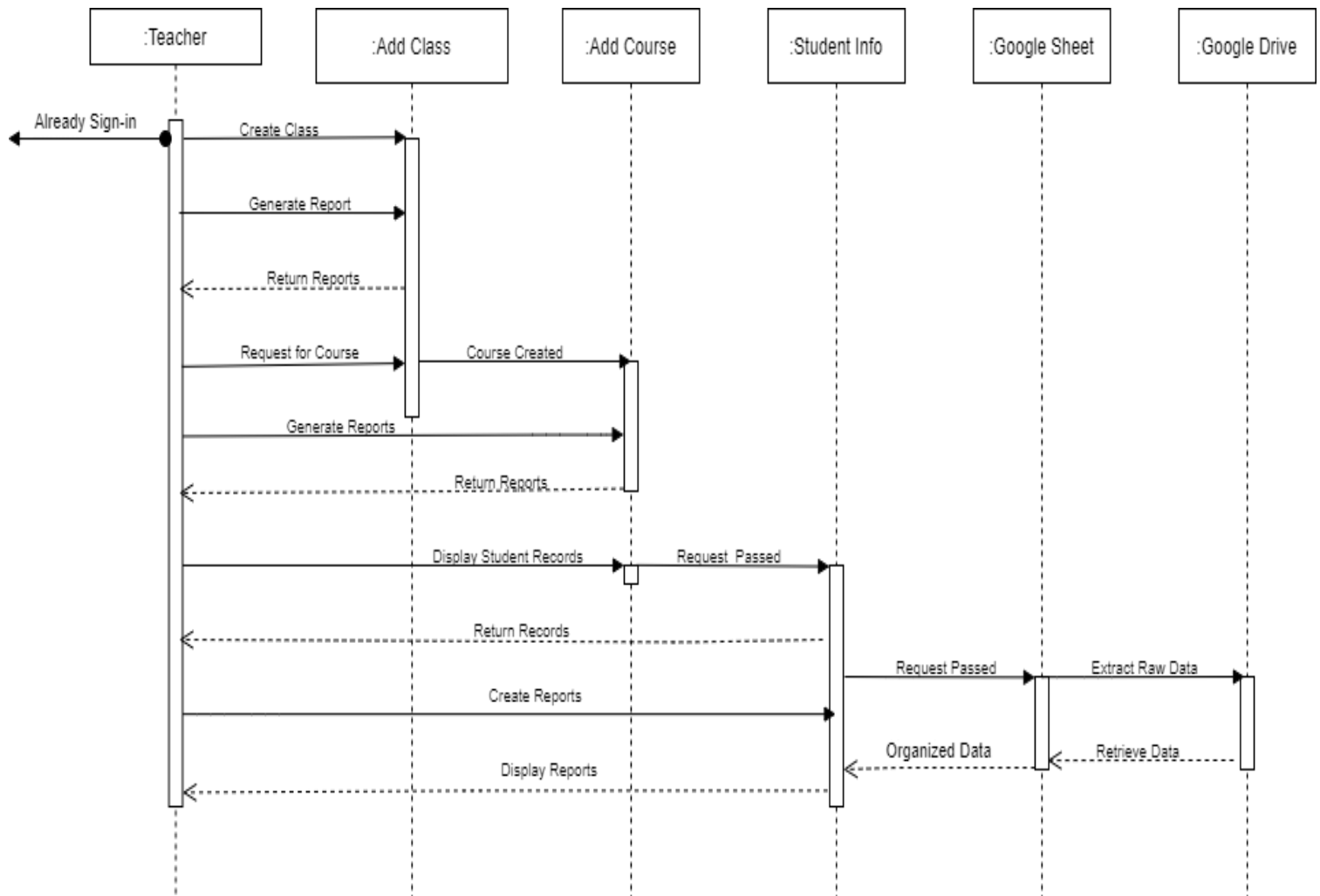


## 6.0 Sequence Diagram

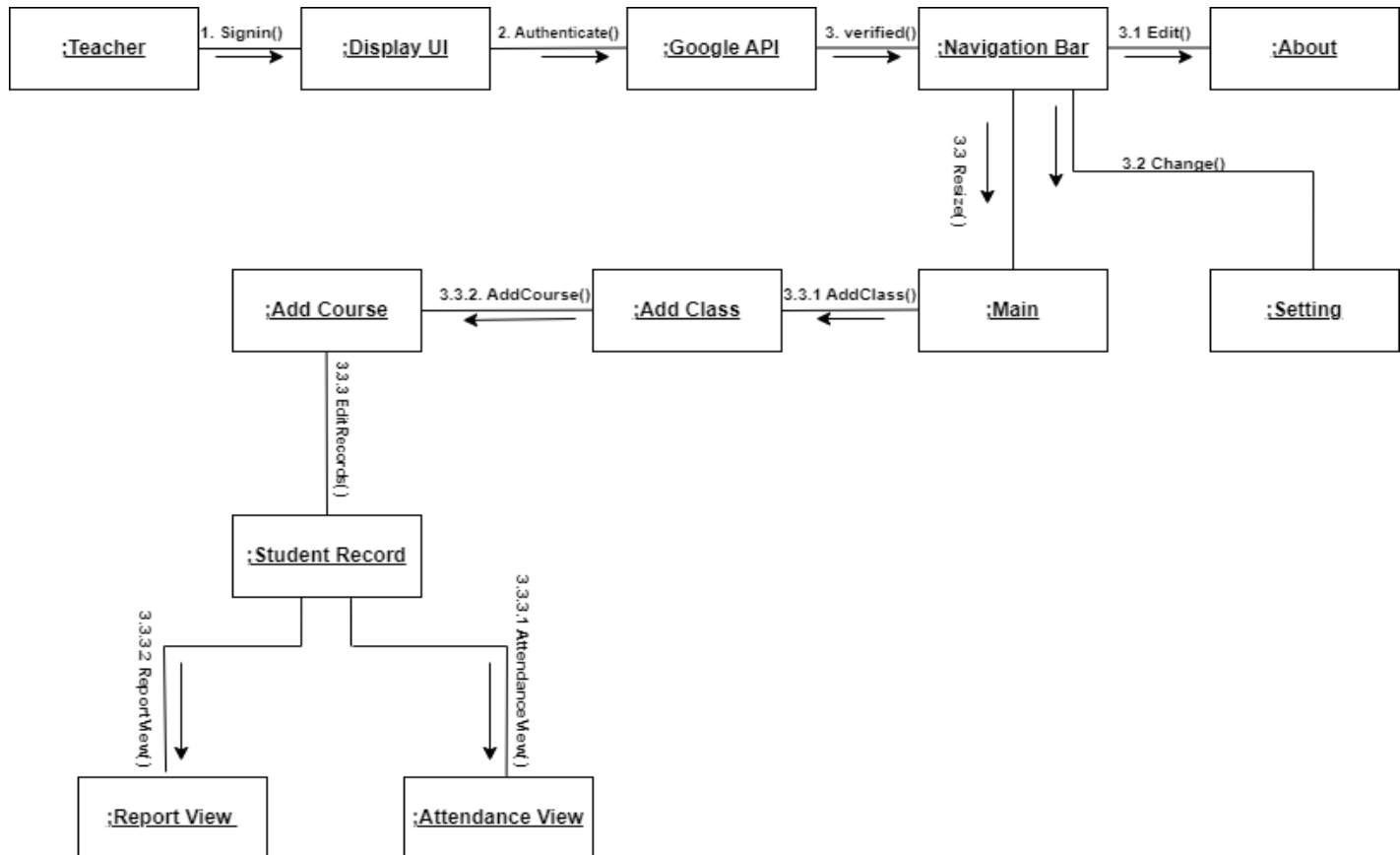
- **Signup**



## • Teacher Assistance



## 7.0 Collaboration Diagram



## 8.0 Operation Contracts

**Name:** Authenticate()

**Responsibilities:** Allow users to sign-up with their Google account to get verified and use Google Drive and Google sheets.

**Cross References:** UC\_Authentication.

**Exceptions:** Invalid credentials.

**Preconditions:**

- o Given email must be valid.

**Post conditions:** The user will properly register in the application and land on the home screen after successful authentication.

**Name:** Resize()

**Responsibilities:** Allow the user to open the default page where all the performance

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

of classes and courses will be available in the pie charts.

**Cross References:** Use Case: UC\_Main

**Exceptions:** Failed to load main menu features due to slow internet connection.

**Preconditions:**

- o User must be authenticated by Google Services.

**Post conditions:** User can easily land on the main menu from navigation bar.

**Name:** Add\_Class()

**Responsibilities:** Allow the user to create multiple classes according to their schedule.

**Cross References:** Use Case: UC\_Class

**Exceptions:** Authentication failed, or Google Drive storage is full for more classes.

**Preconditions:**

- o Class should be instantiated.

**Post conditions:** Teacher can create multiple classes by right clicking the Add Class button on the class card.

**Name:** Add\_Course()

**Responsibilities:** Allow the teacher to create multiple classes according to their schedule.

**Cross References:** Use Case: UC\_Course

**Exceptions:** Google Drive storage full during the course creation.

**Preconditions:**

- o Course should be instantiated.

**Post conditions:** Teacher can easily create multiple courses by right clicking or pressing on the add button on the course card.

**Name:** Edit\_Records()

**Responsibilities:** Allow the teacher to edit and visualize student name, roll no, rank and CGPA

**Cross References:** Use Case: UC\_Student\_Field

**Exceptions:** Student records deleted or lost from the teacher.

**Preconditions:**

- o Course should be instantiated.

