STUDENT RECORD SYSTEM

SOFTWARE USER MANUAL

1. Instructions for use

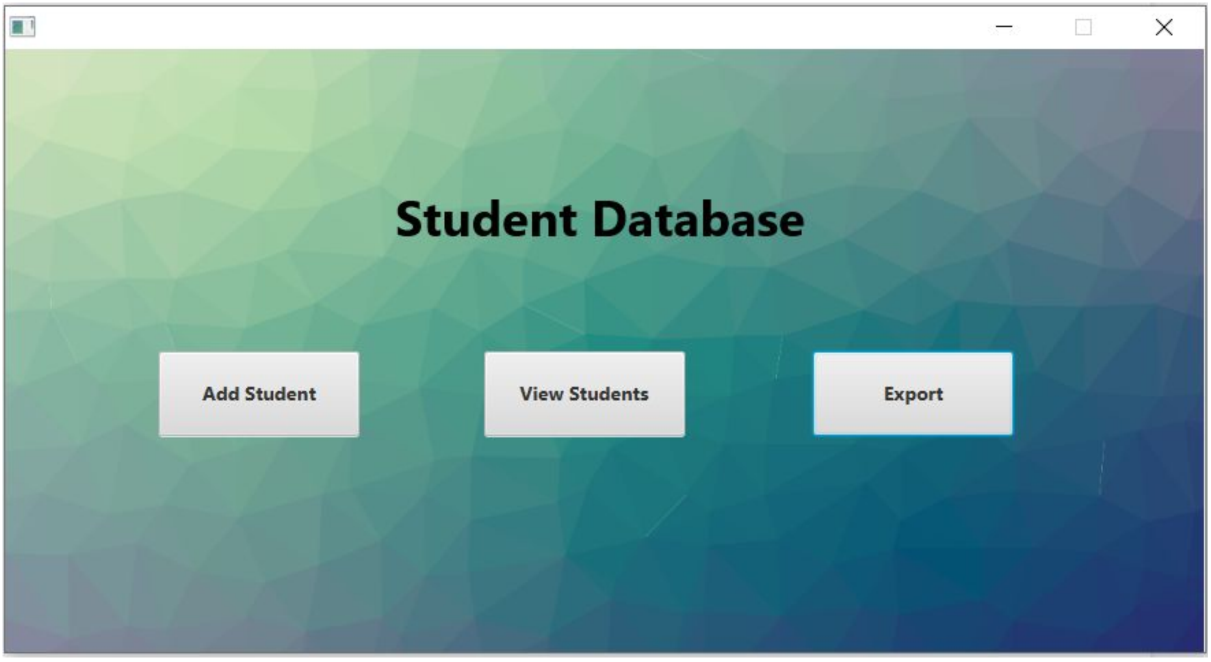
1.0 Deployment

1. Go Onto Releases And Press First Release
2. Download The Application.jar File
3. Open The File In Your Command Prompt
4. Type “**java -jar Application.jar**”
5. Use the Application

1.1 Main Menu

The window should be look like this:

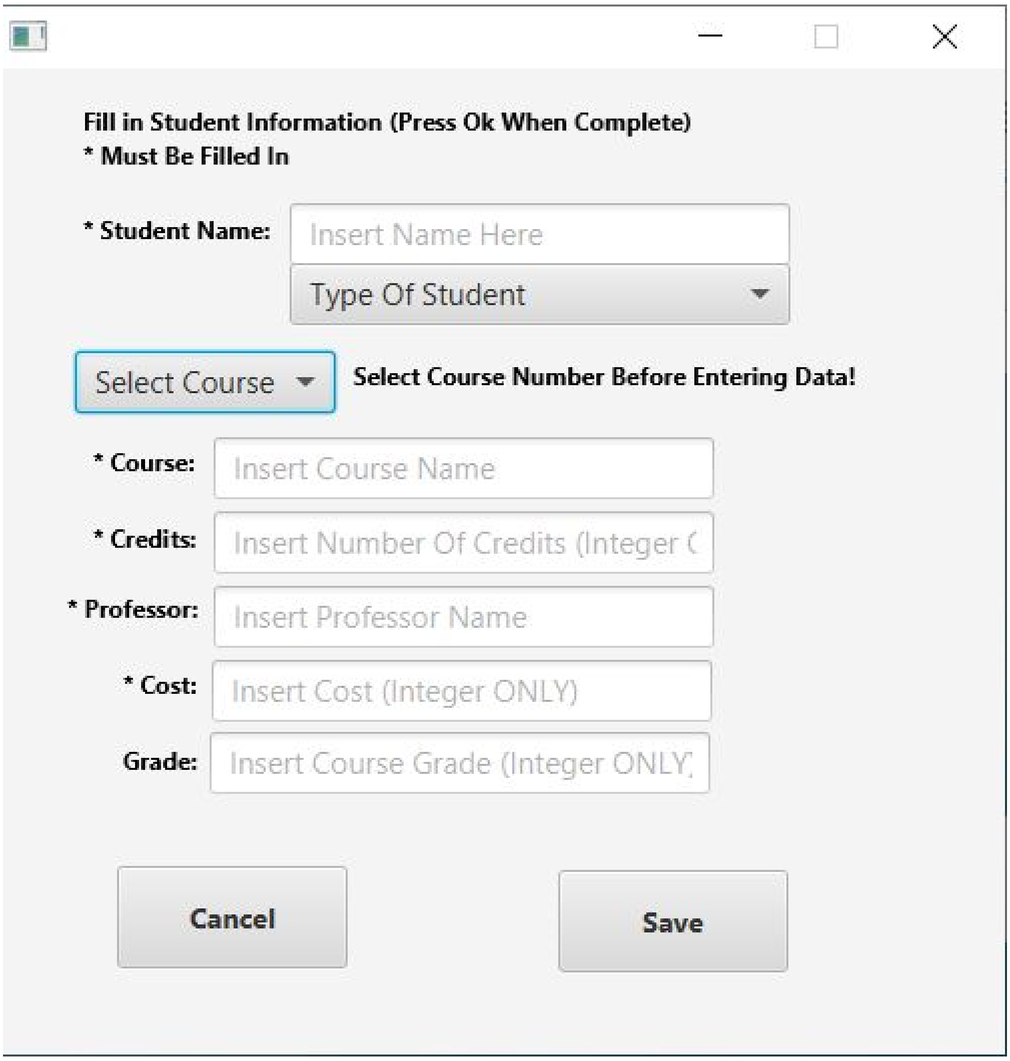
This is the Student Database Application



