CRITERION B: SOLUTION/DESIGN OVERVIEW

Input and Output

Input	Data Type	Normal Range	Example
Student Name	String	Not black	Damon Salvatore
School Name			
Program Name			
Class Type			
Unit Name			
Teacher name			
Class notes	String array/char	0 - 150 char	Improve
Grade	Int	1-10	8
Hours			
Test Grade			
Date		1 day/ 1 month/ 1 year(JCalender)	20/01/2021
Password	Char	Not black	nopassword

Output	Data Type	Normal Range	Example
Current Level	Int	1-7	5
Average Grade			
Parent Mobile		11	0682748193
Parent Name	String	Not left black	Damon
Parent email	varchar	1-150	damonatgmailcom

Testing Plan

Login Tab

Action To Test	Method/Expected Result
Teacher Name	Enter possible data error, option pane showed
Teacher Password	
Login Button	Press of Login button, the fields return to default values

Add New Teacher Tab

Action To Test	Method/Expected Result
Adding teacher name	No int
Add Password	No string, at least five and no more than ten
Re-enter Password	Add possible datas, Should be same as password, if not shown, failure option pane

Student Tab

Action To Test	Method/Expected Result
Search Student Name	data that was last entered for the student in table
Edit Class and Tuition Button	Redirect the user to individual tab
Enter New Student and Parent Information	if an error, then the fields will return to the default.
Enter and Refresh Button	Enter the information in the table under the specific attributes.

Attendance Tab

Action To Test	Method/Expected Result
Today's Data Calendar	date in JCalender. If not highlighted, click again.
Student Class Information	String name, Boolean Class Type, and Boolean Tuition Addition. Should show entered fields with wanted data(example: name).
Sorting Combo Box	Item state is changed, the table should rearrange for sorting option.

Classes Tab

Action To Test	Method/Expected Result
Search Student Name	Present data that was last entered
Enter class informations	enter the data type allowed avoid long data entry.
Sorting Combo Box	item state is changed, table shuld rearrange to accommodate the asked sorting option.
Upload from laptop button	a dialog which has imported the user laptop's all files, text field underneath shows the pathways that the user took in order to open the document. Documents can include: JPEG and PDF

Tuition Tab

| Action To Test	Method/Expected Result
Search Student Name	data that was last entered
Enter Tuition informations	data types that are allowed in the field Avoid long entry
Send Notification to Parent	select yes when the task is done
Sorting Combo Box	Item state is changed, the table should rearrange for sorting option

Final Prototype

File   Edit   Help

Login   Add New Teacher

Hello!

You need to enter the required fields below in order to proceed with the application.

Teacher Name   jTextField1

Password   ●●●●●●●●●●●●●●●

Login

JOptionPane message

OK

Image 1: Login screen for user to input name and password.

File   Edit   Help

Login   Add New Teacher

If you are a new teacher please register by filling out the inforamtion below and then proceed to login page.