**Post conditions:** Teacher can easily add field and add details of students in the fields.

**Name:** Attendance\_View()

**Responsibilities:** Allow the teacher to mark the attendance individually or by

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

selecting all students.

**Cross References:** Use Case: UC\_Attendance\_View

**Exceptions:** Internet disconnect during the attendance session.

**Preconditions:**

- o Student entities must exist in the field.

**Post conditions:** Teacher marks the attendance successfully and performance and performance of each student will also show on the screen.

**Name:** Report\_View()

**Responsibilities:** Allow users to visualize student report cards which will show a clear graphical representation of various easy to read charts. These are made from attendance records, class activities and marks received.

**Cross References:** Use Case:

UC\_Report\_Card\_View

**Exceptions:** Student records unavailable in the field.

**Preconditions:**

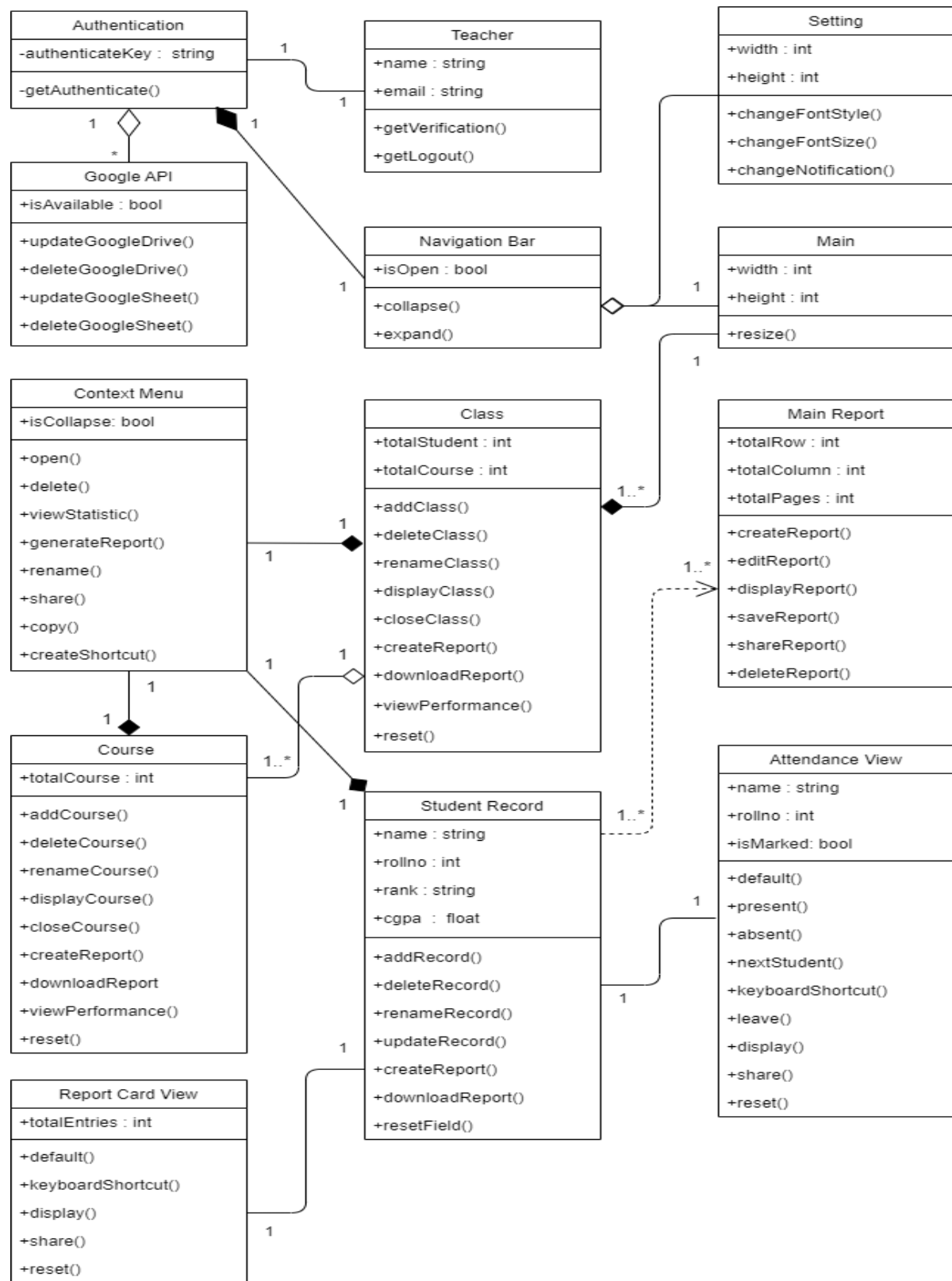
- o The parent entity of the record card must exist.
- o Student records must be available in the field.

**Post conditions:**

Report card successfully generated. It can be downloaded by the teacher or shared with students in the read-only mode.