Student Database Application

1.1.1 Add Student Button

- **Add Student:** This Function will allow the user to add new students into the database. After pressing the button the information sheet should be open



Fill in Student Information (Press Ok When Complete)
* Must Be Filled In

* Student Name: Insert Name Here

Type Of Student

Select Course

Select Course Number Before Entering Data!

* Course: Insert Course Name

* Credits: Insert Number Of Credits (Integer (

* Professor: Insert Professor Name

* Cost: Insert Cost (Integer ONLY)

Grade: Insert Course Grade (Integer ONLY,

Information Sheet

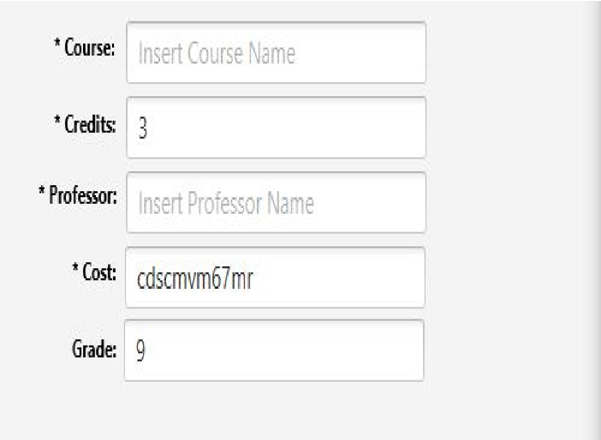
- **Functionality:** Users can now add information into the information sheet so that the information can be put into the database. However there are certain things to keep in mind to make sure that the information gets properly inputted into the database

The Rules to Follow for Inputting Data are:

1. **Credits Must Be A Integer And Not a Double**
2. **Cost Must Be A Integer And Not a Double**
3. **Grade Must Be A Integer And Not a Double**

Failure to do so will result in the inability for the students information to be properly saved.

**** Note: These three fields are permitted to be strings however what will happen is that the system will filter out every non numeric character in the string**



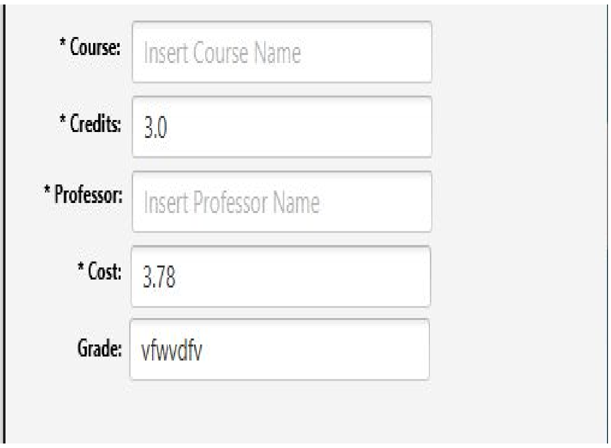
* Course:	Insert Course Name
* Credits:	3
* Professor:	Insert Professor Name
* Cost:	cdscmvm67mr
Grade:	9

<u>Saved Values</u>

Credits: 3

Cost: 67

Grade: 9



* Course:	Insert Course Name
* Credits:	3.0
* Professor:	Insert Professor Name
* Cost:	3.78
Grade:	vfwvdfv