Name   jTextField2

Set Password   jTextField3

Re-enter Password   jTextField4

Add New Teacher

Image 2: Add Teacher screen, user can create new teacher logins.
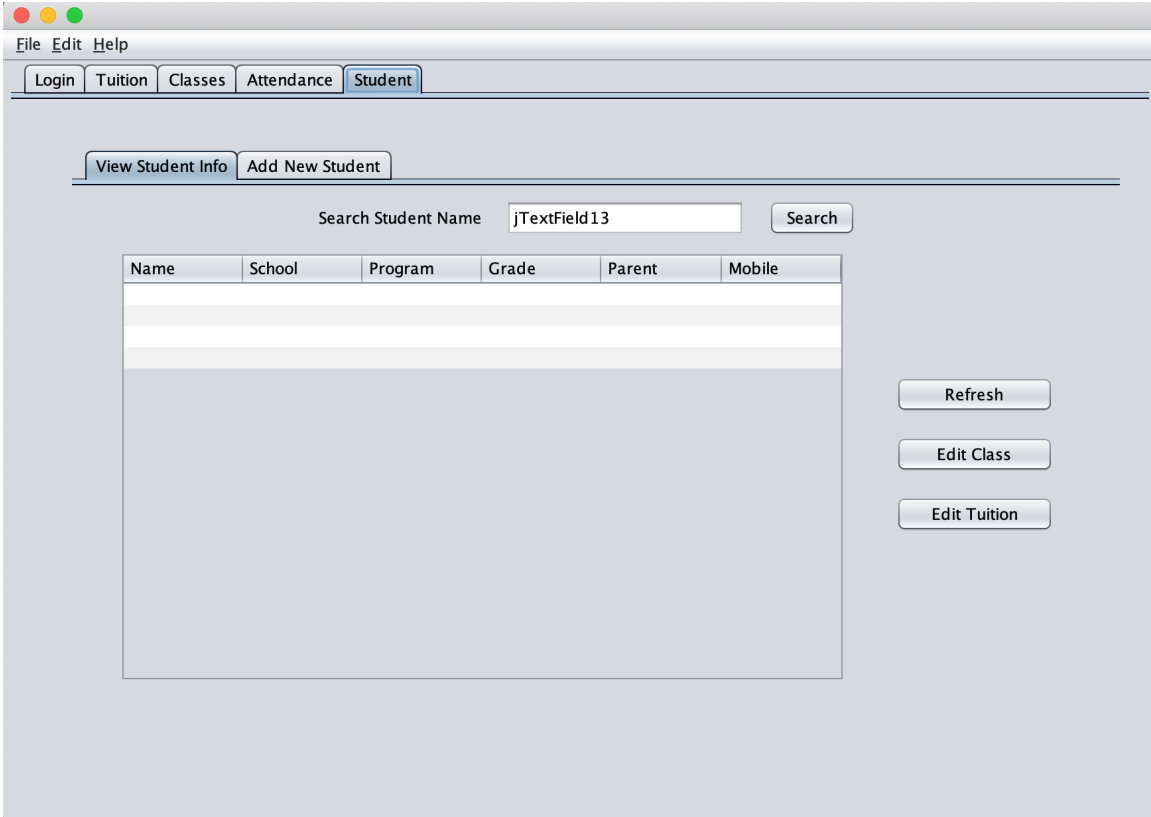
Image 3: View the entered student information in table.
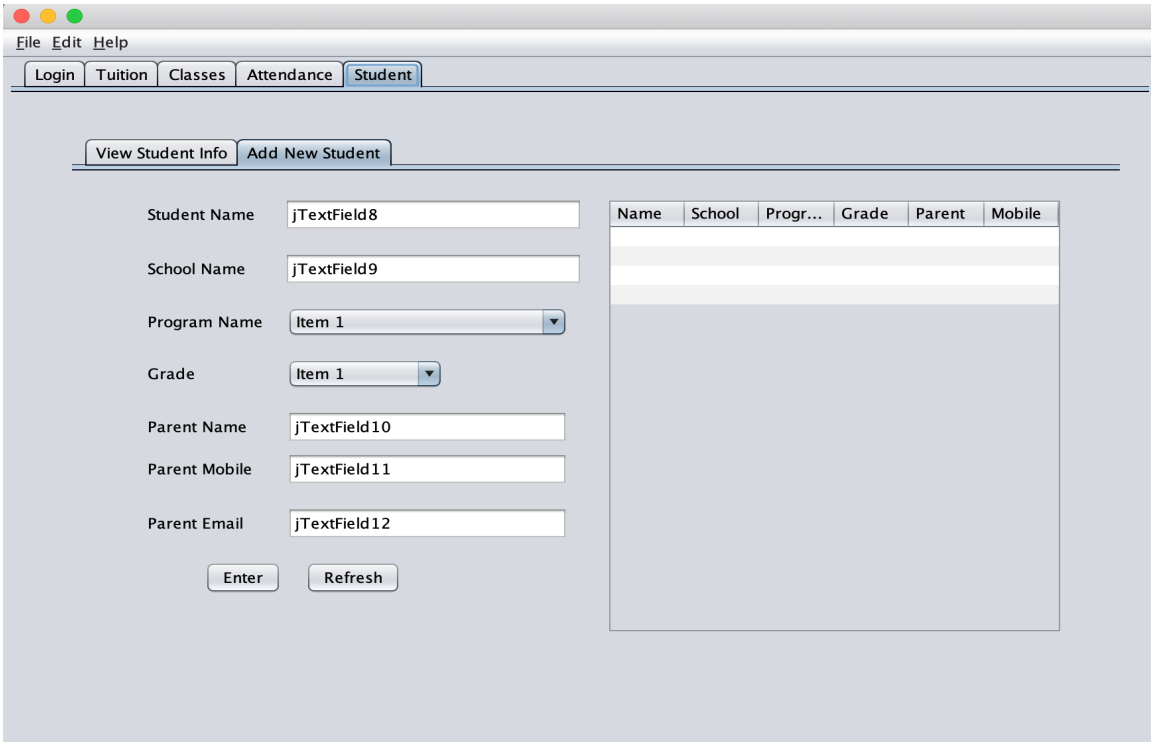
Image 4: Add new student information.