## 9.0 Design Class Diagram



PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 10. Manual

### 10.1 Home Page

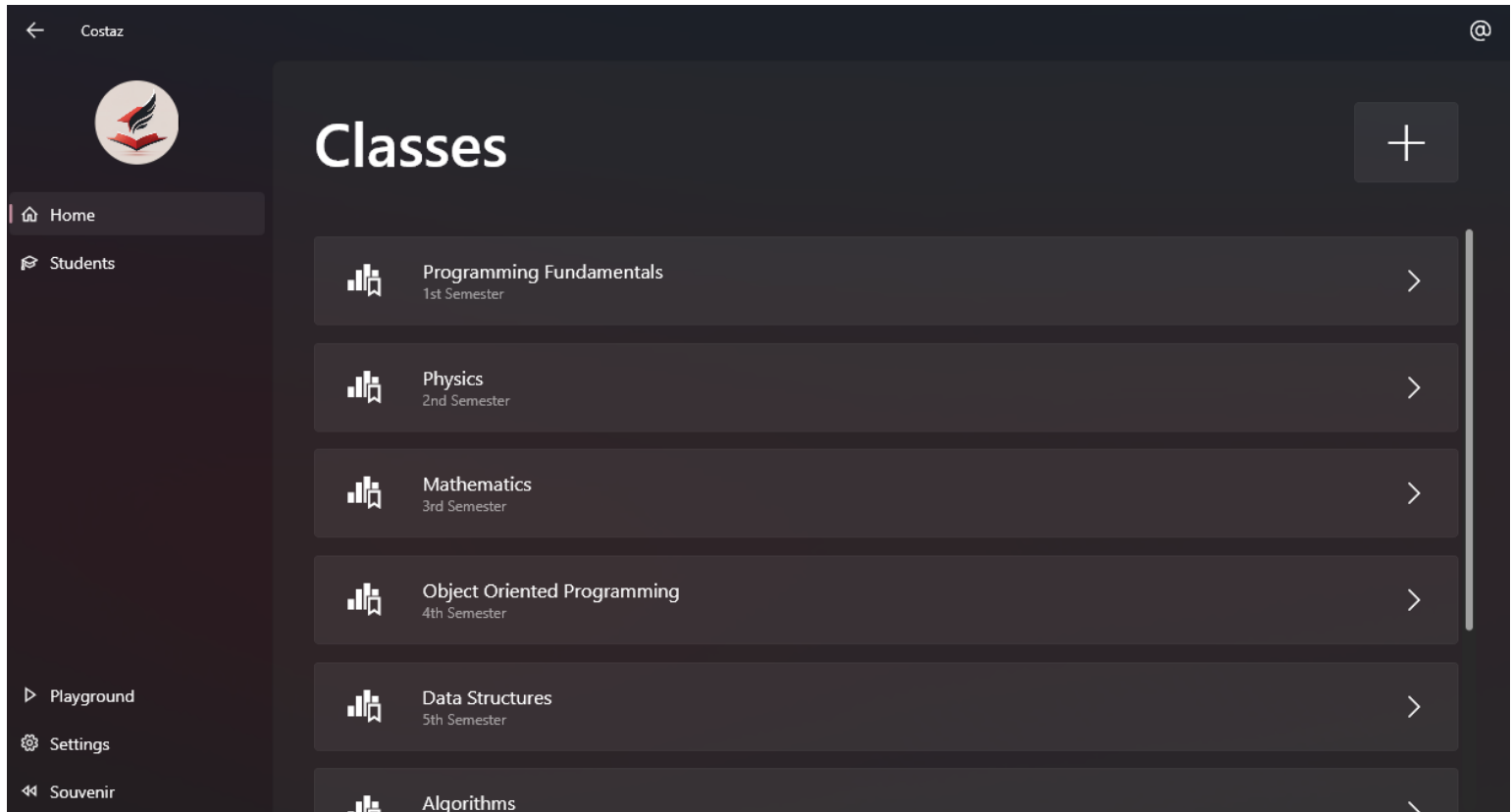


Figure 1 — in Dark Mode

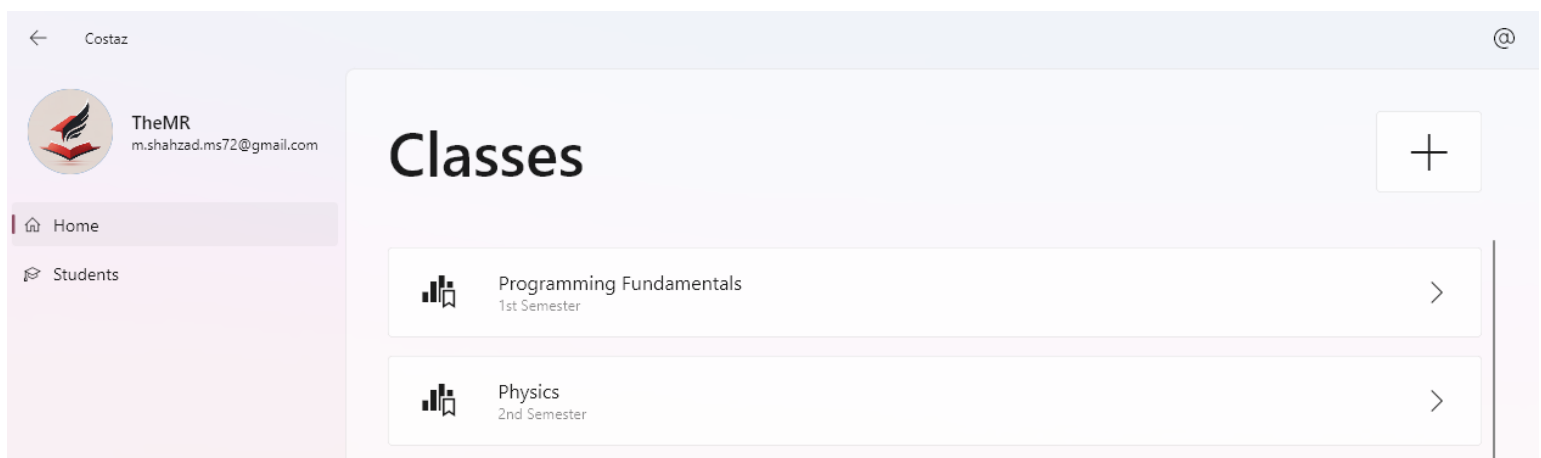


Figure 2 — in Light Mode

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 10.2 Create New Class

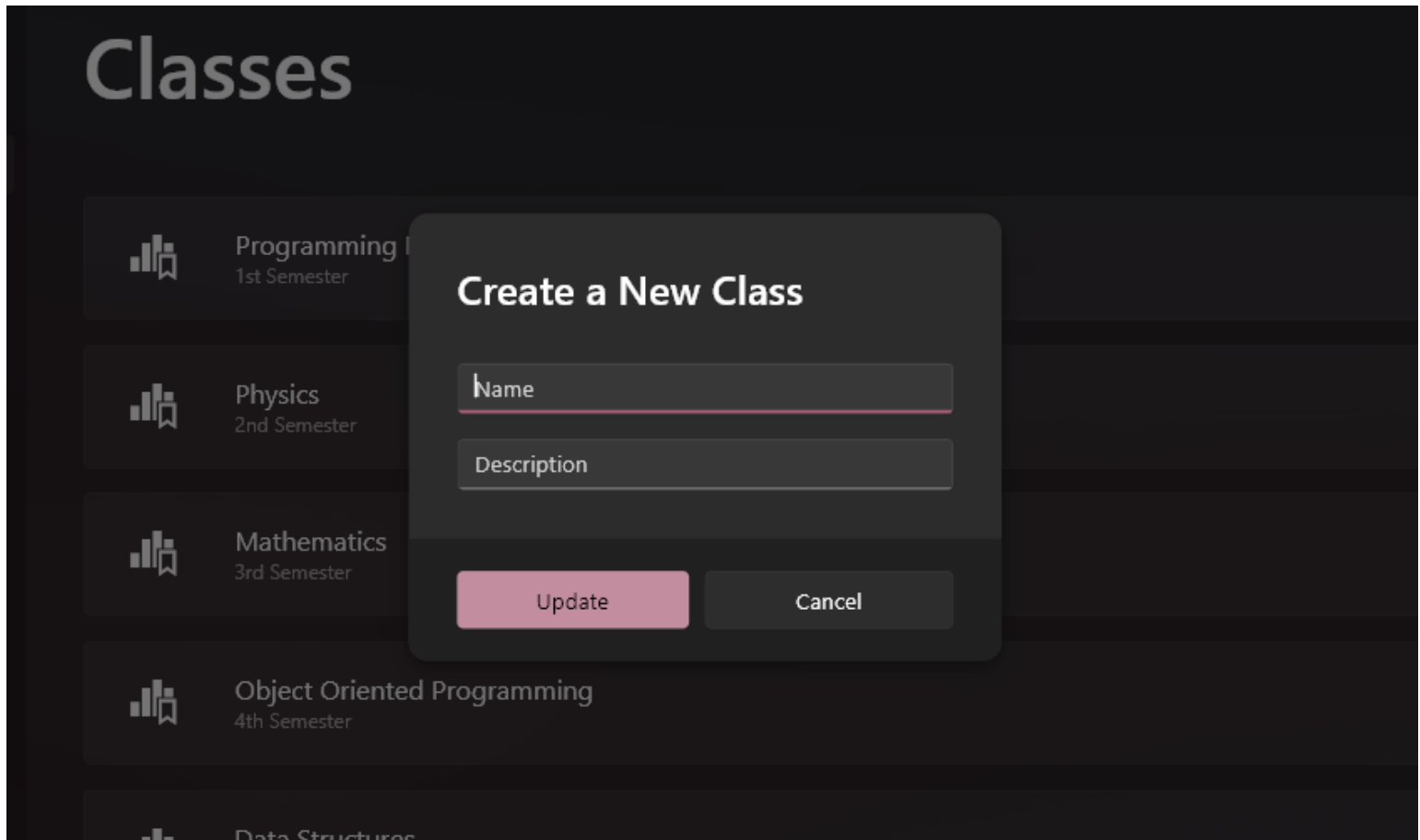


Figure 3 — Button is provided to be clicked on

### 10.2.1 Context Menu for Editing/Deleting Class

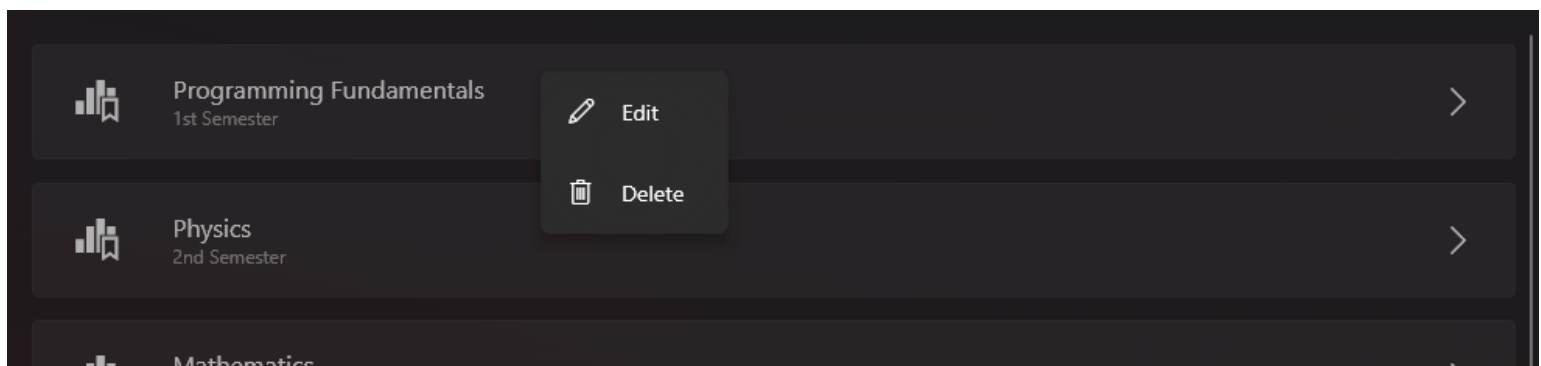


Figure 4 — Context Menu is activated by Secondary Tap (Right Click)

