Student Management System

S.M.S

C451

David Doyle

Table of Contents

1. Customer Problem Statement
2. System Requirements
3. Functional Requirement Specifications
4. System Sequence Diagram
5. Activity Diagram
6. User Interface Specifications
7. Project Plan

Customer Problem Statement:

Keeping track of the personal information for every student can be quite a large task. The proposed system will allow all this personal information to be stored in one location and with the ability to review later. The main objective of the project is to create an efficient and accurate way to track student information, including name, email, date of birth (DOB), emergency contact, and other relative information. The system should also provide an easy-to-use interface that faculty and administrators can use to update information. Using the system, the user will be able to enter, view, and delete information from a wide range of areas. The system will also provide a calendar that can be interacted with to schedule appointments and events. An in-program notepad will also be available so that notes pertaining to each student can be input and saved for review.

System Requirements:

Functional:

REQ-1	HIGH	System should be accessible to all teachers and administrators on a PC.
REQ-2	HIGH	Name, DOB, Student ID, E-mail Address, Contact Information should be able to be stored and saved into database.
REQ-3	MEDIUM	Information in system is secured via a log in screen.
REQ-4	LOW	Date and Time shown in main window
REQ-5	HIGH	The system should allow for the editing/updating of a record after the record has been created.
REQ-6	HIGH	System allows for a record to be deleted.
REQ-7	HIGH	Student Record can be displayed by clicking on student name.
REQ-8	LOW	The database the stores the student records can be searched via student name.

Nonfunctional Requirements:

Functionality, Usability, Reliability, Performance, and Supportability (FURPS)

Priority 1 (HIGH):

Functionality: This specifies the features and functions that the system must provide. The Student Management System can provide a way to input and store a variety of information. This information can then be recalled later, modified, and/or deleted.

Usability: This requirement deals with ease-of-use of the system. Does it flow? Is it easy to understand? The main priority here is the user interface and making sure that it is not confusing.

Priority 2 (MEDIUM):

Reliability: This requirement ensures that the system can perform its intended functions correctly and consistently.

Performance: Specifies the systems response time.

Supportability: Requires that the system can be updated, maintained, and supported over time.

User Interface Requirements:

REQ-1	HIGH	‘Add Student’ will allow student information to be entered and submitted.	Add Student
REQ-2	HIGH	‘Update’ allows for the editing of a previously entered student	Update Record
REQ-3	HIGH	‘Delete’ will allow a student profile.	Delete Record
REQ-4	HIGH	‘Show Record’ will show the currently selected record from the database.	Show Record
REQ-5	HIGH	‘Search’ will allow the user to search the database.	Search
REQ-6	Medium	‘Add Student’ window will allow for the entry of specific information based on field.	“Field Name”: ▭

REQ-7	Medium	'Search Student' window will use same field names as 'Add Student' to search database.	"Field Name": 🟦
REQ-8	MEDIUM	Log In/Password Input	Username: 🟦 Password:🟦

Functional Requirement Specifications:

Stakeholders:

- Students
- Teachers
- School Administration
- Parents; may be stakeholders but not actors.

Actors and Goals:

- Primary Actors
  - Student – The main objective of the student administration system is to organize student information quickly and safely. So, one of the main actors will be the students themselves. The student will be able to access to system and either create a profile for themselves or update any information in their profile. The student will be able to access their own profile and provide it with information such as name, DOB, address, etc.
  - Teacher – The teacher will be able to access the system and have the same options that the student will have. Teachers will be able to access the records stored in the database, such as student name, contact info, email address, mailing address, and emergency contact. The teacher will also be able to retrieve the information of any student in the system.
- Secondary Actors:
  - School Administration – These actors will have the same level of access as the teacher. While they may not need to access the areas where information is input into the system they will be able to pull records from the database and update them if necessary
  - System User – Used to manage the logins of other users, manage student profiles, and authentication.

