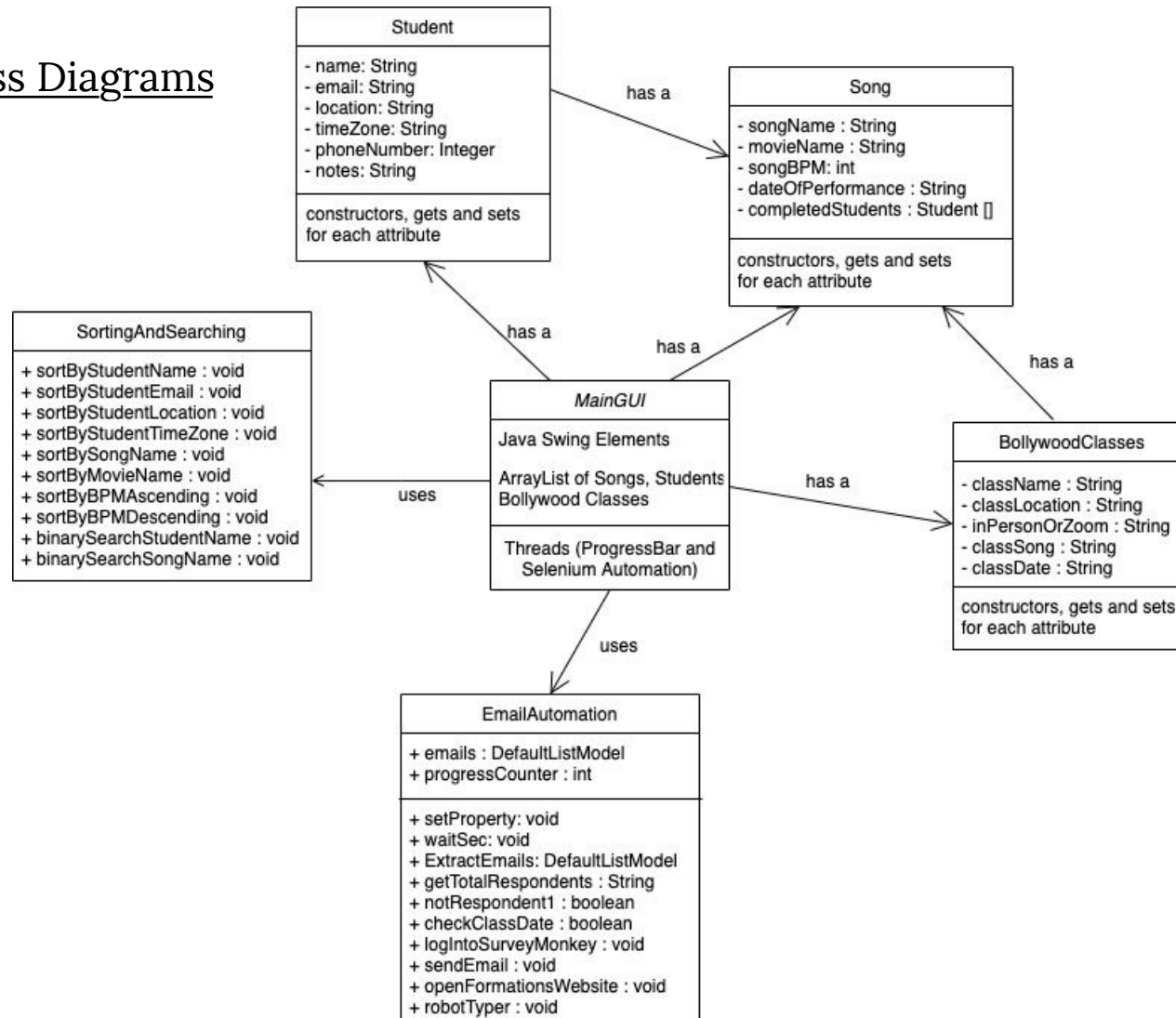


# Criterion B: Bollywood Database Design Overview


## Class Diagrams



## All About Inputs

Input	Data Type	Normal Range etc	Normal Example(s)	Border	Abnormal	Extreme
Student Name	String	Not left blank; normal names of 2 or more characters - recommended to be the first name or first and last name, <b>but the first name first, always</b>	Jerry Shoenberg  Emily	Short names like Jo  Including middle names: Lara Maddison Bower	Numbers or other extraneous characters  <i>Warning message, if field, not filled-&gt; Name must be entered</i>	Very long names of 30+ characters  No limit
Student Email	String	Not left blank; normal emails	<a href="mailto:emily@hotmail.com">emily@hotmail.com</a>  arnoldpalmer02@gmail.com	Short emails like da@gmail.com	<a href="mailto:john@rayworth.info">john@rayworth.info</a>  <i>Display "double-check" message and re-entry if does not end with ".com", or blank</i>	N/A
Student Location	String	Can be left blank, although should be filled.	California  Portland, Oregon  Uzbekistan	N/A	Numbers or other extraneous characters	Very long address  No limit to length
Time Zone (GMT)	String	Can be left blank, although should be filled  Should start in "GMT" for better sorting, though not required by user	GMT+7  GMT+3	N/A	PT+3  <i>Warning message to re-enter time zone</i>	N/A
Phone Number	Integer	Can be left blank, although should be filled and starts with <b>country</b>	669 373 87983  972 028 37490	0860044840  <i>Warning message and</i>	03726345  <i>Warning message and</i>	672938400821723  <i>Warning message</i>

		<b>code</b> - should not start with "0"		the possibility for re-entry, "did you forget to add the country code?"	re-entry for numbers shorter than 7 characters	and re-entry for numbers longer than 15 characters
Student Notes	String	A string of any length. Can be left blank	Blonde hair used to live in China  Beginner; still learning basic moves  N/A	N/A	N/A	N/A
Song Name	String	Not left blank; Normal song names of 2 or more characters	Sweetheart  Trunk Jhalla Wallah	N/A	N/A	Very long song names of 30+ characters  Warning message and re-entry "check your song name input"
Movie Name (year)	String	Can be left blank, but is not recommended: The main movie where the song originated from  Optional to enter the year the movie was published as well	Doom (2005)  Chennai Express	N/A	N/A	Very long movie names of 30+ characters  Warning message and re-entry "check your song name input"
Song BPM	Integer	Can be left blank, but must be an integer between 40 and 240	76  162	238  42	Anything that is not an integer  73.4	N/A