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

### 10.2.3 Editing Class

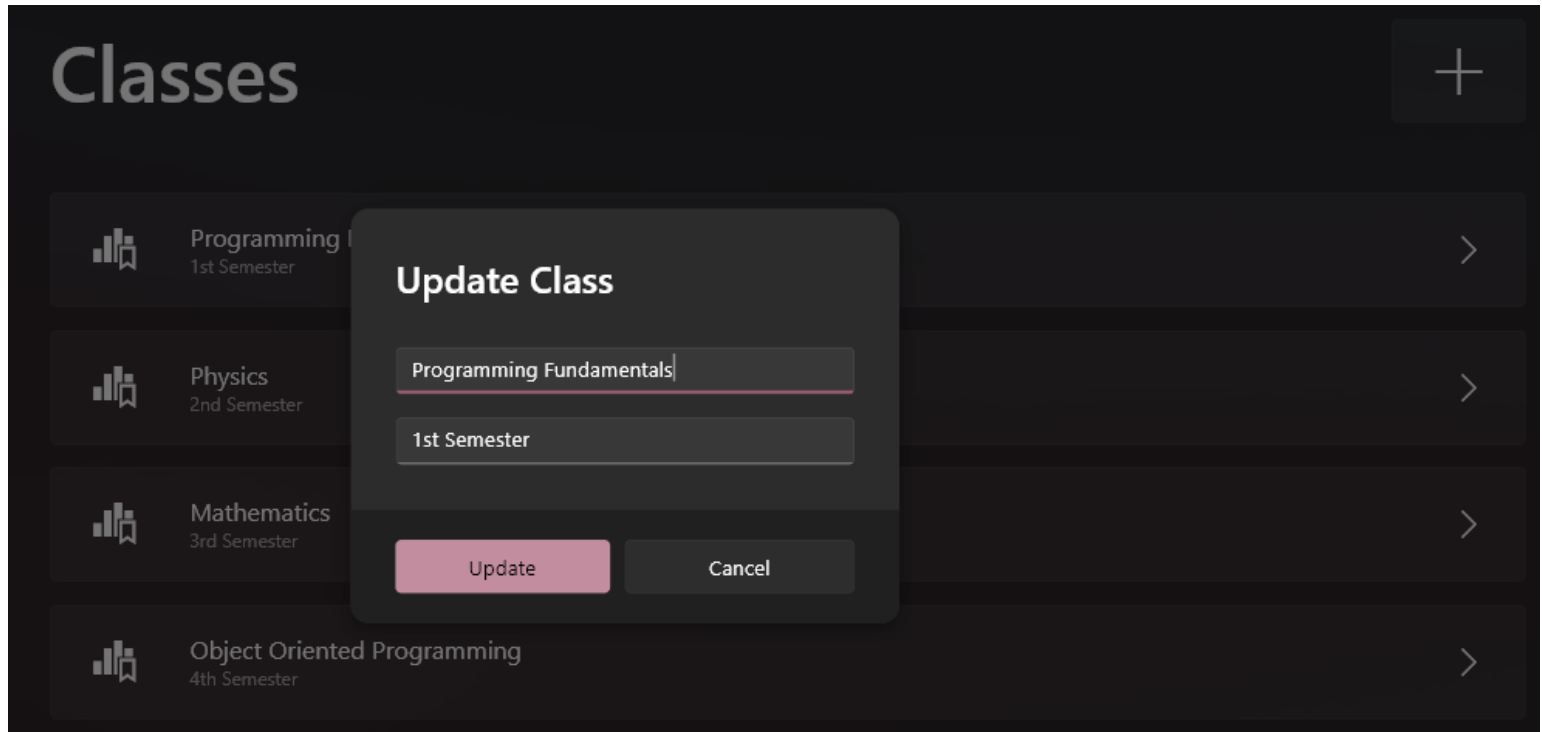
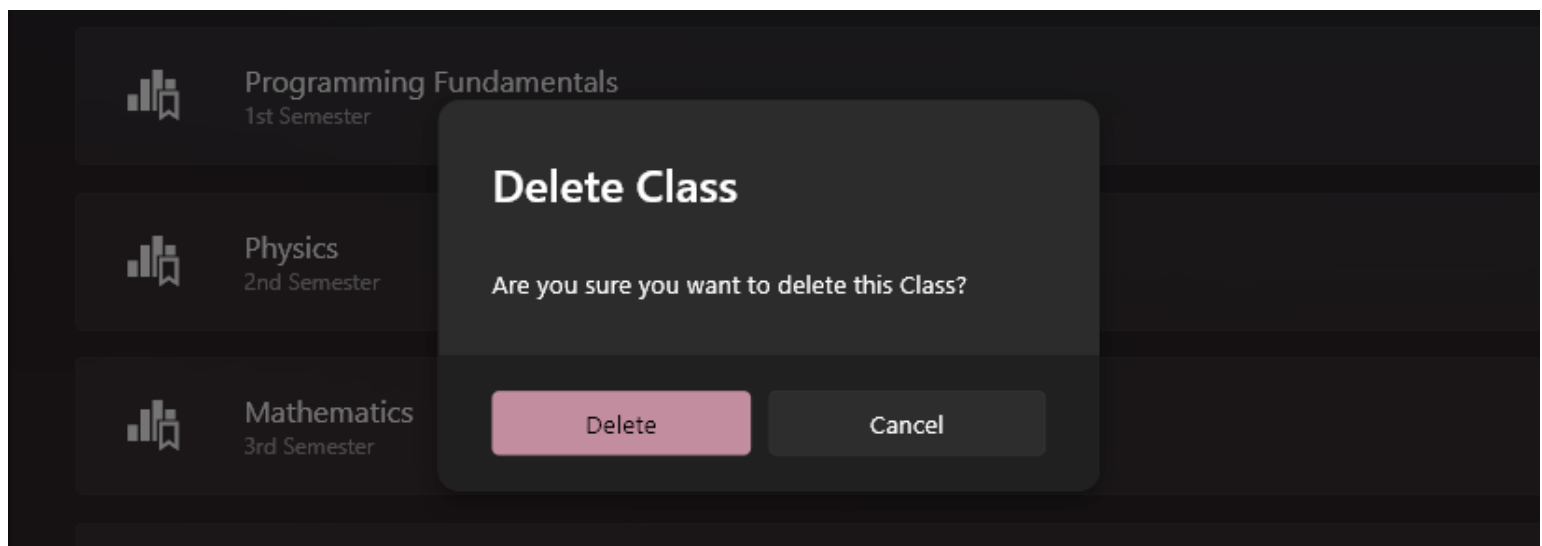