Use Cases: 1 per workable hour

Admin (total: 9)

- Login/Logout: To login/logout of portal used to access the system (2)
- Change Account Password: Change password option for students and teachers (2)

- View Profile: To add windows to show system options and system contents (database) (2)
- Manage Student: Allows for the management of a student profile (1)
- Update Profile: Allows for the update of student profile (2)

Student (total: 5)

- View Profile: Allow the access of student profile in the system (1)
- Login/Logout System: Allow the student to log into the system (1)
- Update Profile: Update the content of student profile in database (2)
- Change Account Password: Change password for login (1)

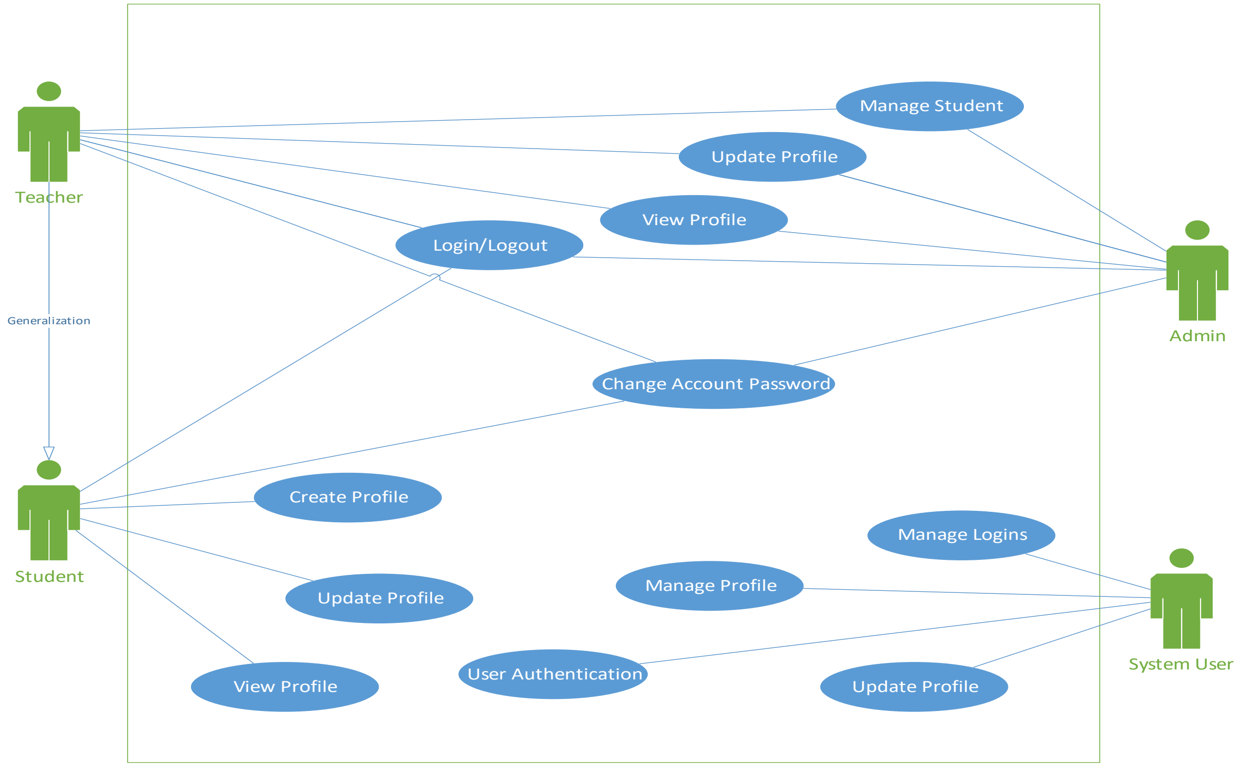
Teacher (total: 6)

- Login/Logout: Allow login and logout of system (2)
- View Profile: View selected record/profile of student (1)
- Update Profile: Allows for information update of already existing record (2)
- Change Account Password: Change password for login (1)

System (total: 6)

- Manage Logins: To manage the logins/logouts of student and teachers (2)
- Manage Profile: To manage the profiles created by student and teachers (1)
- User Authentication: Used to authenticate the login/logout process of a student, teacher, or administrator (1)
- Update Profiles: Manage the updating of user profiles (2)