<u>Saved Values</u>

Credits: 30

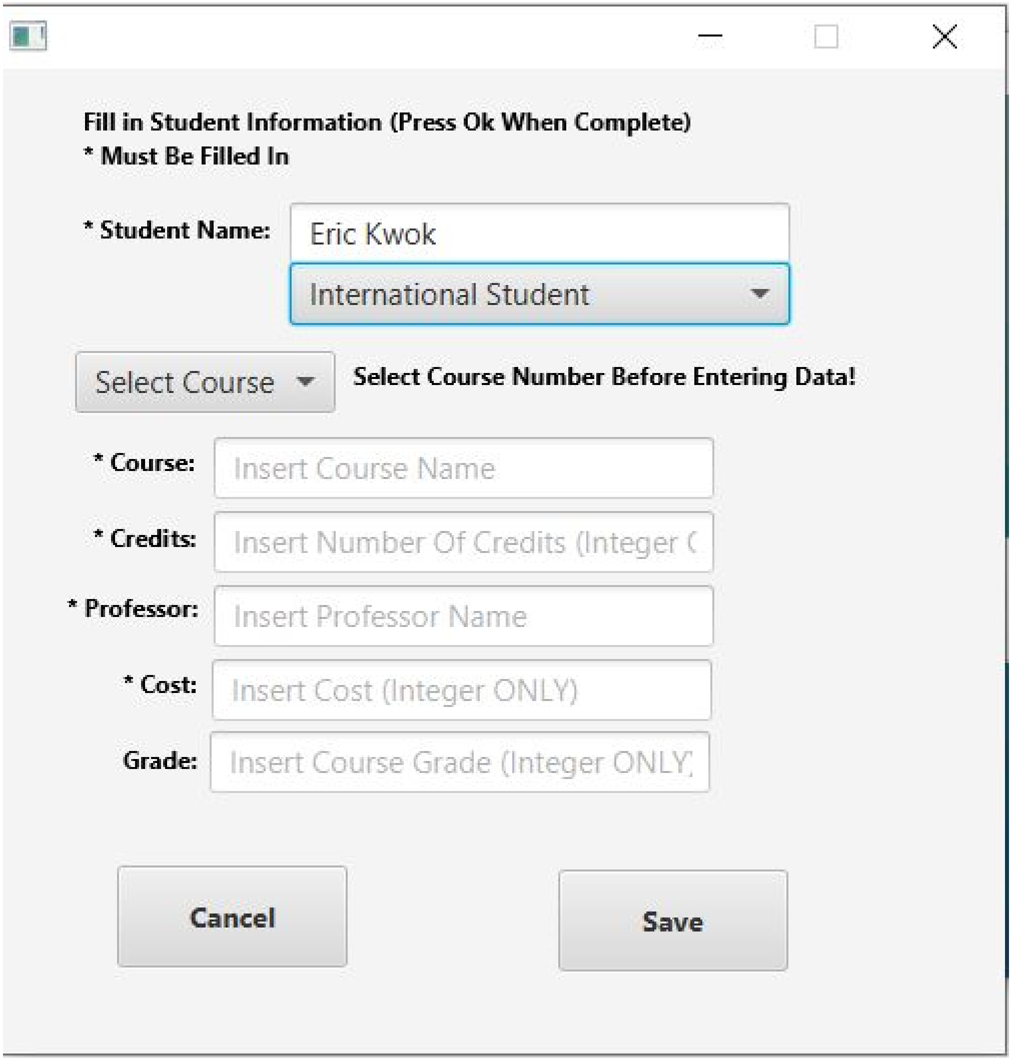
Cost: 378

Grade: 0

- **Functionality (Continued):** For a information sheet to be valid enough for a student to be inputted into the database, the user must input the minimum requirement in order to do so

There are two situations in which the user has inputted enough information for the student to be valid:

Situation #1: User has Inputted a student name and specified what type of student he/she is



Fill in Student Information (Press Ok When Complete)
* Must Be Filled In

* Student Name: Eric Kwok
International Student

Select Course — Select Course Number Before Entering Data!

* Course: Insert Course Name

* Credits: Insert Number Of Credits (Integer (

* Professor: Insert Professor Name

* Cost: Insert Cost (Integer ONLY)

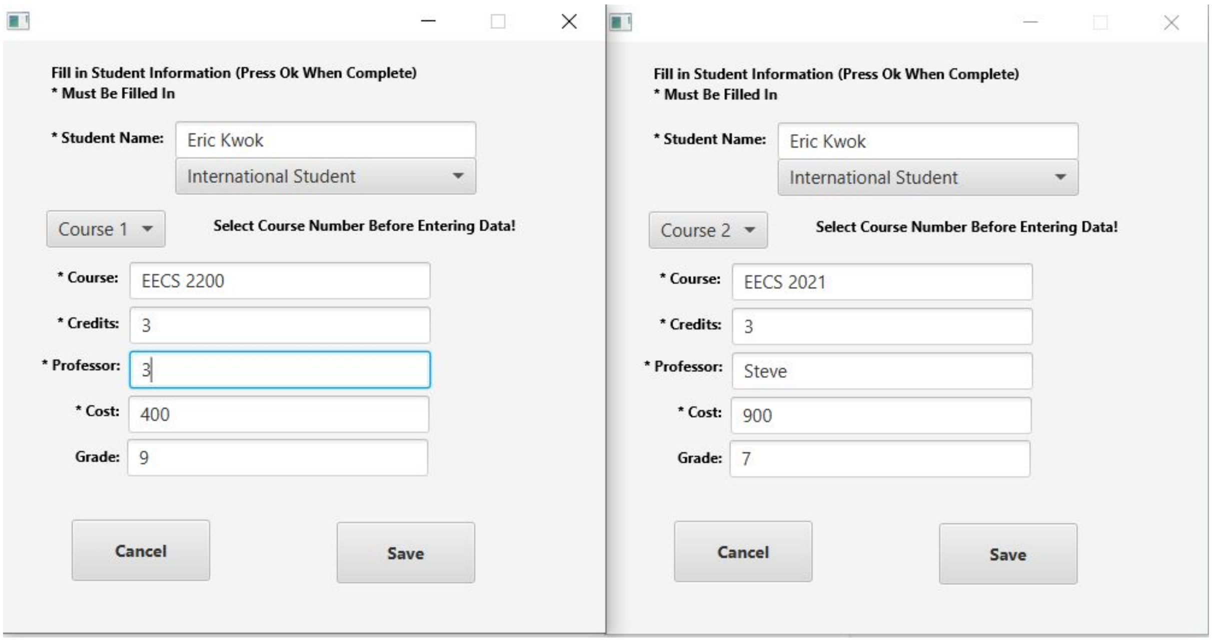
Grade: Insert Course Grade (Integer ONLY,