Figure 5 — by clicking the "Edit" on Context Menu

### 10.2.4 Deleting Class



PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 10.3 Student Page

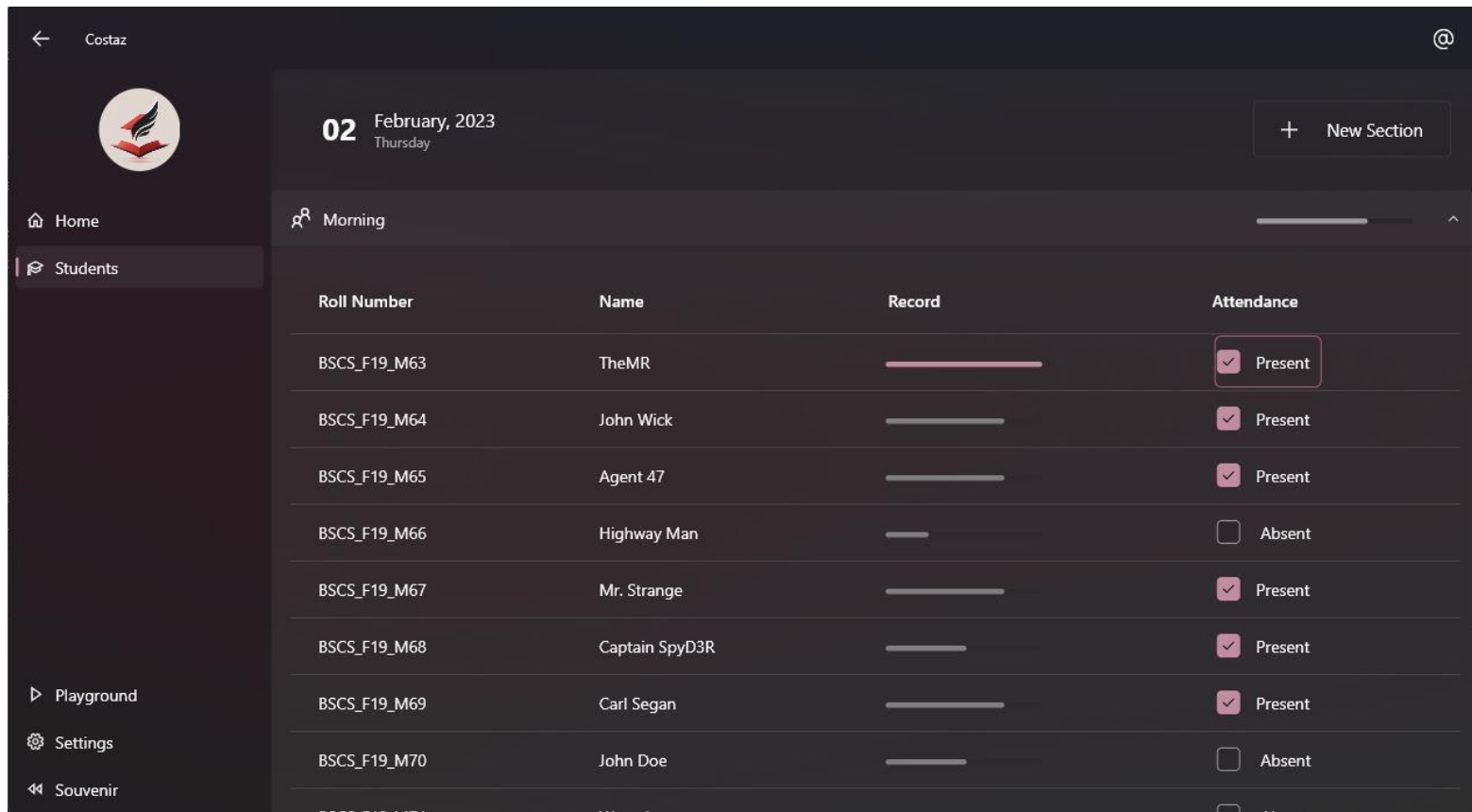


Figure 6 —Similar case is with Light Mode

### 10.3.1 Student's Context Menu

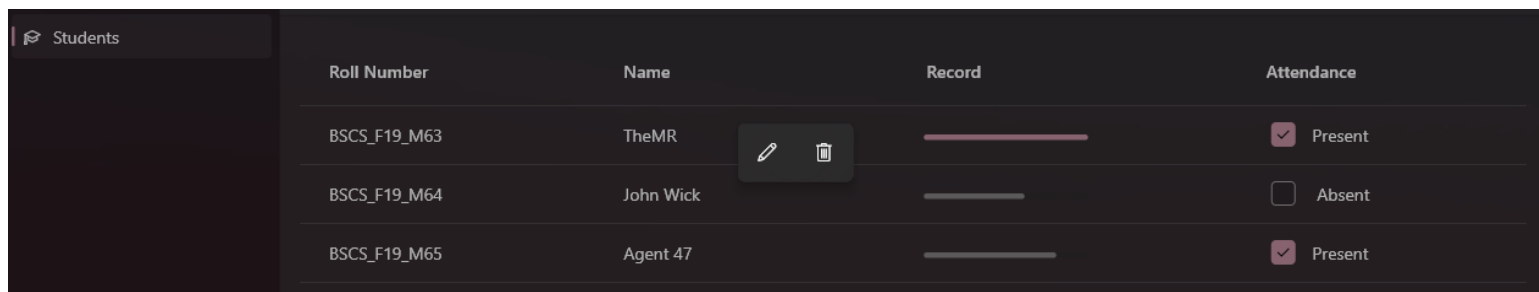


Figure 7 — Student has the Smart version of Context Menu

### 10.3.2 Adding Student Details

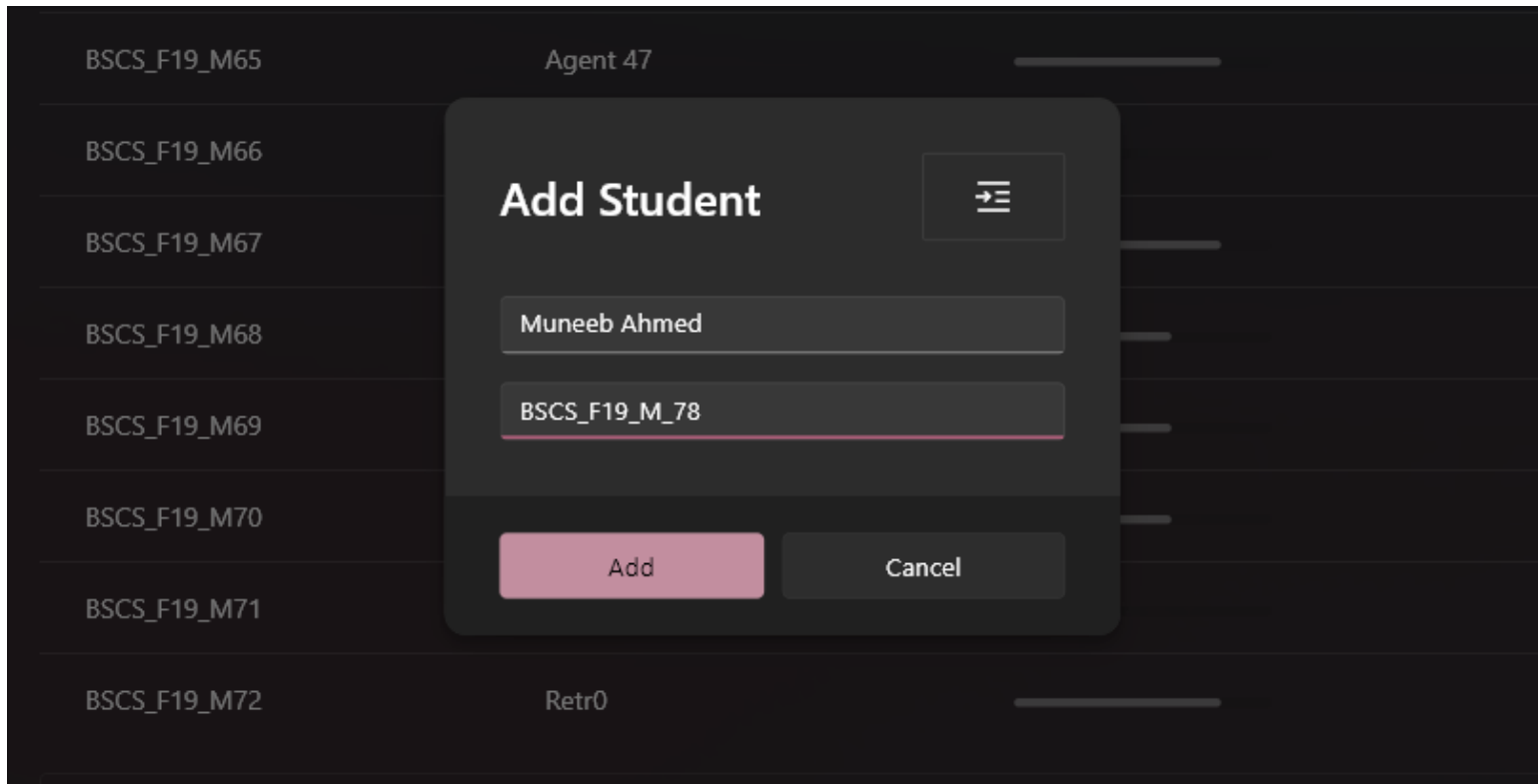
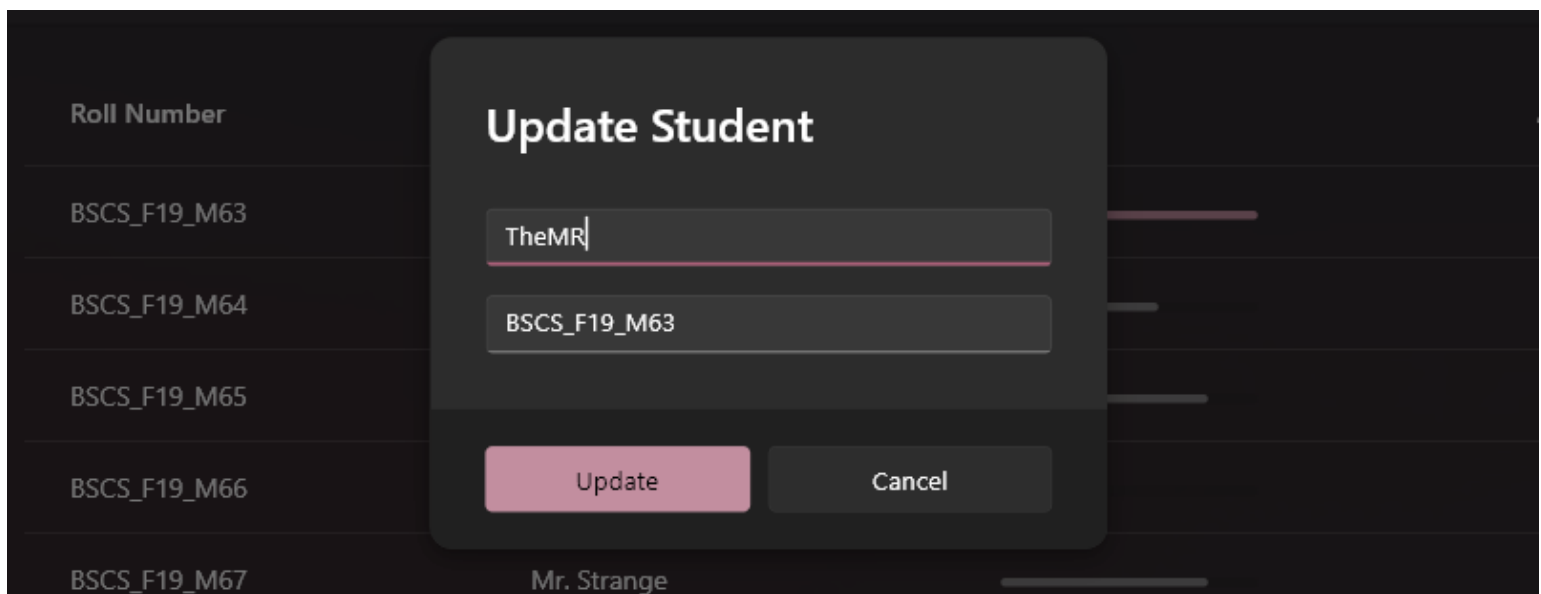
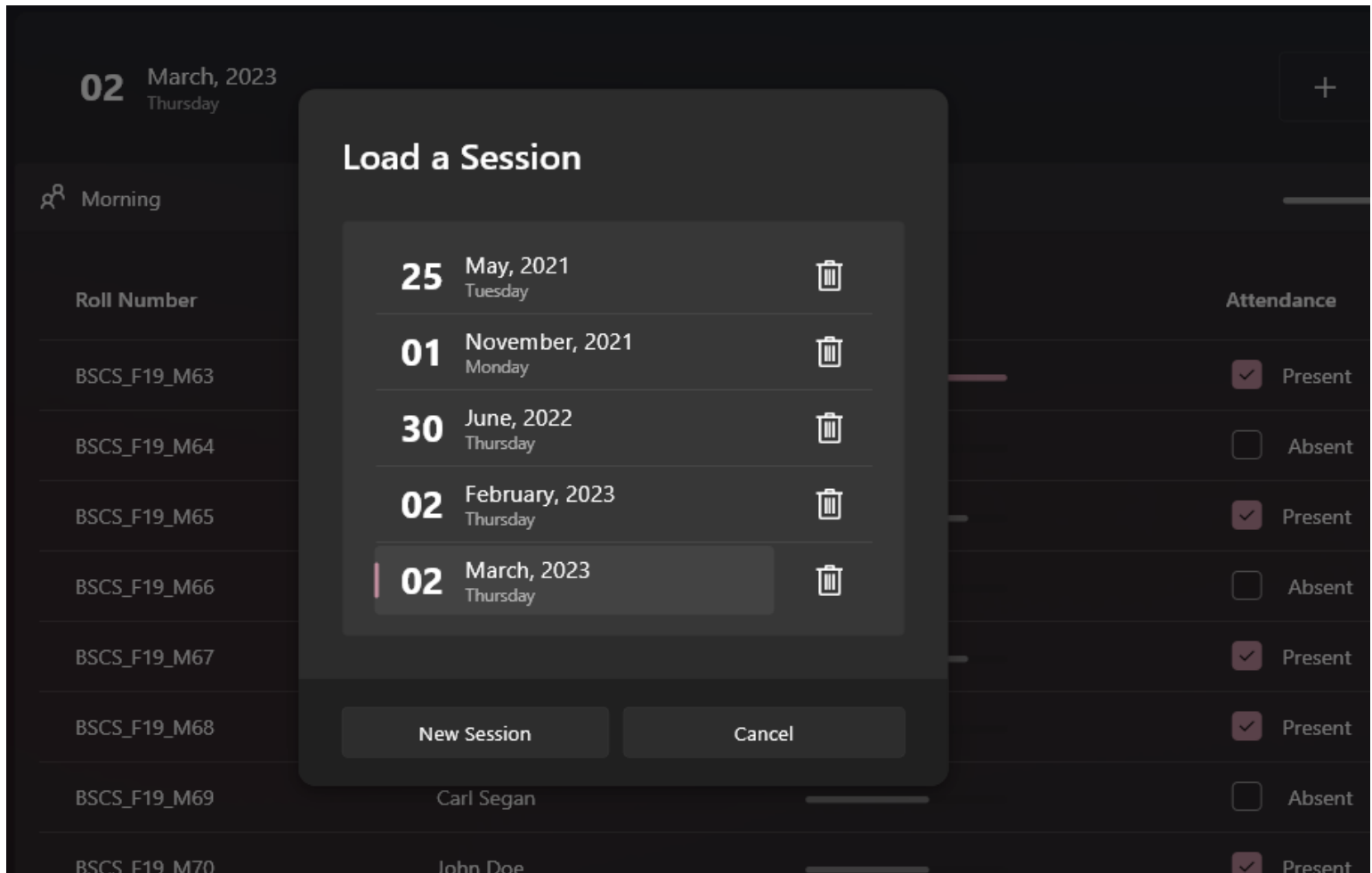