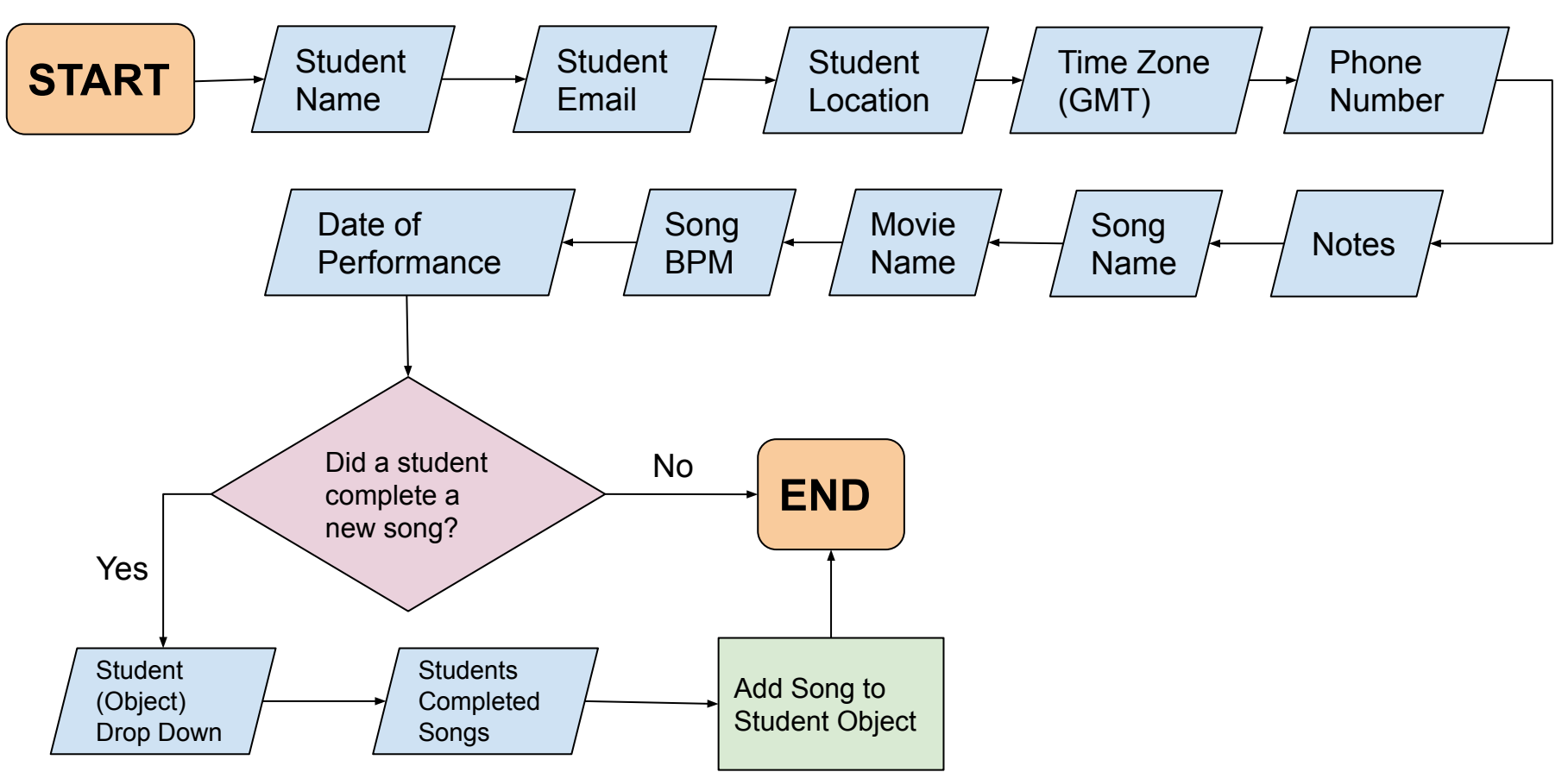
					Very fast song  Warning message and re-entry if not an integer or between 40 and 240	
Date of Performance	Date (object) - can be converted toString()	Inputted via the text field- MUST follow the format: (Month) (Date), (Year) for example, <u>February 21, 2021</u> . The font color will be GREEN if input is correct, but RED if incorrect.  Optionally use -> to select the date   Can be left blank	12/11/21  13/03/2021	N/A	3/11/21  31/31/21  February 21, 2021  Warning message and re-entry for anything that does not follow the format of DD/MM/YY(YY)	N/A
Songs that a Student Have Completed	Linked List	Dynamic - ONLY songs that have already been inputted into the program; selected from a drop-down menu	It's Time to Disco, Yes Sir I Can Boogie, Dum Dum...	N/A	N/A	N/A
Class Name	String	Any string of any length. Preferably under 5 words. Can be left blank.	Bollywood #3  Advanced Only Class	N/A	N/A	N/A
Class Location	String	Must be filled - Any string of any length.	Home Dance Room	N/A	Numbers or special characters	Very long locations of 5+ words.

		Preferably under 5 words.	Nichada Dance Room Secret Garden		No warning message	Warning message and re-entry; "are you sure"
Class Song	String	Must be filled - Choose a song from a drop-down menu including all "Song Names"	Nashe Si Chadh Gayi Garam Masala	N/A	N/A	N/A
Session is In-Person	Boolean	Checkbox: Selected or Unselected (both In-Person and On-Zoom can be selected simultaneously)	True	N/A	Both are not-selected.	N/A
The session is On Zoom	Boolean		False		Warning message and re-entry: "one must be selected"	
Class Date	String	Must follow format of:  Month (short/long) + Day, Year  Can be manually written in or selected using the JCalendar popup. Text will be green if a valid input	Jan 21, 2021 Aug 06, 2021	N/A	Feb 13  Warning message and re-entry for anything that does not follow the format of (Month + Day, Year)	N/A
Additional Emails	String	Can be blank... ie, no additional emails added. Valid email addresses only.	<a href="mailto:baba@gmail.com">baba@gmail.com</a> <a href="mailto:lexica-pog@yahoo.com">lexica-pog@yahoo.com</a>	Short emails like da@gmail.com	<a href="mailto:john@rayworth.info">john@rayworth.info</a>  Display "double-check" message and re-entry if does not end with ".com"	N/A
Subject of	String	Limit of 50 characters -	Canceling tomorrows	N/A	N/A	N/A