Valid Entry

Situation #2: User has satisfied **Situation #1** and has inputted information for up to 6 courses with each course having sufficient information put in for it to be valid

Note: When Filling in information for a course make sure to choose the course number before doing so or else the information will not be saved**

Note: If course information is inputted into a course number and it is valid then its state will be saved and the user can safely go to fill in other course numbers and go back to edit one they already filled in whenever they want**



Fill in Student Information (Press Ok When Complete)
* Must Be Filled In

* Student Name: Eric Kwok
International Student

Course 1
Select Course Number Before Entering Data!

* Course: EECS 2200
* Credits: 3
* Professor: 3
* Cost: 400
Grade: 9

Fill in Student Information (Press Ok When Complete)
* Must Be Filled In

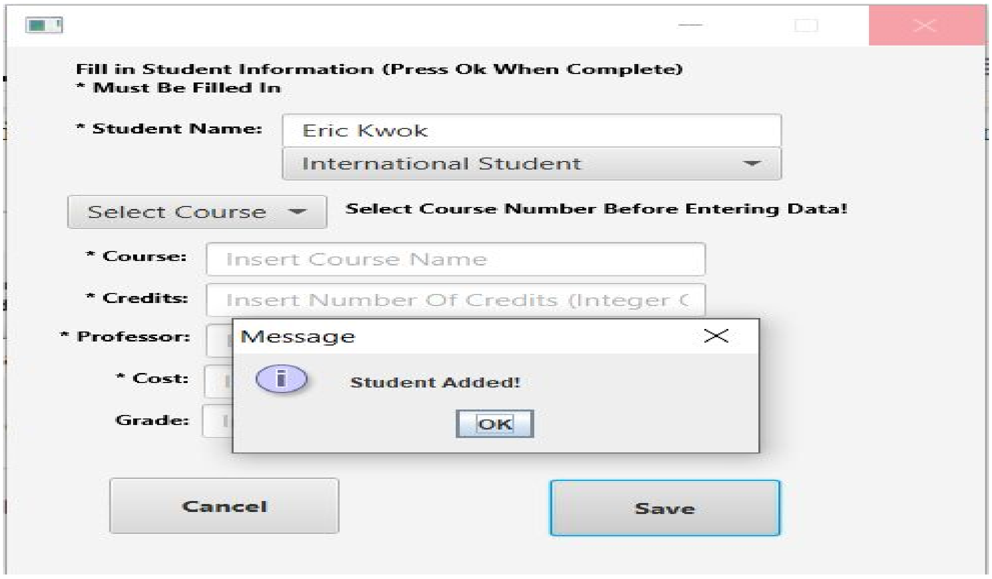
* Student Name: Eric Kwok
International Student

Course 2
Select Course Number Before Entering Data!

* Course: EECS 2021
* Credits: 3
* Professor: Steve
* Cost: 900
Grade: 7

Valid Entry Example (Same Window)

- **Save:** If the user enters a valid entry then pressing the save button will put the student information into the database and if the information is not valid then it will alert the user that it is not a valid entry



Fill in Student Information (Press Ok When Complete)
* Must Be Filled In

* Student Name: Eric Kwok

International Student

Select Course Number Before Entering Data!

* Course: Insert Course Name

* Credits: Insert Number Of Credits (Integer (

* Professor:

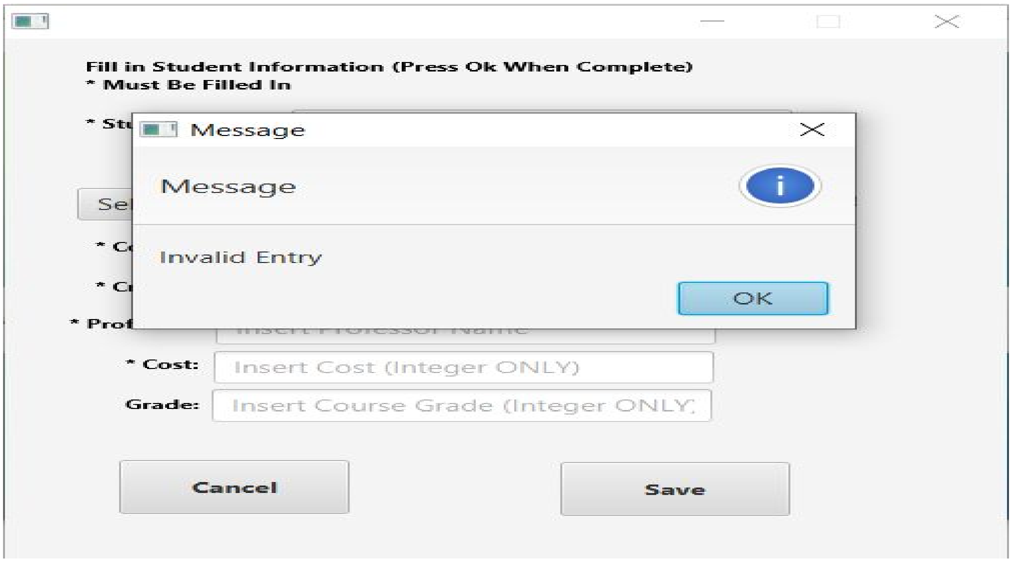
* Cost:

Grade:

Message

Student Added!