Figure 8 — Button "Add Student" is provided inside the Section's Dropdown Menu

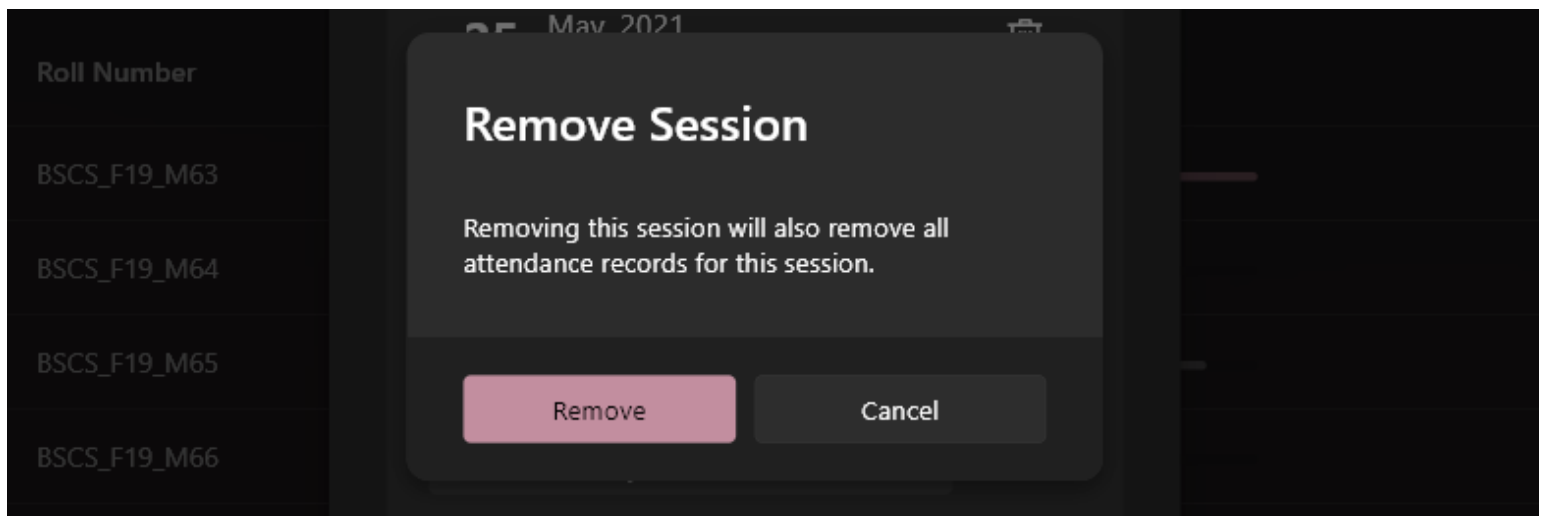
### 10.3.3 Editing Student Details



## 10.4 Session Management



### 10.4.1 Delete Session



PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 10.5 Section Tile (“Morning” in this Example)

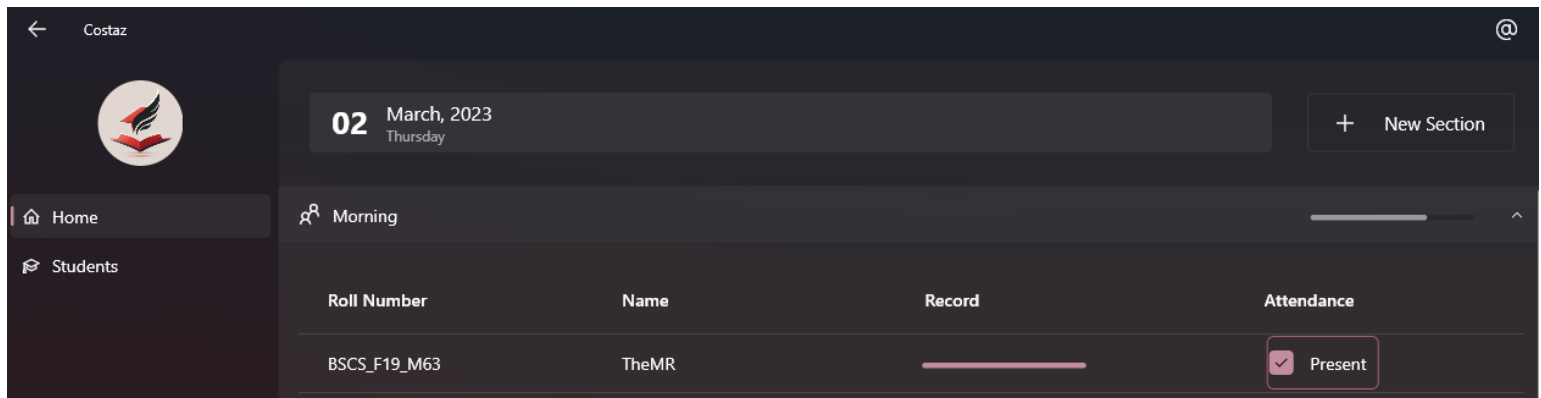


Figure 9 — Sections are made to categorize students

### 10.5.1 Creating a New Section

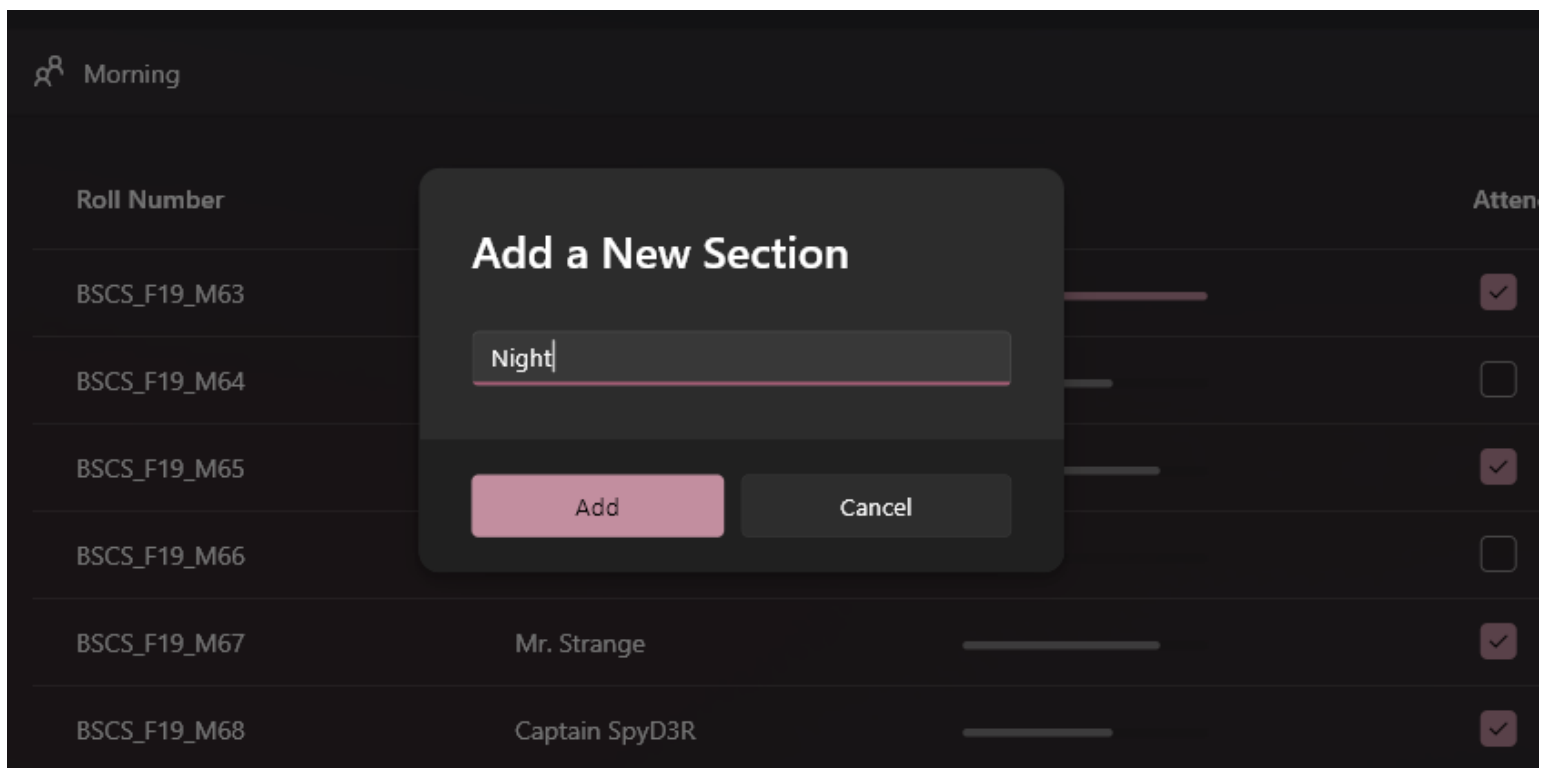


Figure 10 — by clicking the "New Section" Button, as shown in Figure 9



## 10.5.2 Context Menu to Edit/Delete Section

Morning			
Roll Number	Name	Record	Attendance
BSCS_F19_M63	TheMR		<input checked="" type="checkbox"/> Present