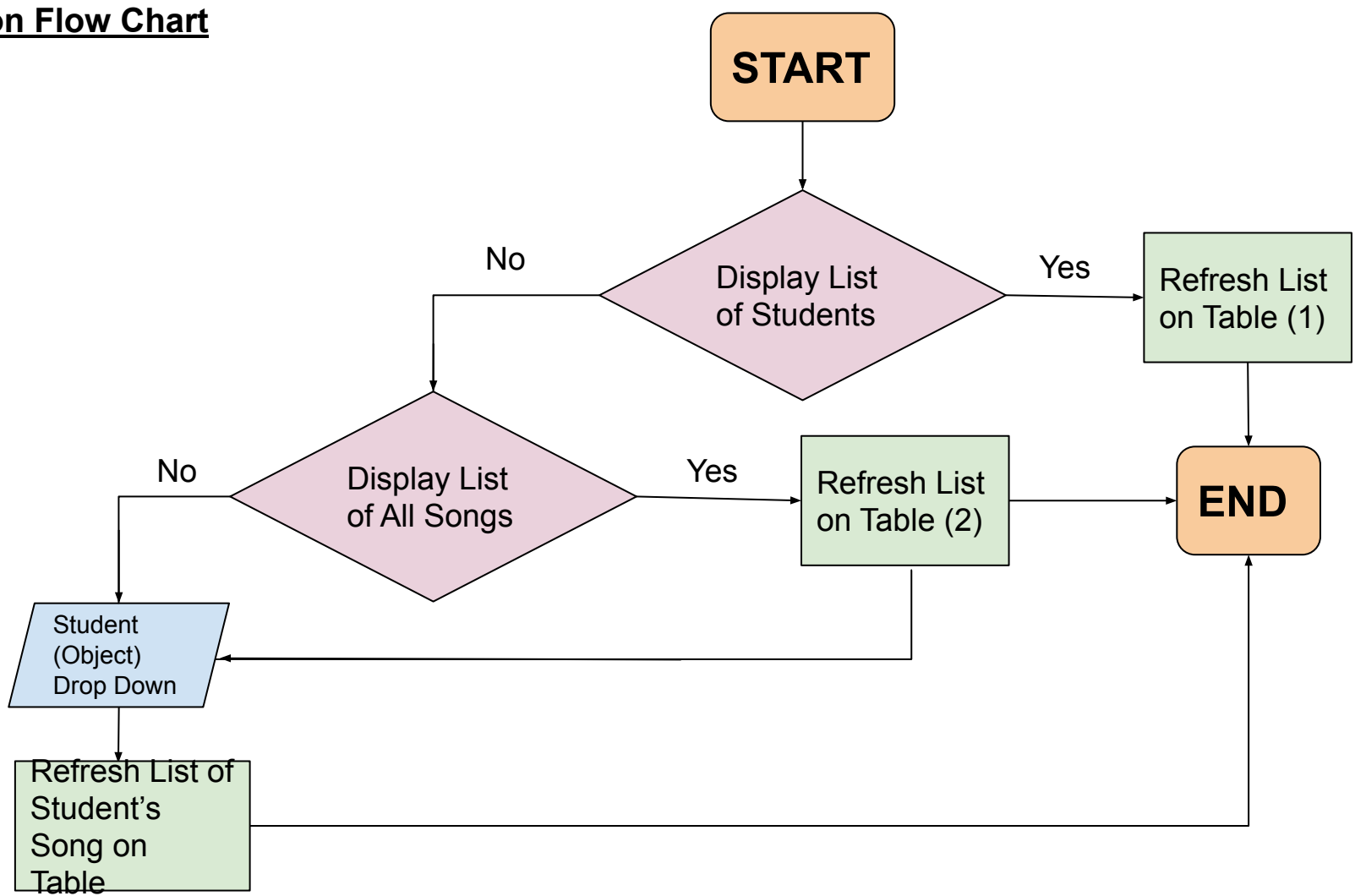
Email		can be any string including relevant and central information	Lesson Zoom Link for Today's Lesson			
Body of Email	String	Limit of 38,400 characters (Gmail limit) - can be any string with any relevant information	Hi Ladies! Something came up and I, unfortunately, cannot do the lesson we planned for tomorrow...  The zoom link for today's class is pasted below:...	N/A	N/A	N/A
Email Username	String	Must be a valid GMAIL address and password associated with that account.	usersemail@gmail.com	N/A		
Email Password	String		ur4567654BB			
# of Students in Existing Dance Formation	Integer	-Any integer that is a realistic number of students. -If it is between 2 and 20, a website option is available -If it is 0, all formations are displayed	0 18 12 3	36  Many students, probably unrealistic but still possible.	172 (never going to have 172 students in one dance formation/performance)  -5 (can't have a negative number of students)  Display warning message when negative, or > 50 students - re-entry necessary	N/A
Dance	2D Array	Must have height and	15 x 15	30x30	1 x 50	10000 x 10000

Formation Grid Size	(array of arrays)	width > 0. Any valid integers that may simulate the size of the performance stage	7 x 10	(Really big, but acceptable)	(Rarely any stages are just a long rectangular shape - formations become limited)  Warning message and re-entry - possibly wrong dimensions (when W is 10 or more times greater than H, or vice versa)	Way too big.  Display warning message when H/W > 50 - re-entry necessary
------------------------	----------------------	--	--------	---------------------------------	--	---