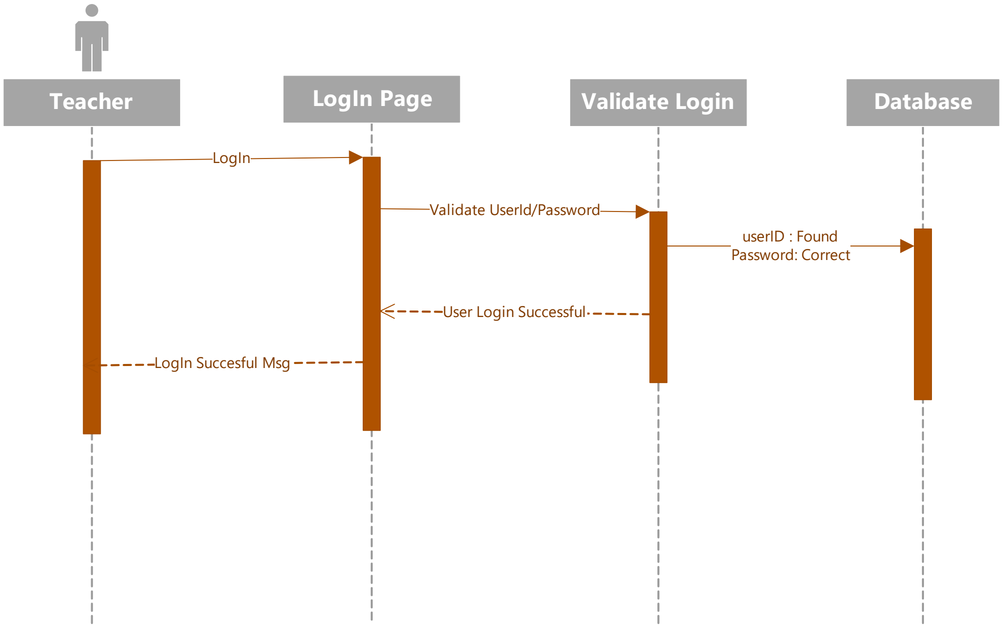
Class Diagram:



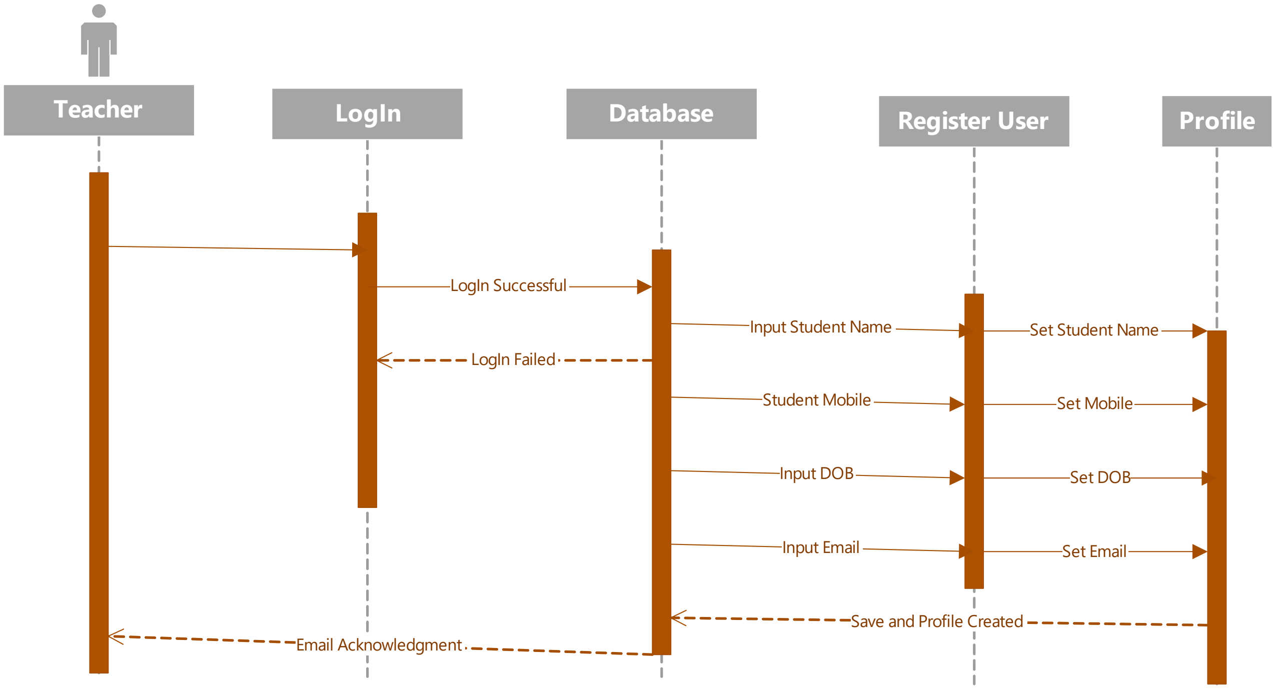
System Sequence Diagrams:

Sequence Diagrams -

User (Teacher) LogIn:

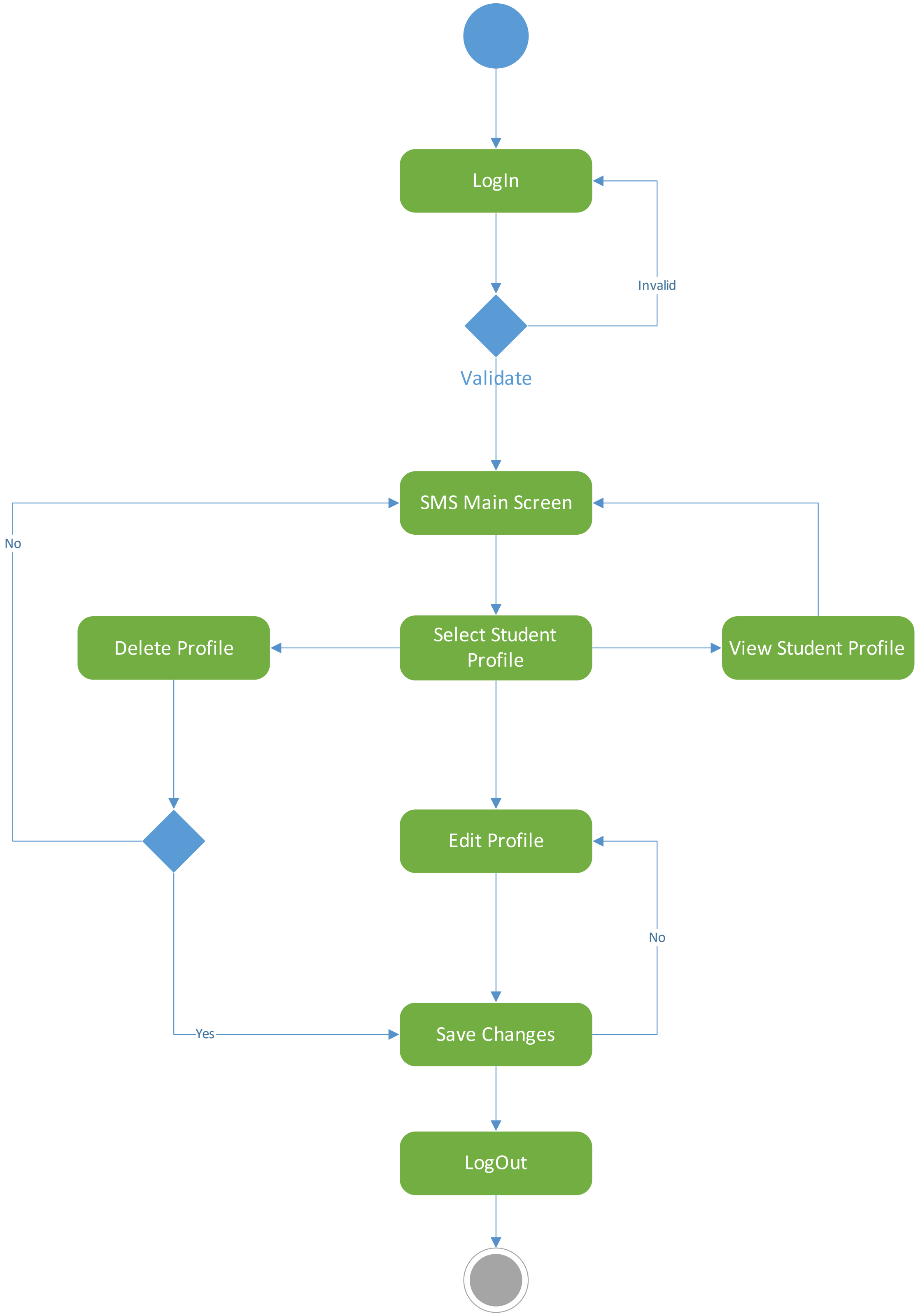


Create New Student Profile:



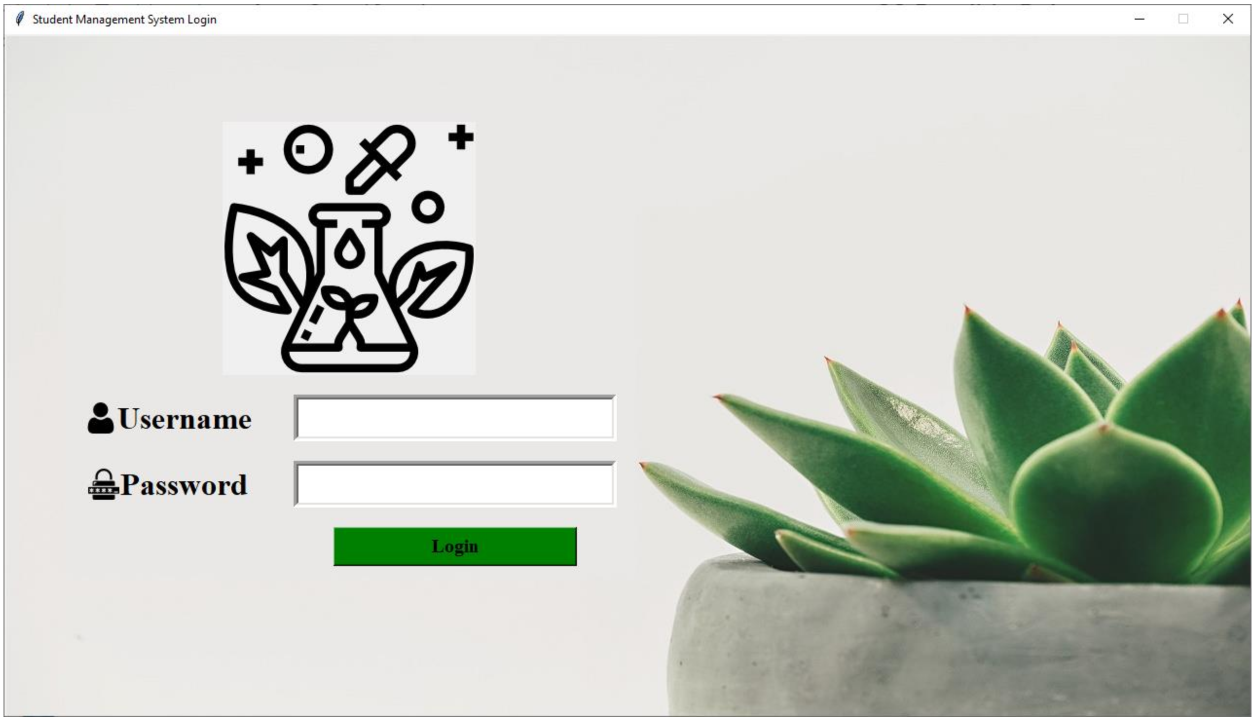
Activity Diagram:

LogIn, Edit, Delete, View Student Profile, Save Changes:



User Interface:

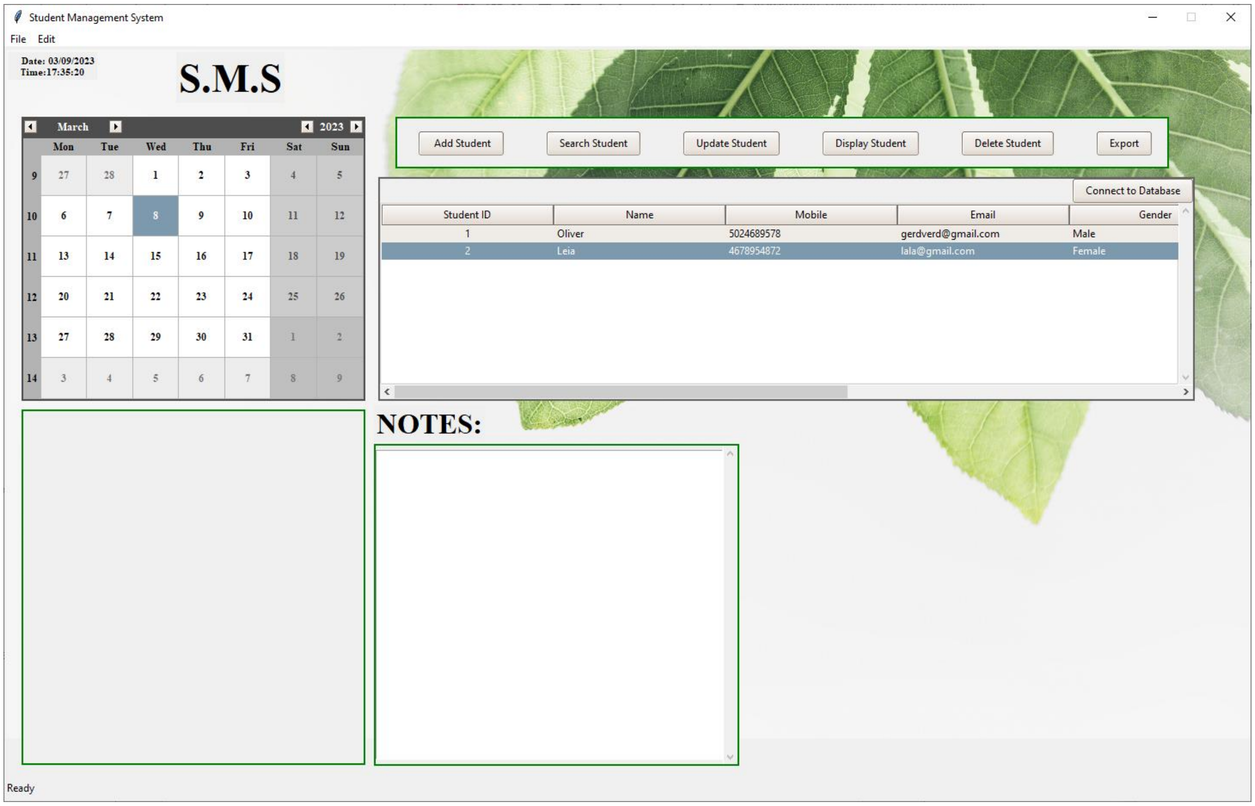
Log In:



Here is a screenshot of my user login page. There are 3 options that can occur on this page.

1. User enters the correct credentials and get a login successful prompt which then proceeds to the main application.
2. User enters incorrect credentials and will get a 'Error' 'Please enter correct Username and Password' prompt.
3. User enters no credentials which and will get a 'Error' 'Please input Username and Password' prompt.

Main:



	Mon	Tue	Wed	Thu	Fri	Sat	Sun
9	27	28	1	2	3	4	5
10	6	7	8	9	10	11	12
11	13	14	15	16	17	18	19
12	20	21	22	23	24	25	26
13	27	28	29	30	31	1	2
14	3	4	5	6	7	8	9

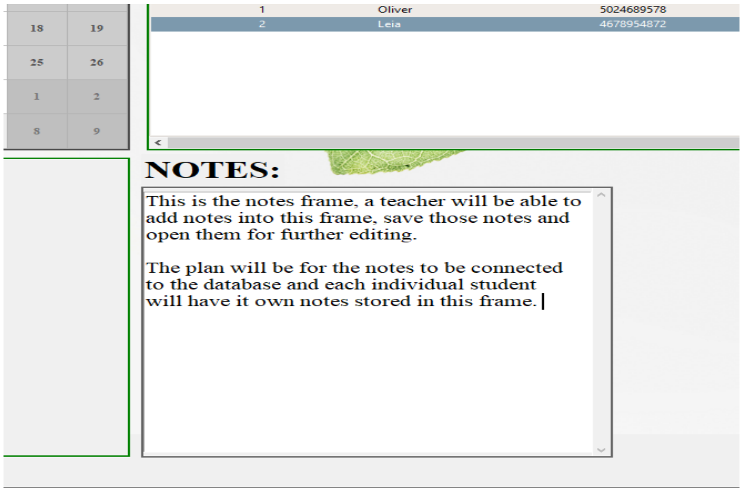
Student ID	Name	Mobile	Email	Gender
1	Oliver	5024689578	gerdverd@gmail.com	Male
2	Leia	4678954872	lala@gmail.com	Female

This is the main application part of my Student Management System (SMS). This entire window is divided up into 6 frames.