## 10.5.3 Editing Section

Roll Number
BSCS_F19_M63
BSCS_F19_M64
BSCS_F19_M65
BSCS_F19_M66

### Edit Section

## 10.5.4 Delete Section

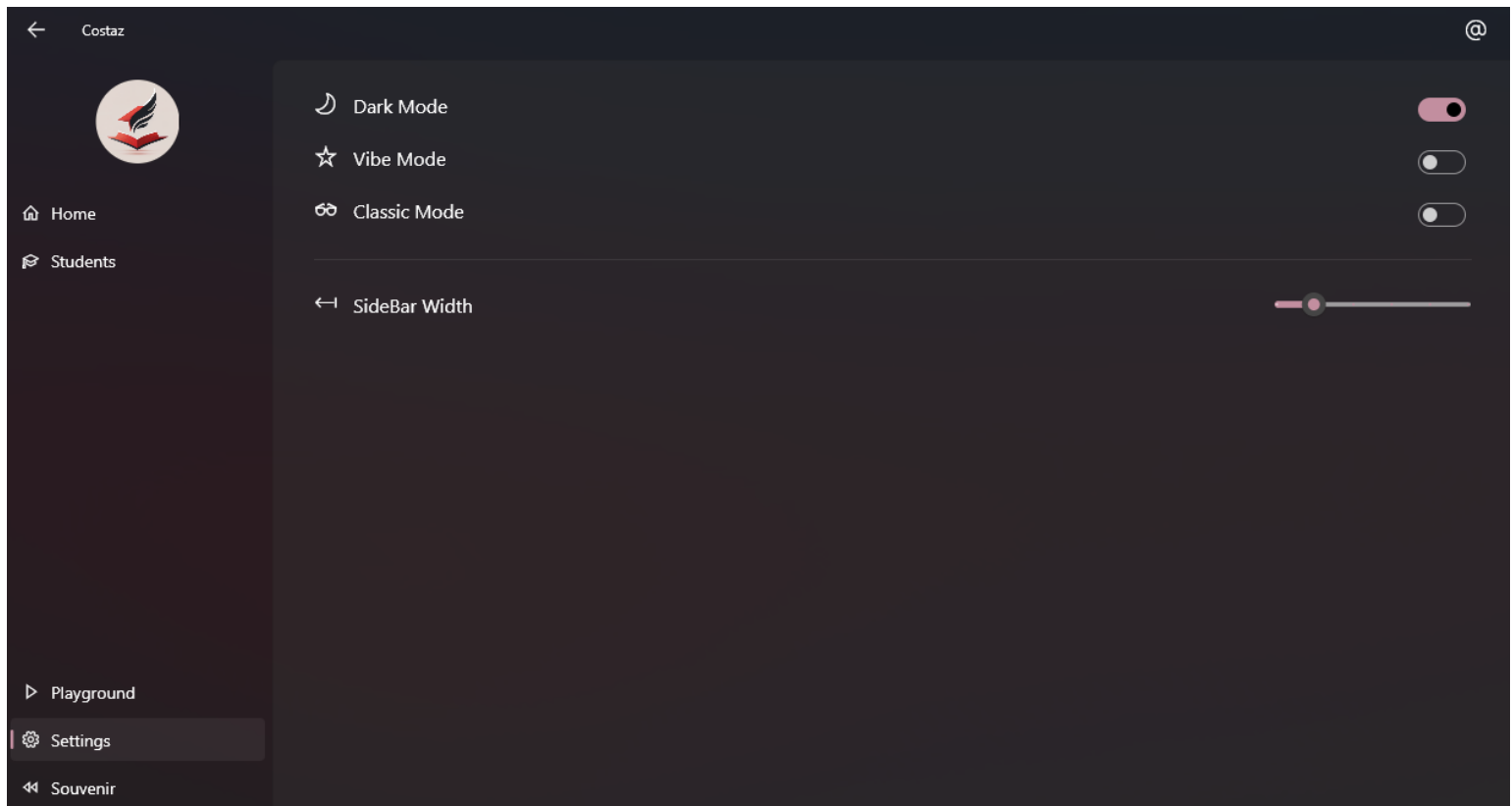
Roll Number
BSCS_F19_M63
BSCS_F19_M64
BSCS_F19_M65
BSCS_F19_M66

### Delete Section

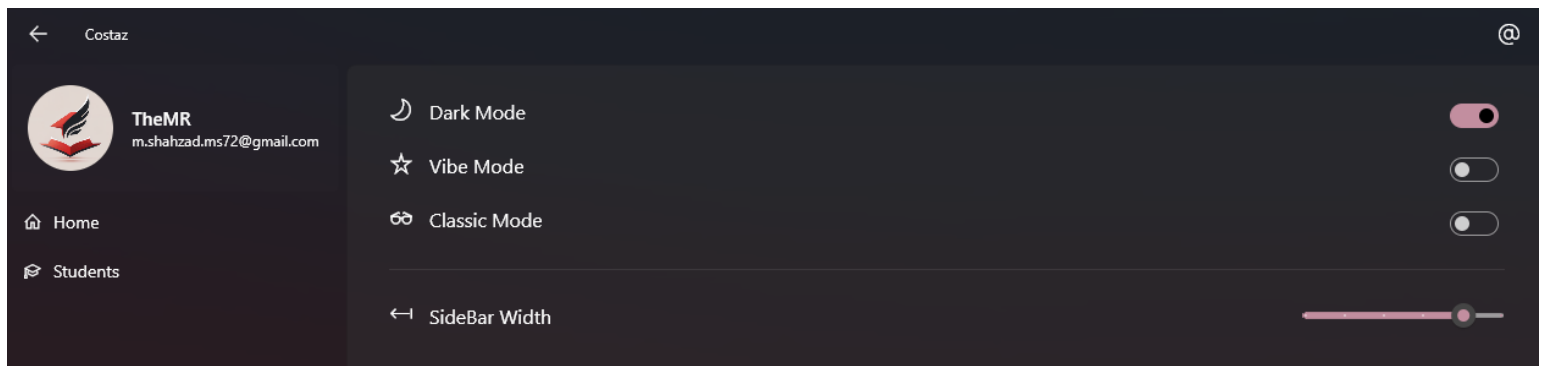
Are you sure you want to delete this Section?

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 10.6 Settings Page



## 10.7 Adjustable Sidebar



PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## 10.8 Theme Modes

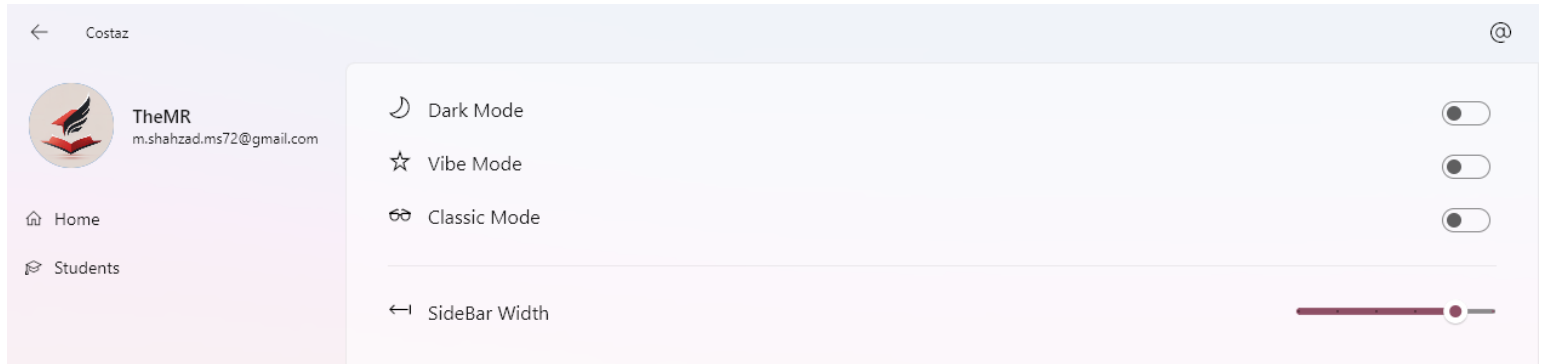


Figure 11 — Light Mode (Mica by default)