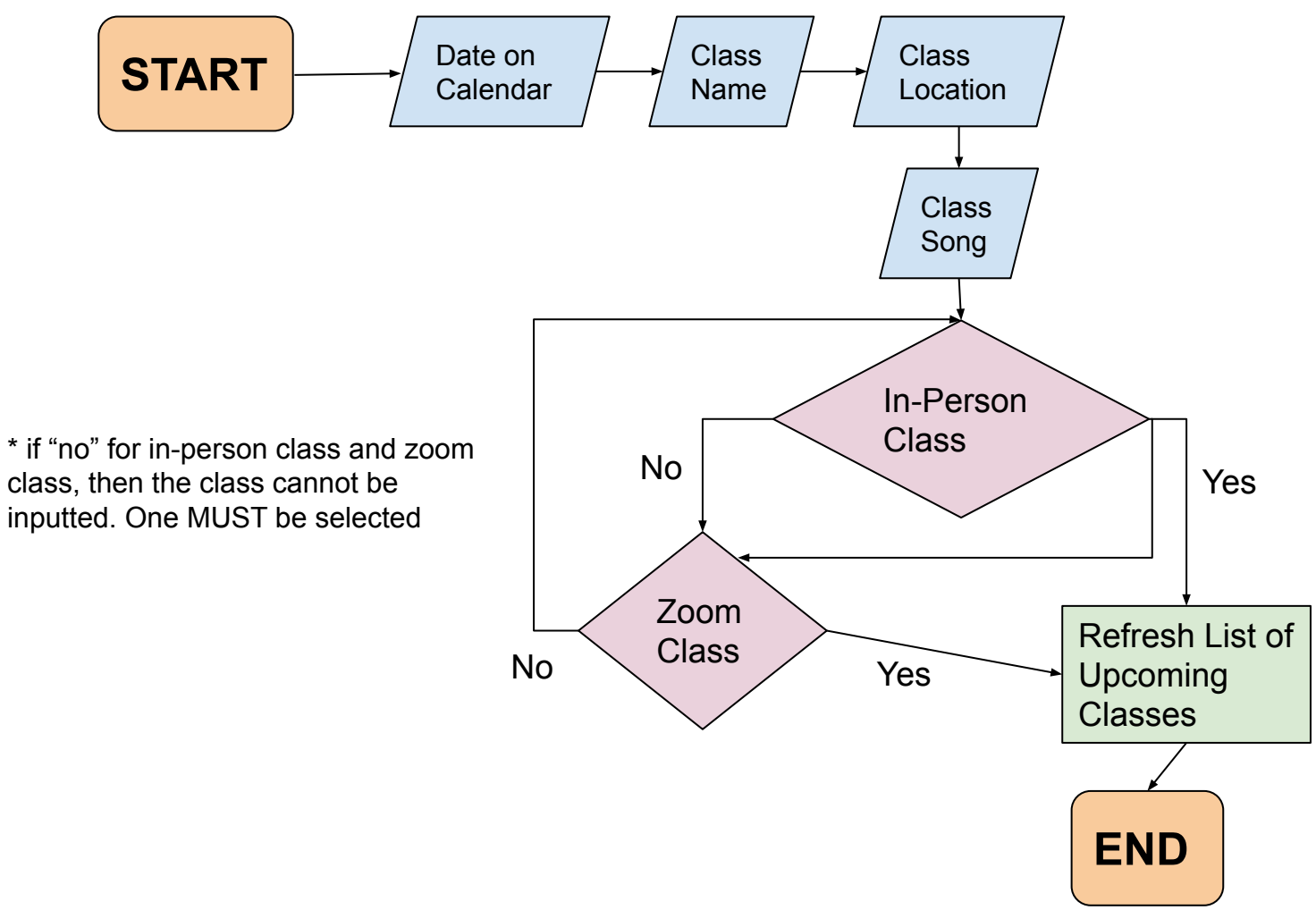
**Inputting Information Tab Flow Chart**



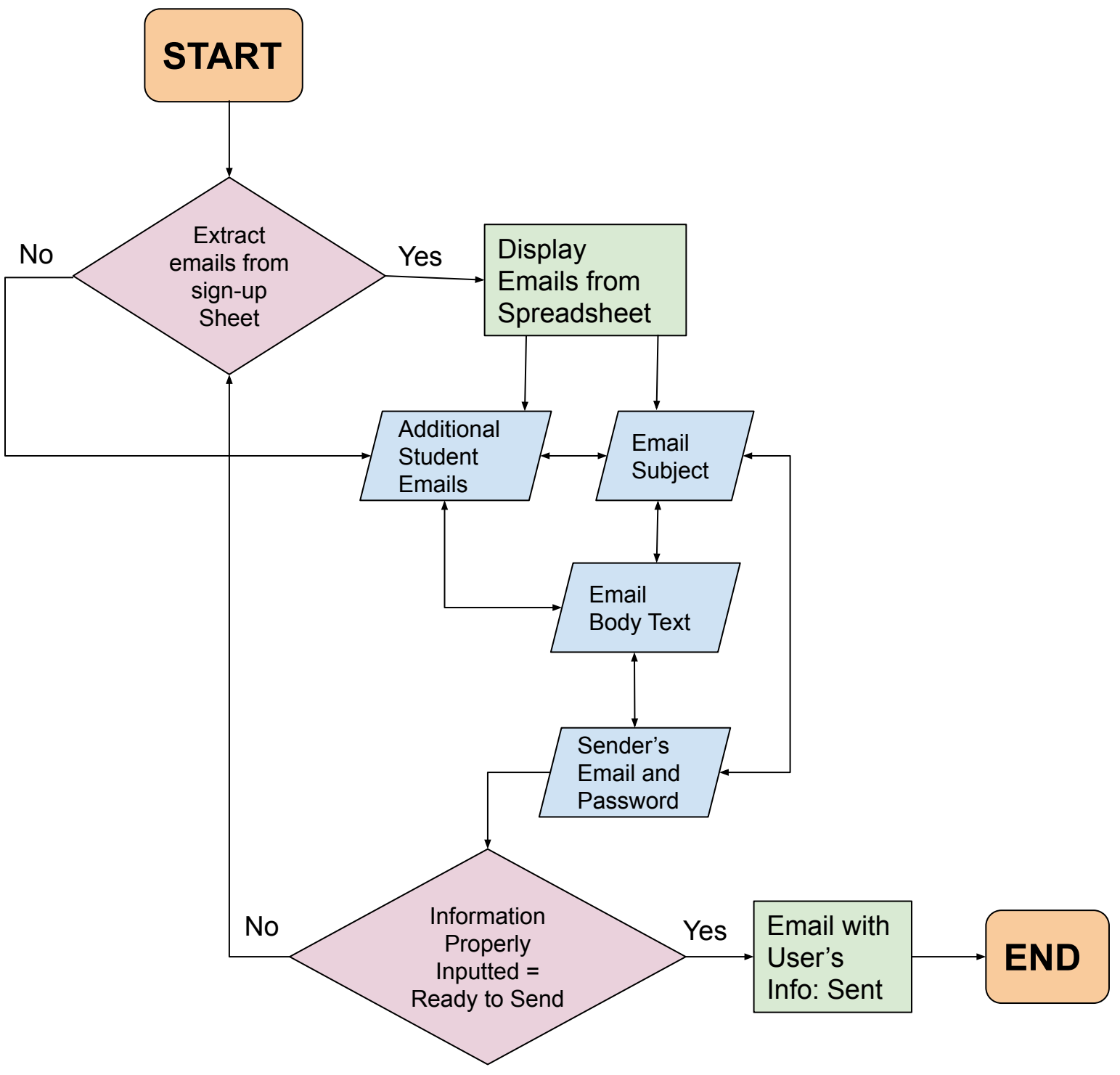
**Displaying Students and Songs Information Flow Chart**