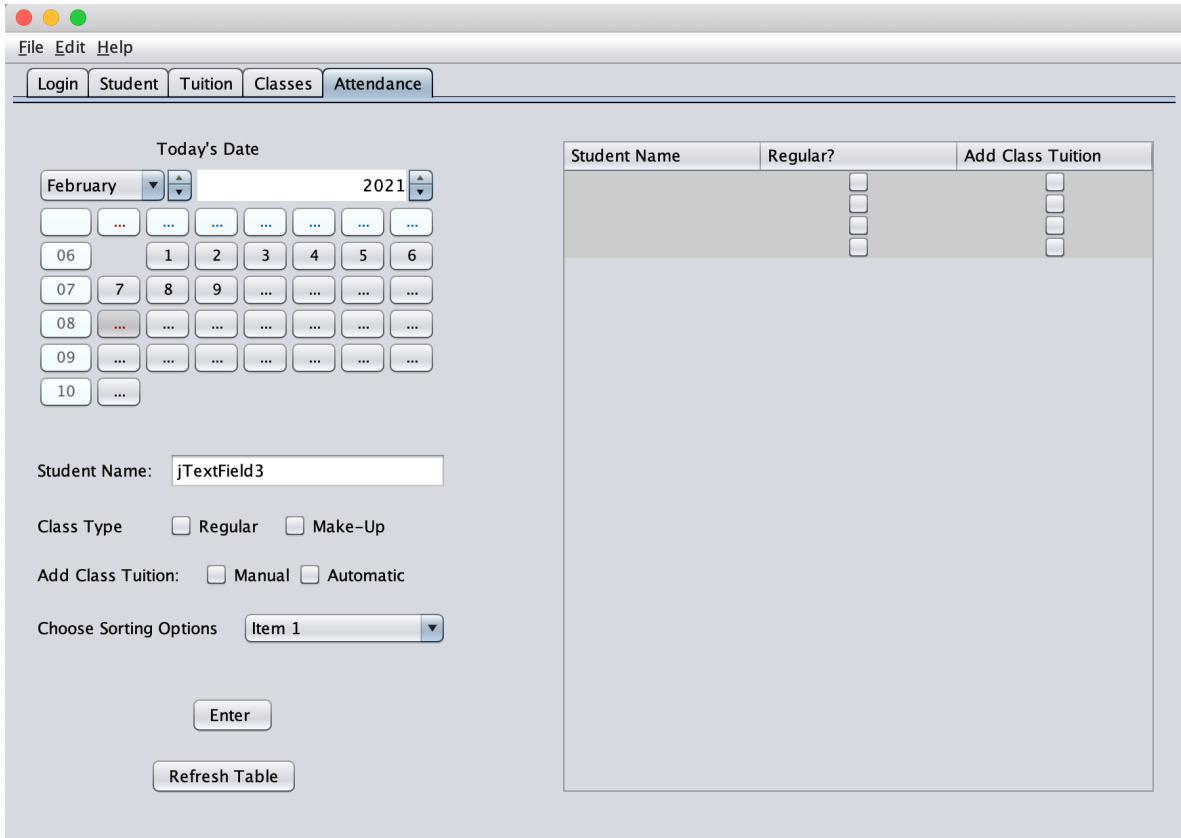
Image 5: Take attendance of students.
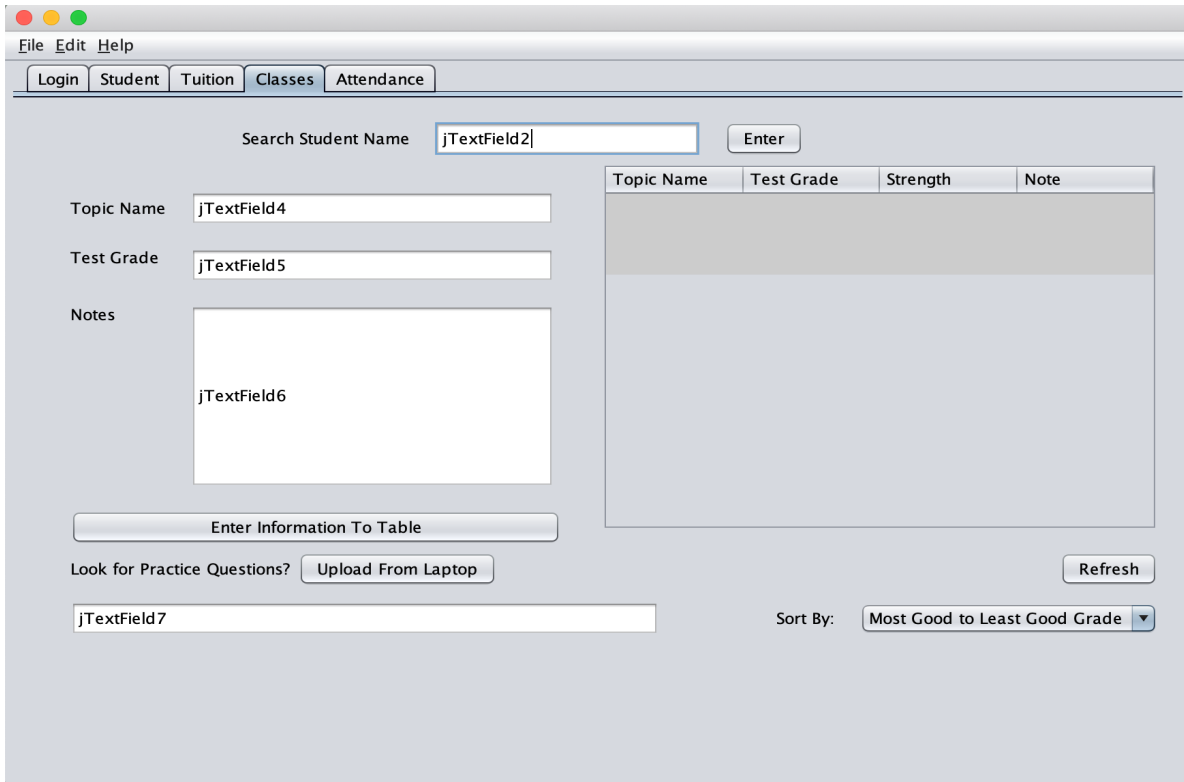
Image 6: Search entered students and add individual class information.