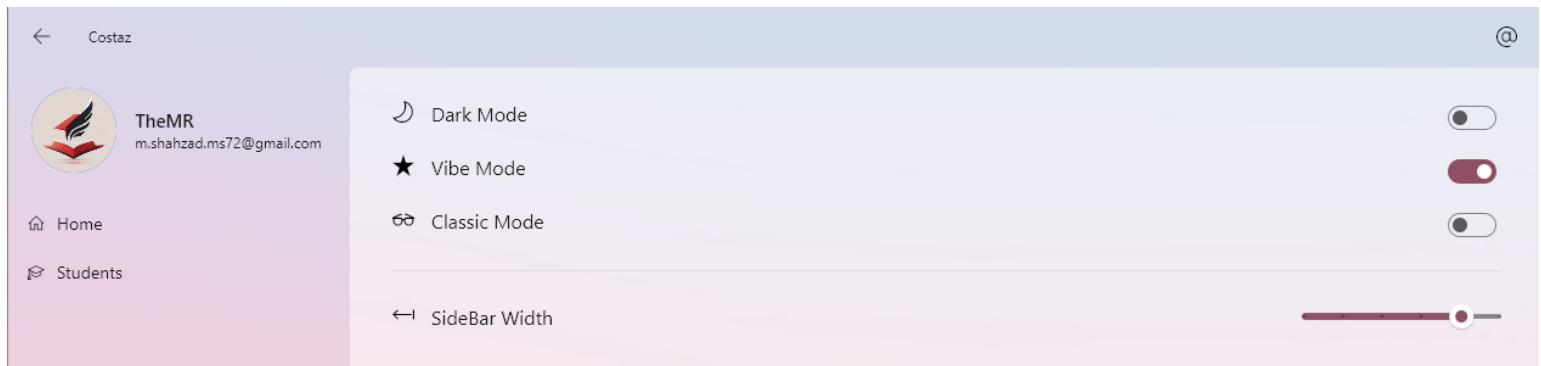


Figure 12 — Tabbed Design (from Windows 11 22H2)

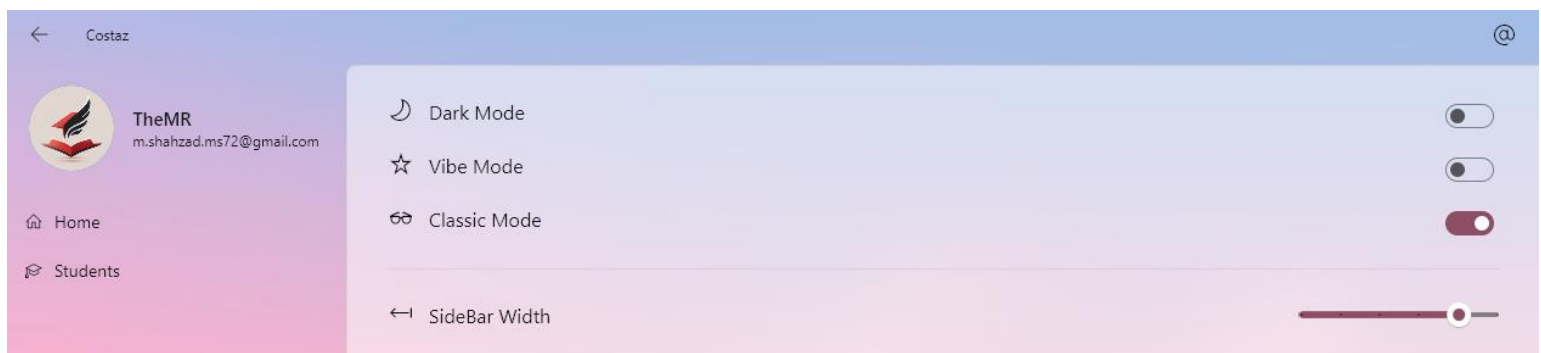


Figure 13 — Acrylic Design (from Windows 10)

## 10.9 Records Preview

**Attendance Record**

Roll Number	Record	Attendance
BSCS_F19_M63	25 May	<input checked="" type="checkbox"/> Present
BSCS_F19_M64	01 November	<input type="checkbox"/> Absent
BSCS_F19_M65	30 June	<input checked="" type="checkbox"/> Present
BSCS_F19_M66	02 February	<input type="checkbox"/> Absent
BSCS_F19_M67	02 March	<input checked="" type="checkbox"/> Present
BSCS_F19_M68		<input checked="" type="checkbox"/> Present
BSCS_F19_M69	Carl Segan	<input type="checkbox"/> Absent
BSCS_F19_M70	John Doe	<input checked="" type="checkbox"/> Present

**Attendance Record**

Roll Number	Record	Attendance
BSCS_F19_M63	25 May	<input checked="" type="checkbox"/>
BSCS_F19_M64	01 November	<input checked="" type="checkbox"/>
BSCS_F19_M65	30 June	<input checked="" type="checkbox"/>
BSCS_F19_M66	02 February	<input checked="" type="checkbox"/>
BSCS_F19_M67	02 March	<input checked="" type="checkbox"/>
BSCS_F19_M68	06 March	<input type="checkbox"/>
	Captain SpyD3R	

## 10 Testing

### 10.1 Test case create class:

ID	Description	Steps	Data	Expectation	Obtained Result	Passed
01	Creating a Class	Click on Add Class Button	Name: 'Anything'	Class Created	Class Created	Yes
02	Creating a Class	Click on Add Class Button	Name: 'Anything' Description: 'nice'	Class Created	Class Created	Yes
03	Creating a Class	Click on Add Class Button	Name: '' Description: 'nice'	Shown Error Message	Shown Error Message	Yes

### 10.2 Test case Editing Class:

ID	Description	Steps	Data	Expectation	Obtained Result	Passed
01	Editing a Class	— Right Click on any class — Click on Edit Button	Name: 'Anything'	Class Edited	Class Edited	Yes
02	Editing a Class	— Right Click on any class — Click on Edit Button	Name: 'Anything' Description: 'nice'	Class Edited	Class Edited	Yes
03	Editing a Class	— Right Click on any class — Click on Edit Button	Name: '' Description: 'nice'	Shown Error Message	Shown Error Message	Yes

### 10.3 Test case creating Session:

ID	Description	Steps	Data	Expectation	Obtained Result	Passed
01	Creating a Session	— Click on any class — Click on Session	Running App for the first time	Session created for Today	Session created for Today	Yes
02	Creating a Session	— Click on any class — Click on Session	Clicked on “New Session”	Session Created	Session Created	Yes
03	Creating a Session	— Click on any class — Click on Session	Clicked again on “New Session”	Shown Error Message	Shown Error Message	Yes
04	Creating a Session	— Click on any class — Click on Session	Clicked again on “New Session”, the other day	Session Created	Session Created	Yes

### 10.4 Test case creating Section:

ID	Description	Steps	Data	Expectation	Obtained Result	Passed
01	Creating a Section	— Click on any class — Click on “New Section”	Name: ‘Anything’	Section Created	Section Created	Yes
02	Creating a Section	— Click on any class — Click on “New Section”	Name: ‘’	Shown Error Message	Shown Error Message	Yes

### 10.5 Test case Editing Section:

ID	Description	Steps	Data	Expectation	Obtained Result	Passed
01	Editing a Section	— Right Click on any class — Right Click on any Section — Click “Edit”	Name: ‘Anything’	Section Edited	Section Edited	Yes
01	Editing a Section	— Right Click on any class — Right Click on any Section — Click “Edit”	Name: ‘’	Shown Error Message	Shown Error Message	Yes

## 10.6 Test case Adding Students:

ID	Description	Steps	Data	Expectation	Obtained Result	Passed
01	Editing a Student's Details	<ul style="list-style-type: none"> <li>— Right Click on any class</li> <li>— Expand any Section</li> <li>— Right Click any student</li> <li>— Click "Edit"</li> </ul>	Name: 'Anything' Roll: 'anything12'	Student Edited	Student Edited	Yes
02	Editing a Student's Details	<ul style="list-style-type: none"> <li>— Right Click on any class</li> <li>— Expand any Section</li> <li>— Right Click any student</li> <li>— Click "Edit"</li> </ul>	Name: '' Roll: 'anything12'	Shown Error Message	Shown Error Message	Yes
03	Editing a Student's Details	<ul style="list-style-type: none"> <li>— Right Click on any class</li> <li>— Expand any Section</li> <li>— Right Click any student</li> <li>— Click "Edit"</li> </ul>	Name: 'Anything' Roll: ''	Shown Error Message	Shown Error Message	Yes

## 10.7 Test case Editing Students Details:

ID	Description	Steps	Data	Expectation	Obtained Result	Passed
01	Adding a Student	<ul style="list-style-type: none"> <li>— Right Click on any class</li> <li>— Expand any Section</li> <li>— Click "Add Student"</li> </ul>	Name: 'Anything' Roll: 'anything12'	Student Created	Student Created	Yes
02	Adding a Student	<ul style="list-style-type: none"> <li>— Right Click on any class</li> <li>— Expand any Section</li> <li>— Click "Add Student"</li> </ul>	Name: '' Roll: 'anything12'	Shown Error Message	Shown Error Message	Yes
03	Adding a Student	<ul style="list-style-type: none"> <li>— Right Click on any class</li> <li>— Expand any Section</li> <li>— Click "Add Student"</li> </ul>	Name: 'Anything' Roll: ''	Shown Error Message	Shown Error Message	Yes

PUJCIT-Project Coordination Office	Final Version
Final Project Documentation	10 March, 2023

## PROJECT COMPLETION CERTIFICATE

It is confirmed that the report entitled "Costaz Desktop" confirms that Muhammad Ammar Khan Roll #BSCS-F19-M-63 and Muneeb Ahmed Roll # BSCS-F19-M-78 Bachelor of Computer Sciences students from the Department of Information Technology, University of Punjab Jhelum Campus contains sufficient resources required to offer the above degree.

**Ms. Aminah Ali** (Supervisor)

Lecturer

Email: aminaali@pujc.edu.pk

Department of Information Technology  
University of the Punjab, Jhelum Campus