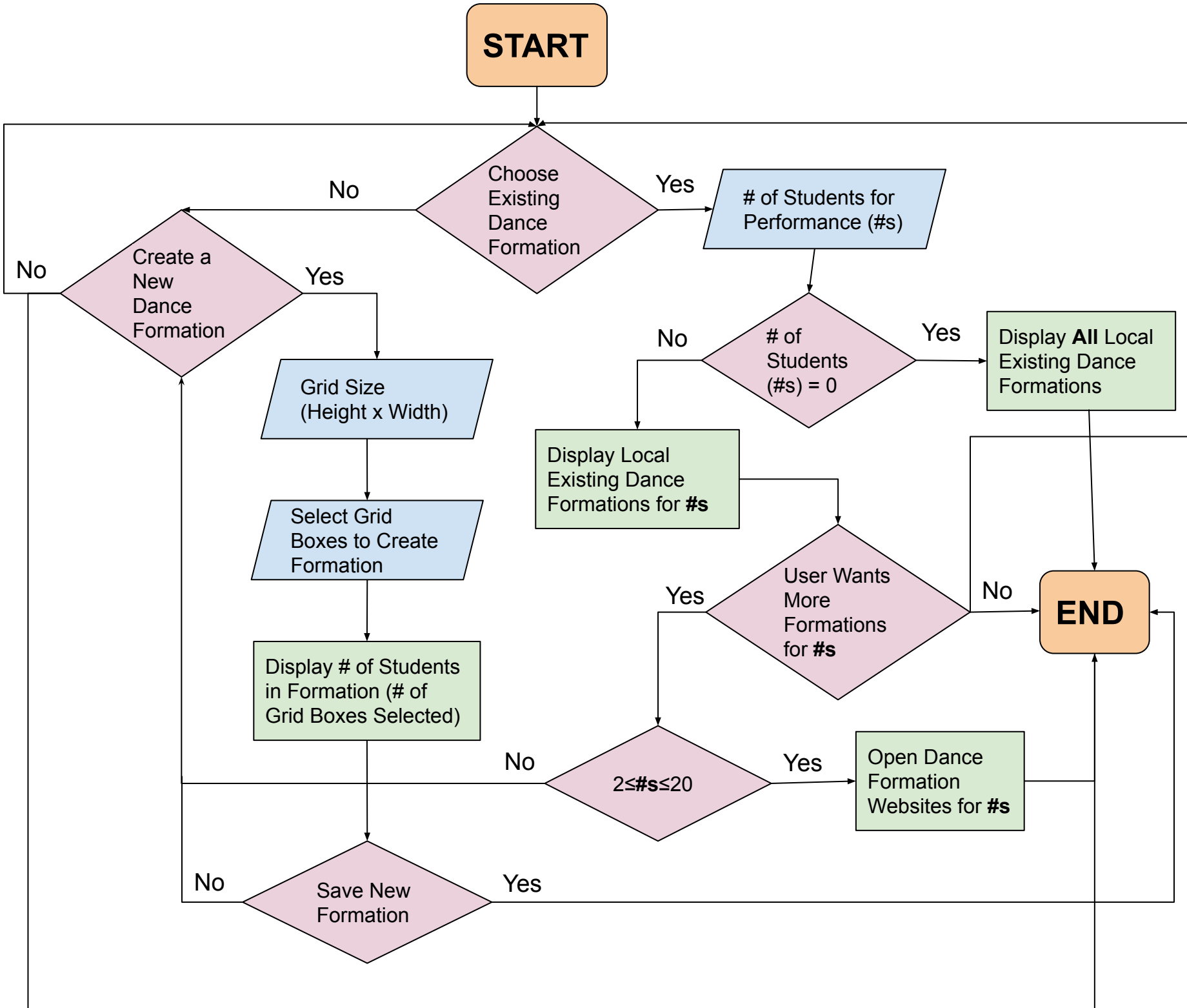
**Calendar Flowchart**



**Automated Email Flow Chart (see Appendix 6 for original sketch, planning, and extra notes)**



**Dance Formations Flow Chart**





# Extracting Emails Method Pseudocode

```
WebDriver d = new ChromeDriver()
driver.get(surveyLink)
logIntoSurveyMonkey(username, password)
waitSec(15)
getTotalRespondents()
MainGUI.mainGUI.threadProgressBar.start()
while (not Respondent1) {
  if (correct class Date) {
    String emailAnswer = driver.findElement(By.xpath(...))
    emails.addElement(emailAnswer)
    progressCounter++
    driver.goToNextRespondent
  }
}
setProgressBar(max)
return emails
```

making new instance of chrome driver for use of webdriver automation.

opens link on chrome

give time for web elements to load.

returns int with total respondents

↳ starts progress bar thread which constantly checks for updates on respondent #...

→ while not last respondent...