Valid Entry Saved



Fill in Student Information (Press Ok When Complete)
* Must Be Filled In

Message

Message

Invalid Entry

* Cost: Insert Cost (Integer ONLY)

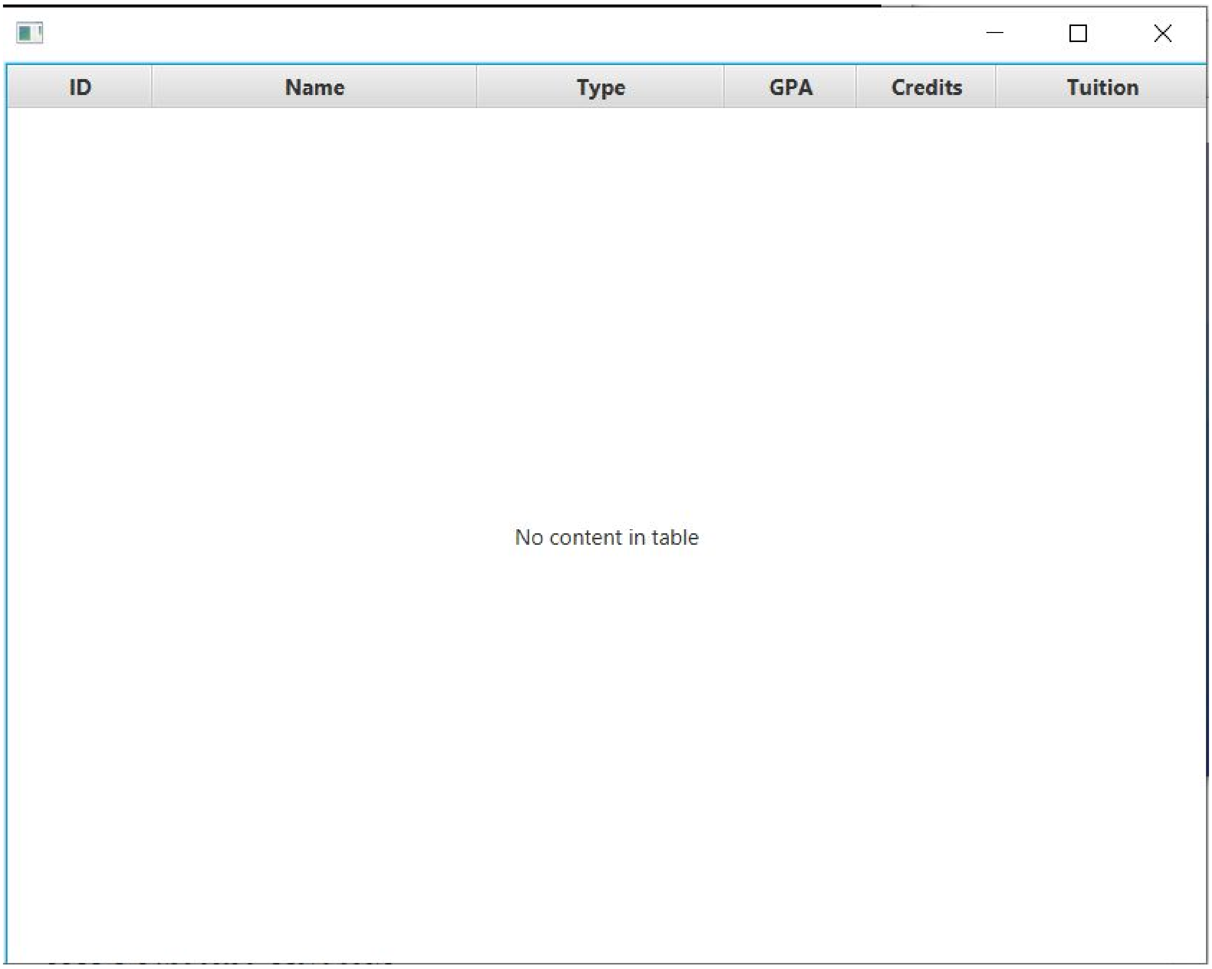
Grade: Insert Course Grade (Integer ONLY)

Invalid Entry Not Saved

- **Cancel:** If the User decides to terminate the current information sheet they can press the cancel button and the window will close with nothing saved

1.1.2 View Students Button

- **View Student:** This Function will allow the user to view all students into the database. After pressing the button the table view should be open