Image 7: Search entered students and add tuition .

Class Diagram

Teacher
- LoginUsername: String
- LoginPassword: Int

+ setLoginUsername: String
+ setLoginPassword: Int
+ getLoginUsername: String
+ getLoginPassword: Int

has a

Uses

Grade
- totalAverageGrade: double
- topicName: String
- topicTestGrade: Int

+ Student()
+ Student(name: String, schoolName: String, programName: String, gradelevel: int)
+ settotalAverageGrade: double
+ settopicName: String
+ settopicTestGrade: Int
+ gettotalAverageGrade: double
+ gettopicName: String
+ gettopicTestGrade: Int

has a

Text

Parent
- parentName: String
- parentMobile: Int
- parentEmail: String

has a

Student
- name: String
- schoolName: String
- programName: String
- gradelevel: Int

+ setName: String
+ setschoolName: String
+ setprogramName: String
+ setgradelevel: Int
+ getName: String
+ getschoolName: String
+ getprogramName: String
+ getgradelevel: Int

MainGUI

//InputTab
- nameTF: JTextField
- schoolNameTF: JTextField
- programNameCB: jComboBox
- gradelevelCB: jComboBox

//DisplayTab
- StudentInfoTable: JTable

//InputTab
- addNewStudent(): void

//DisplayTab
- refreshStudentInfoTable: void(when clicked)
- sortByProgramName(): void
- sortByGradeLevel(): void
- sortBySchoolName(): void

//InputTab
- searchStudentName(): void
- topicNameTF: JTextField
- topicTestGradeCB: jComboBox
- PracticeQuestionCHB: jCheckBox