→ check class date matching on survey

↓  
finds respondent email and adds to DLM; used for display on mainGUI

→ to update progressbar

→ done extracting emails

# Adding a Student who completed a song → Pseudo

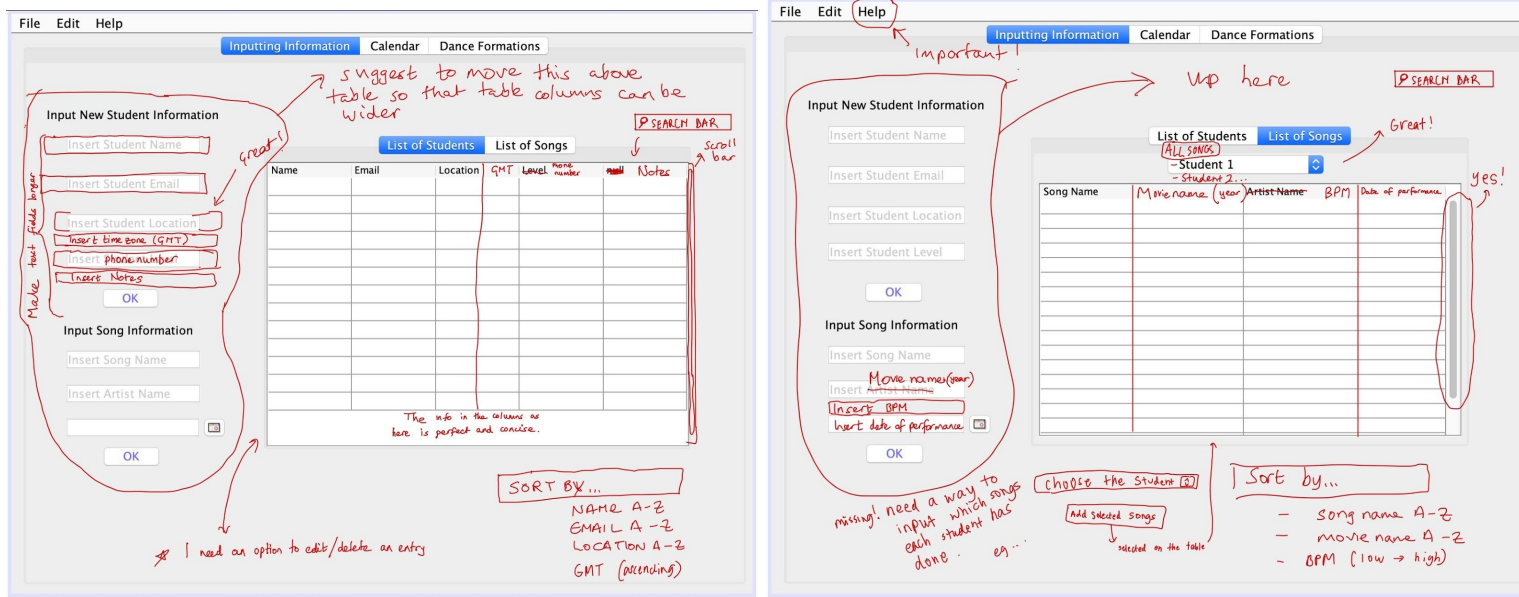
```
for (i = 0, i < students.size(); i++) {
    if (students.get(i).getName().equals(AddStudentToSong.comboBox)) {
        if (ListofSongsT.getSelectedRow() == -1) {
            Display Error message POP-up → using JOptionPane.showMessageDialog
        } else {
            int s = ListofSongsT.getSelectedRow()
            songs.get(s).addCompletedStudents(students.get(i), s)
        }
    }
}
```

→ loop through all students to find corresponding one to combo box

→ no selected row

using this method in Song class, the student is added to the selected song → ready for display on the "List of Students" tab

# Testing Plan (on edits of first prototype)



## Things Done on this Tab

### - Inputs

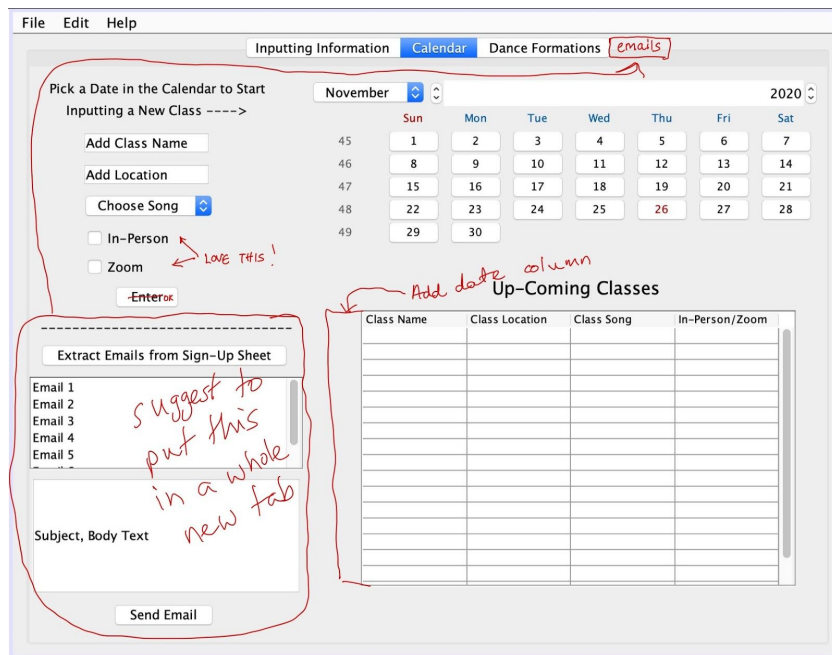
- Bollywood student name, student email is mandatory to input by the user
- Student location, student time zone, student phone number, and student notes are optional but suggested
- The Bollywood song can be inputted by the user writing the song name, movie name with the year of the release date, song's beats per minute (BPM), date of performance where the song will be performed.
- A standard sort and search which can be chosen by the user
- While on the "list of songs" tab, the user will be able to add an existing song to a Student class, certifying their completion
  - Then, the user can display all songs inputted or a specific Student's completed songs on the table.

### - Internal Functioning

- Pressing each of the “ok” buttons will create a new instance of a Student class or Song class, with the respective class attributes inputted by the user.
- New instances of students and songs will be added and displayed on the table
- When adding a selected song to the Student, a Linked List of songs will be added as one of the Students attributes.

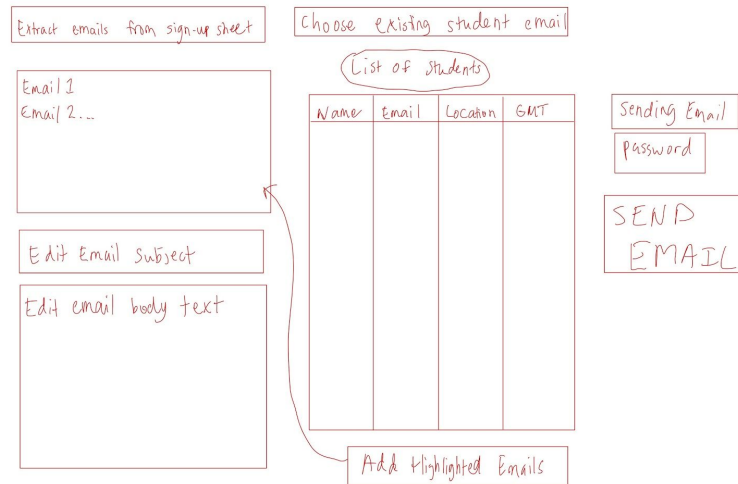
- **Other Notes**

- The tables will “automatically” refresh once a new Student/Song instance is created/inputted-however the user can refresh the table without needing to input new information by clicking on the tab headers again.
- Opening the “help” tab will direct the user to an explanation on how to use the inputting information tab.
- The user will also be able to edit/delete information on the tab by changing it directly on the respective table, and then clicking “enter” or clicking on the tab headers again.



## Things Done on this Tab

- **Inputs**
  - Class date, class name, class location, class song, and class details (in-person/online zoom) are inputs.
- **Internal Functioning**
  - Pressing the “ok” button will create a new instance of an upcoming class including the respective inputted attributes.
  - Up-Coming Classes table will refresh automatically after a new input
- **Other Notes**
  - The user must choose a date on the calendar before also having to input the class location, song, details, while the class name is optional.