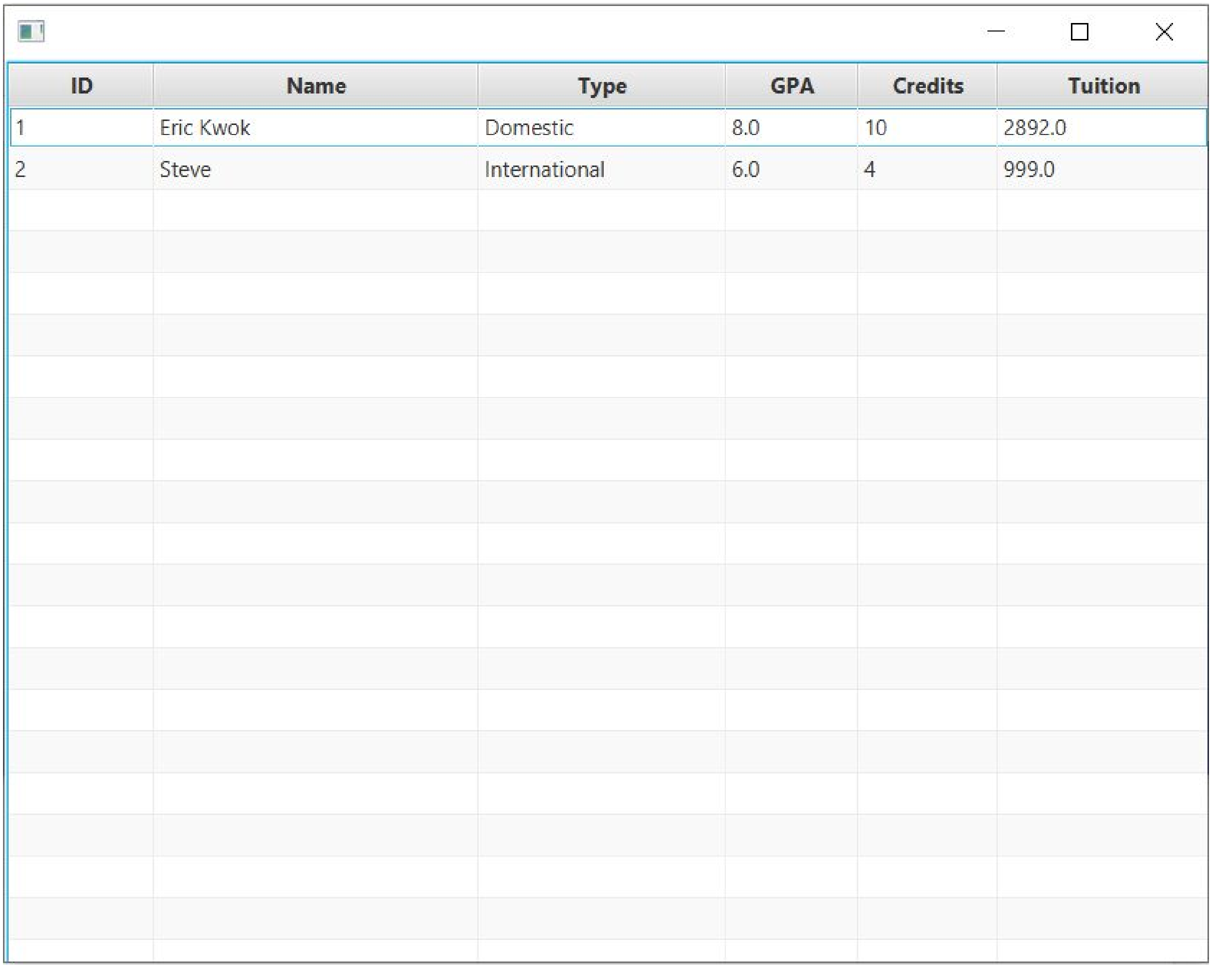


ID	Name	Type	GPA	Credits	Tuition

No content in table

Table View

- **Functionality:** Users can now view their entered students on this table with statistics such as GPA, Tuition, and Total Credits which are calculated based on course information when adding the student



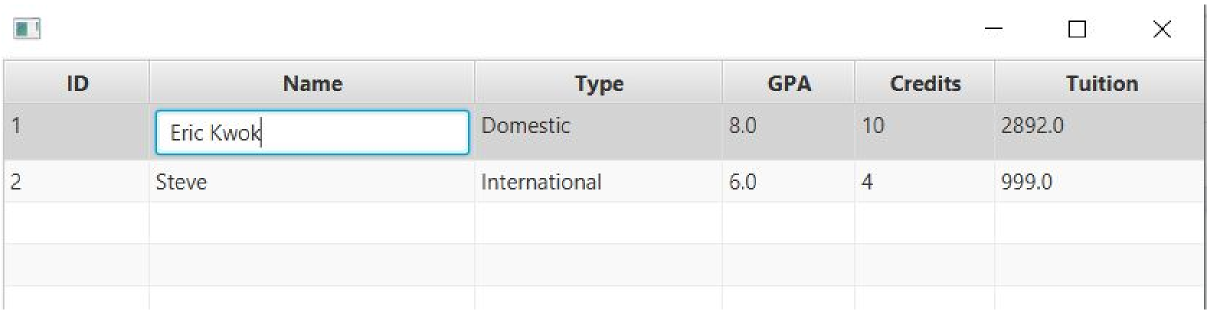
ID	Name	Type	GPA	Credits	Tuition
1	Eric Kwok	Domestic	8.0	10	2892.0
2	Steve	International	6.0	4	999.0

Table Filled From Information Based on Inputted Student Data

- **Functionality (Continued):** Users can also edit information in the case that they would like to change the students data. After doing this the students information will be updated in real time and even if the table is closed and pressed again the updated change will show. The ID is the only field that is **Immutable** as that is for the system to keep track of the students.