Calendar and Event Frame:

The top left frame is the Calendar frame, and it will work alongside the frame below it. This is where the user will be able to add events or appointments and in the frame below the Calendar will display upcoming events and appointments.

Notes Frame:



1	Oliver	5024689578
2	Leia	4678954872

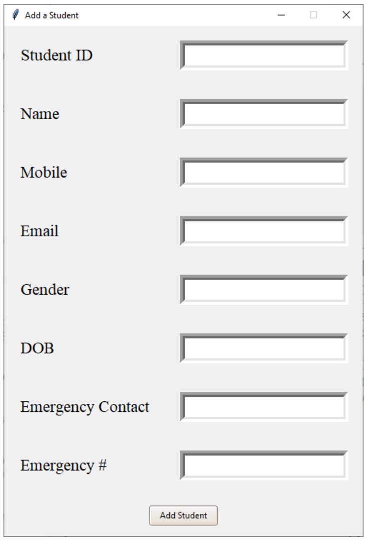
This is the notes frame, a teacher will be able to add notes into this frame, save those notes and open them for further editing.

The plan will be for the notes to be connected to the database and each individual student will have it own notes stored in this frame.

Button Frame:

The top middle frame is known as the Button Frame, and it works with the main database frame in the middle of the program. This frame contains the Add, Search, Update, Display, Delete, and Export buttons. This is the main interactive feature of the program where a teacher will modify the database. When a button is clicked a window will open for the teacher to provide the inputs.

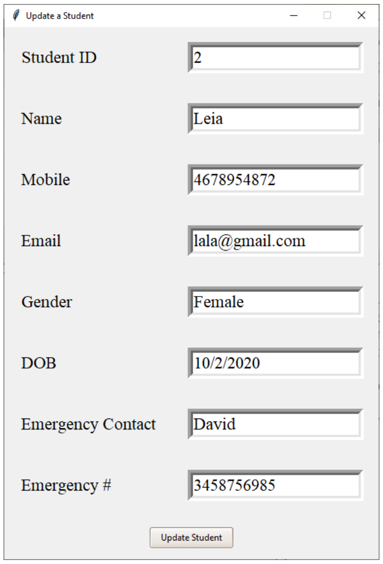
Add:



Add a Student

Student ID	
Name	
Mobile	
Email	
Gender	
DOB	
Emergency Contact	
Emergency #	

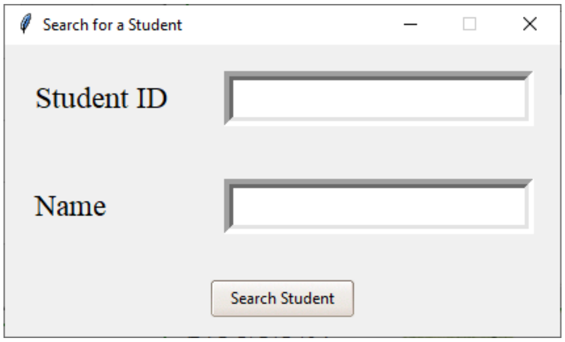
Update: (When this button is clicked while a record is selected the current information will load in the window)



Update a Student

Student ID	2
Name	Leia
Mobile	4678954872
Email	lala@gmail.com
Gender	Female
DOB	10/2/2020
Emergency Contact	David
Emergency #	3458756985

Search:



The Display button can be used to Display the entire database and the Delete button will delete the selected record.

The final frame on the lower right is still in development. I am hoping to have a grading section in this frame, where a teacher can input grades, store grades, and even calculate grades.

Project Plan:

With the main frames of the project complete I am starting to work on the Calendar frame and the Grading Frame (this frame may be reworked and not be a grading frame). I want the Calendar frame to show upcoming events and have the ability to edit any previous events that have been added. The grading frame is still in its early stages but I am hoping to have an area that will show the subjects or classes being taught and then have another area where grades can be input, updated, and collected for final grades. One thing that I have been working on is to change the buttons in the “Button Frame” to icons from the text buttons that are their already. I have included the icon images in the main source code already but am having trouble getting the icon images to act as buttons. As this is more of a visual problem I will wait until I have completed the work in the Event/Appointment and Grading frames.