## Things Done on this Tab

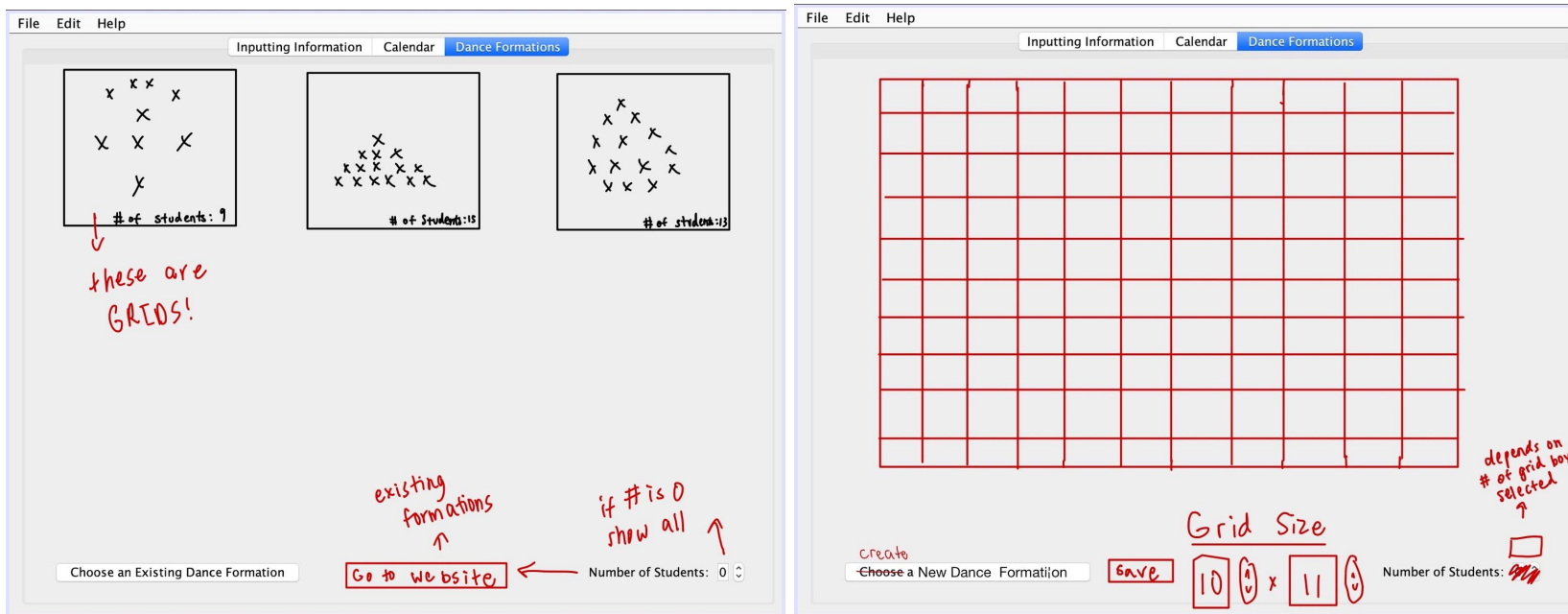
### - Inputs

- The user may extract emails from an online spreadsheet, which the users students should sign up for classes on a Google form; filling the information out on the spreadsheet
  - These emails can be edited/deleted, and more emails can be added manually by selecting the student on the table
- At least one of the email subject and email body text must be filled in. Sending email and password must be filled in as well.

### - Internal Functioning

- Extracting emails takes several steps for the program, and will require external imported libraries from Java's Selenium (web automation library).
  - Major steps are... open up Chrome Driver, send cookies to log in to the email account, go to spreadsheet, copy inputted emails from a spreadsheet, paste on the program.

- Most likely will run “headless”, meaning this automated action will happen in the background without the Chrome GUI interfering.
- **Other Notes**
  - The user must have ALL the Selenium libraries installed and Wifi (to send the email) so that this part of the program does not instill a plethora of errors.



## Things Done on this Tab

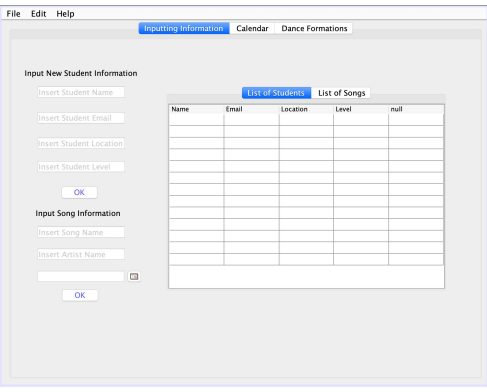
- **Inputs**
  - The user will initially choose whether they want to create a new dance formation or choose an existing one.
  - Creating a new dance formation, the user will input the grid size (height x width) and select the appropriate grid boxes.

- In order to save their creation to existing formations, the Save button must be clicked.
- The user can choose an existing dance formation by inputting their chosen number of students- 0 students mean showing all available formations.
- Existing dance formations include the ones previously inputted by the user, or “default” formations from websites online.
- **Internal Functioning**
  - Choosing existing formations from a website will open up a non-headless Chrome driver, going to a link that is previously coded into the program for the ease and convenience of the user.
  - The grid and chosen boxes will be saved - the non-empty (not “ ”) elements of the 2D array are grid selections by the user.
  - The grid 2D arrays will be saved on a Linked List (since the list is a dynamic structure)