//DisplayTab
- studentGradeTable: JTable
- sortByTestGrade(): void
- sortByTopicName(): void
- sortByStudentProgram(): void

//InputTab
- nameTF: JTextField
- classTypeCHB: jCheckBox
- AddTuitionCHB: jCheckBox
- ClassDate: jCalender

//DisplayTab
- sortByStudentName(): void
- sortByClassType(): void
- sortByClassDate(): void

//InputTab
- searchStudentName(): void
- AddTuitionCHB: jCheckBox
- ClassDate: jCalender

Parent

parentName: String

parentMobile: Int

parentEmail: String

+ Student()
+ Student(name: String, schoolName: String, programName: String, gradelevel: int)

+ setparentName: String
+ setparentMobile: Int
+ setparentEmail: String

+ getparentName: String
+ getparentMobile: Int
+ getparentEmail: String

Class

classType: Boolean

Date: Int

Hours: Int

Notes: String

+ Student()
+ Student(name: String, schoolName: String, programName: String, gradelevel: int)

+ setClassType: Boolean
+ setDate: Int
+ setHours: Int
+ setNotes: String

+ getClassType: Boolean
+ getDate: Int
+ getHours: Int
+ getNotes: String

Tuition

+ setPerClassMoney: Int
+ setTotalMoney: Int
+ setSentReminder: Boolean

+ Class()
+ Class(Student(), classType: Boolean, Date: Int, Hours: Int, Notes: String)

+ Parent()
+ Student(parentName: String, parentMobile: Int, parentEmail: String)

+ Student()
+ Student(name: String, schoolName: String, programName: String, gradelevel: int)

SentReminder: Boolean

TotalMoney: Int

PerClassMoney: Int

+ getPerClassMoney: Int
+ getTotalMoney: Int
+ getSentReminder: Boolean

- sortByClassType(): void
- sortByClassDate(): void

//InputTab
- searchStudentName(): void
- AddTuitionCHB: jCheckBox
- ClassDate: jCalender
- ClassHoursTF: JTextField
- SentNotifCHB: jCheckBox

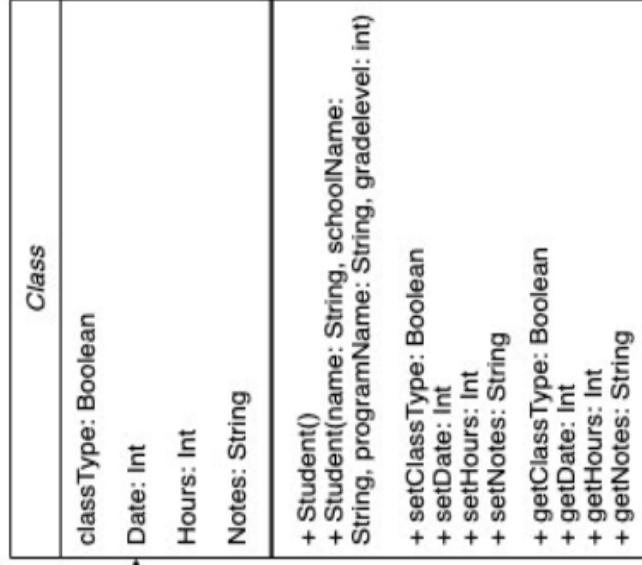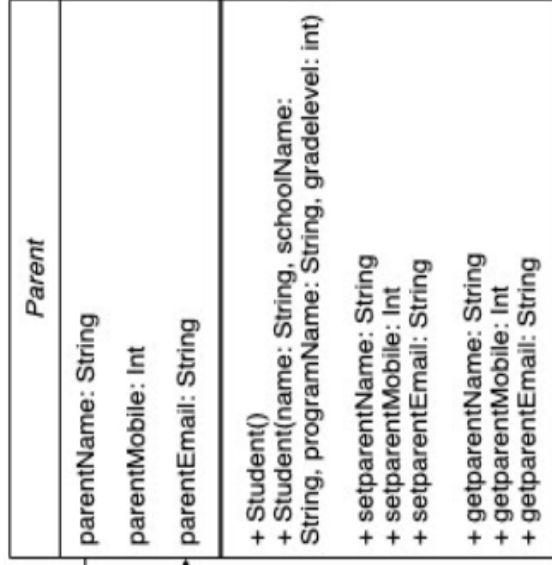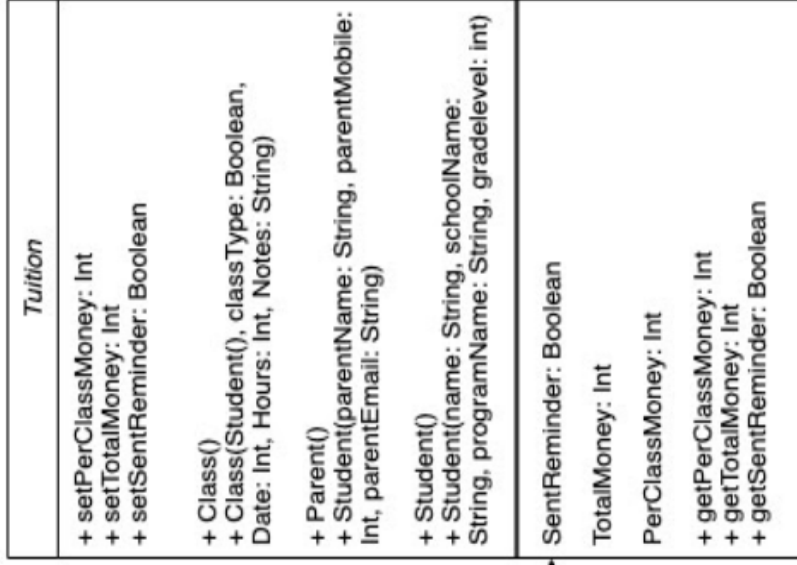//DisplayTab
- TuitionTable: JTable
- sortByClassDate(): void
- sortBySendNotif(): void