Note: When Editing information on the table make sure that you only change the field to the specific type it's displaying. For example you can only change the GPA field with another number, not a word**

Note: The Type field can only be changed to be "International" or "Domestic" and other change will not be saved when closing the table**

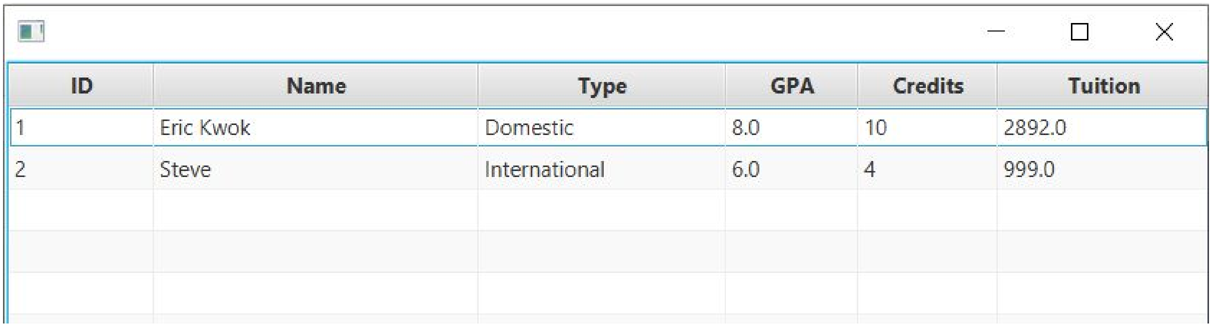


ID	Name	Type	GPA	Credits	Tuition
1	Eric Kwok	Domestic	8.0	10	2892.0
2	Steve	International	6.0	4	999.0

Editing the Name

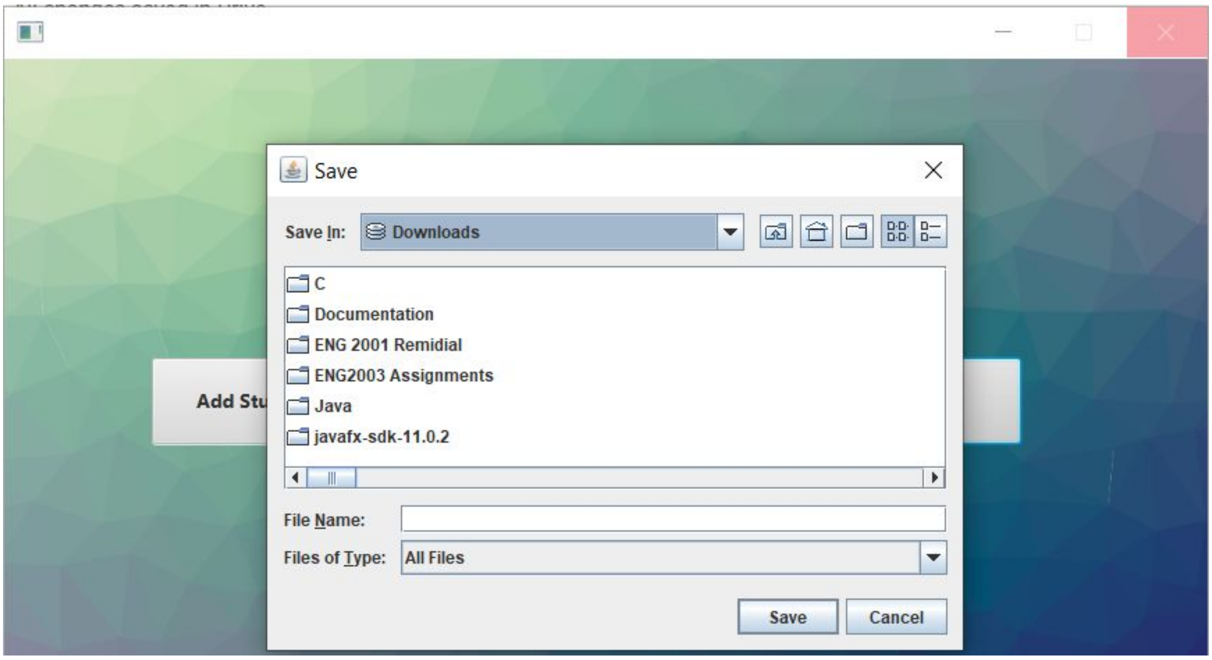
1.1.3 Export Button

- **Export:** This Function will allow the user to export their information into a csv. This can only be done if students exist within the system and if the file is a valid xlsx file or else an error will occur.



ID	Name	Type	GPA	Credits	Tuition
1	Eric Kwok	Domestic	8.0	10	2892.0
2	Steve	International	6.0	4	999.0

Current Entries in the Application



Save

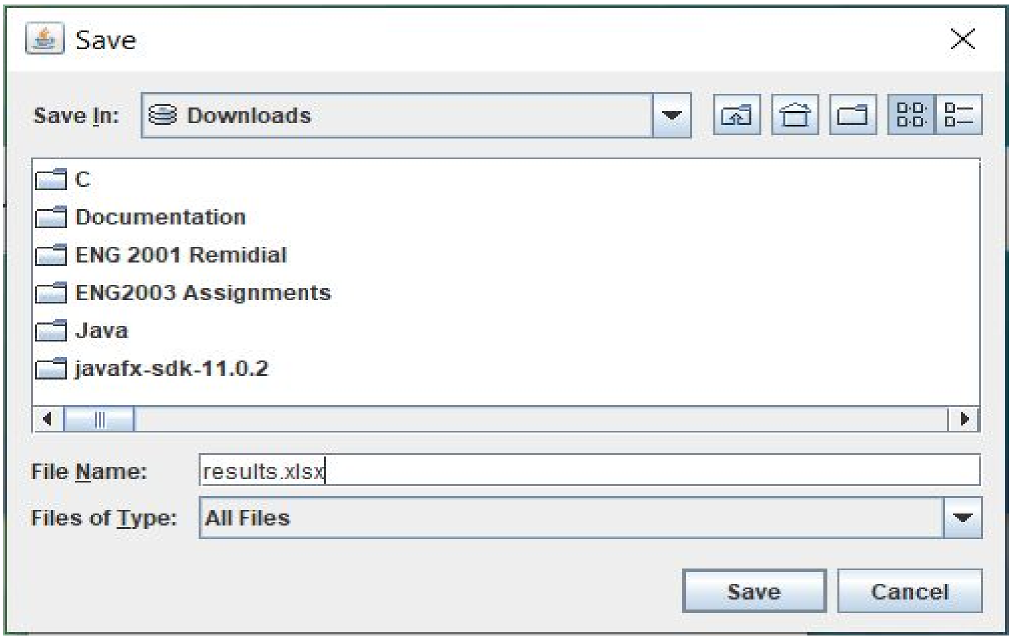
Save In: Downloads

C
Documentation
ENG 2001 Remidial
ENG2003 Assignments
Java
javafx-sdk-11.0.2

File Name:

Files of Type: All Files

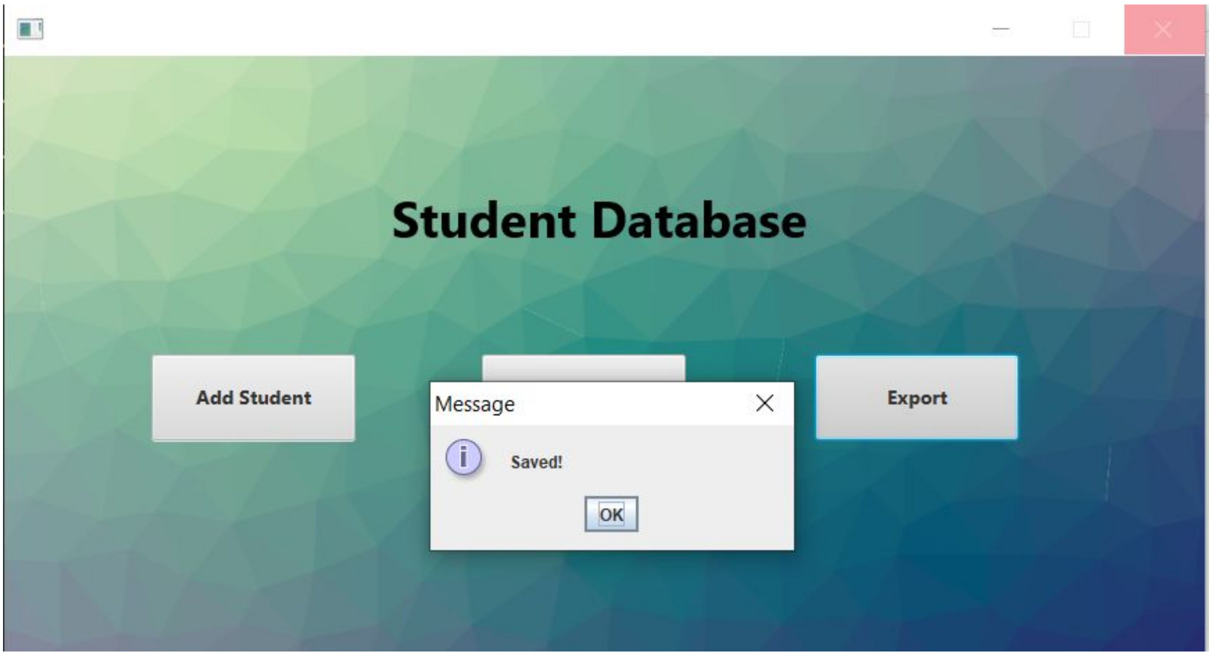
After Pressing Export Button



File Name: results.xlsx

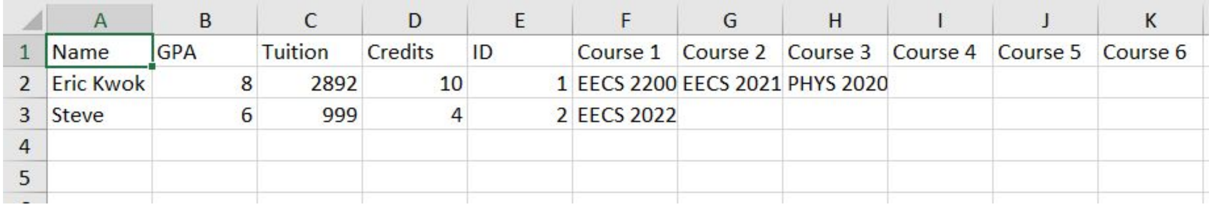
Files of Type: All Files

Type In A Valid .xlsx File



Saved!

Entries Were Successfully Saved



	A	B	C	D	E	F	G	H	I	J	K
1	Name	GPA	Tuition	Credits	ID	Course 1	Course 2	Course 3	Course 4	Course 5	Course 6
2	Eric Kwok	8	2892	10	1	EECS 2200	EECS 2021	PHYS 2020			
3	Steve	6	999	4	2	EECS 2022					
4											
5											

View Information For Use