//InputTab
- LoginUsernameTF: JTextField
- LoginPassword: jPasswordField

//DisplayTab
- SuccessFailOptionPane(): void

has a

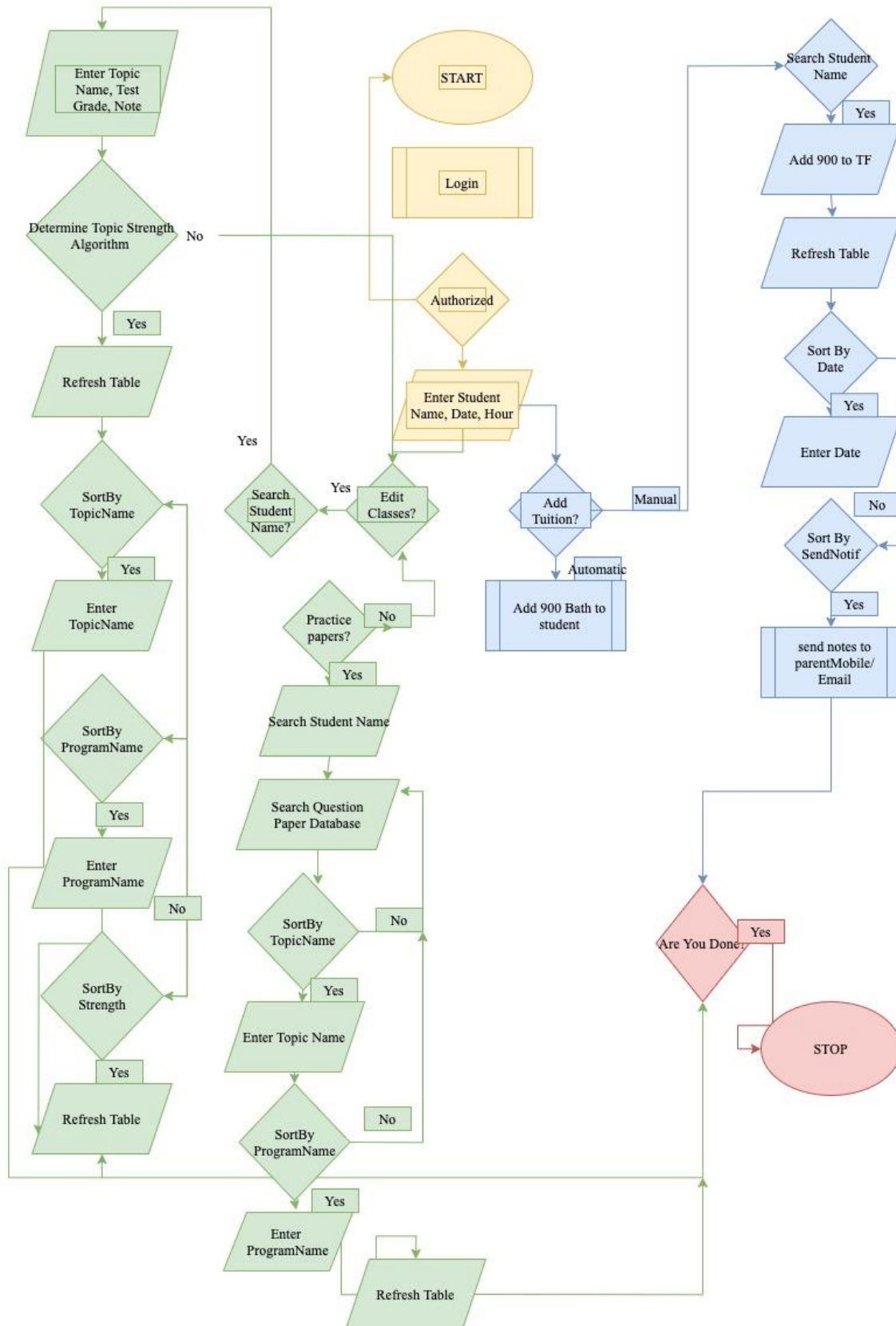has a

Uses

Flow Chart Diagram

```
                                          START

                                          Login

                                        Authorized

Enter Topic
Name, Test                         Enter Student
Grade, Note                        Name, Date, Hour

                                                                    Search Student
Determine Topic Strength    No                                          Name
Algorithm
                                                                    Yes
                                                                    Add 900 to TF
                Yes
                                                                    Refresh Table
Refresh Table
                   Yes                                              Sort By
                                  Yes    Edit        Add    Manual   Date
SortBy              Search              Classes?    Tuition?
TopicName          Student                                          Yes
                   Name?                  Automatic
    Yes                                                             Enter Date
                                         Add 900 Bath to
Enter                       Practice  No  student                          No
TopicName                   papers?
                                                                   Sort By
             SortBy            Yes                                 SendNotif
             ProgramName
                           Search Student Name                     Yes
        Yes
Enter                                                              send notes to
ProgramName               Search Question                          parentMobile/
                   No     Paper Database                           Email

SortBy                    SortBy      No
Strength                  TopicName
                             Yes
      Yes                                                          Are You Done?  Yes
                          Enter Topic Name
Refresh Table
                          SortBy       No                               STOP
                          ProgramName

                                 Yes
                          Enter
                          ProgramName

                             Refresh Table
```

Algorithms

Student Tuition Display

Search Student name
Add the individual class payment to the student tuition table
- *For example {date, hour, tuition, class type}*

For the class type column
- *If a regular class*
  - *Display box as green*
- *Else if make-up class,*
  - *Display box as yellow*

If studentNonExistant = true
- *Add New Student*
  - *Name Counter ++*

Need-to-Make-Up

If classCancelled = true
- *Schedule make-up class*

Else if classCancelled = false
- *Check TuitionTable if needToMakeUp = true*
  - *Schedule make-up class*
- *Else if needToMakeUp = false*
  - *Print no make up class needed for XYZ*

To-Update-Daily-Attendance

Look up the date
- *Enter student name*
- *Enter date*
- *Enter hour*

If doesAnyStudentNeedNotes = true
- *Search student name*
- *Enter note*

Else if doesAnyStudentNeedNotes = false
- *If wantToAddManualNotes = true*
  - *optionPaneReminder prints "please go to Student Tuition Display Panel*
- *Else if wantToAddManualNotes = false*
  - *Tuition counter ++ for entered students*

Edit-Student-Stat

If Class pane = open
- *Search student by name*
  - *Display StudentStatTable*
    - *For example {Topic Name, Test Grade, Strength, Note}*
  - *For strength column*
    - *Search StudentProgramName*
      - *If IB then take TestGrade(1 to 7) and divide by 7*
    - *SetStrengthbyGrade*
      - *If IB and is 1,2,3,4,5,6,7*
        - *Set box color to red,red,red,yellow,yellow, green, green*
    - *Repeat for A-level and AP*