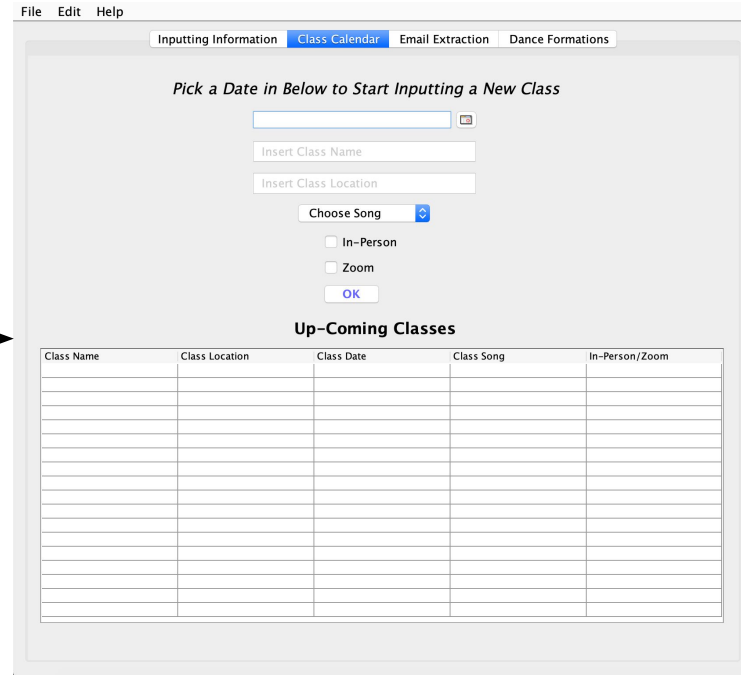
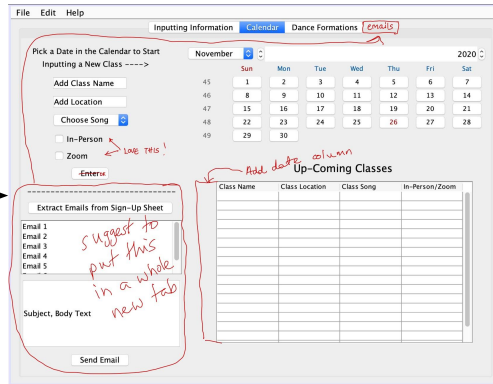
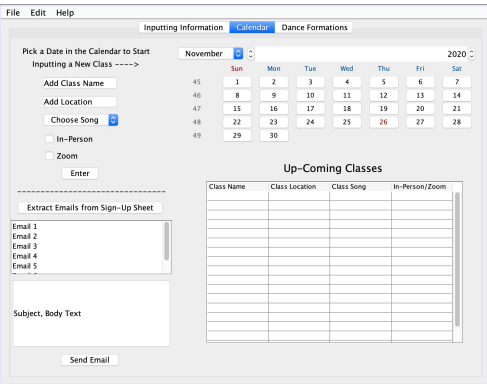
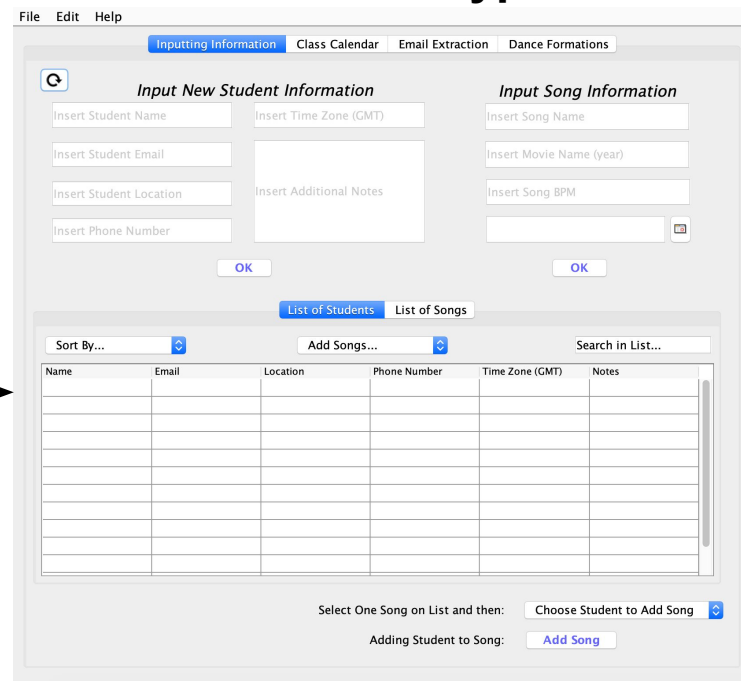
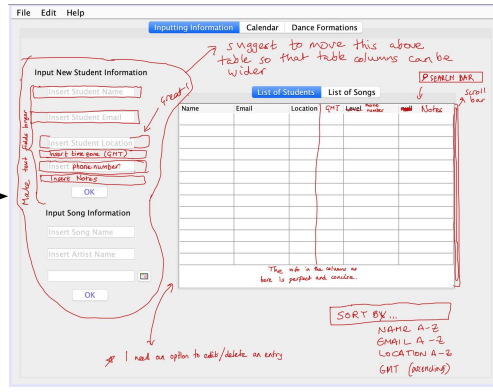


# Final Prototype

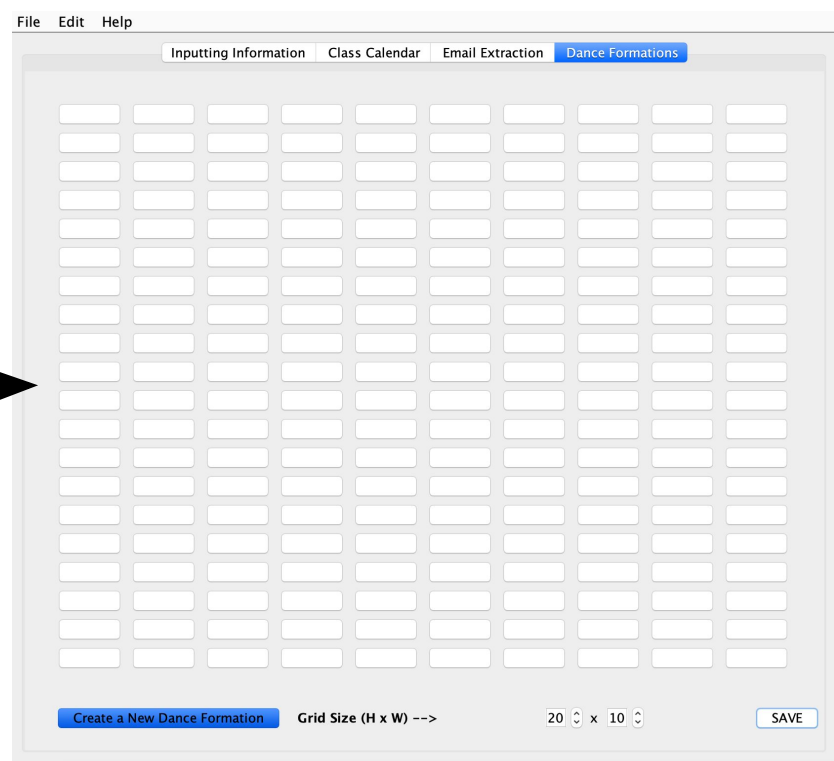
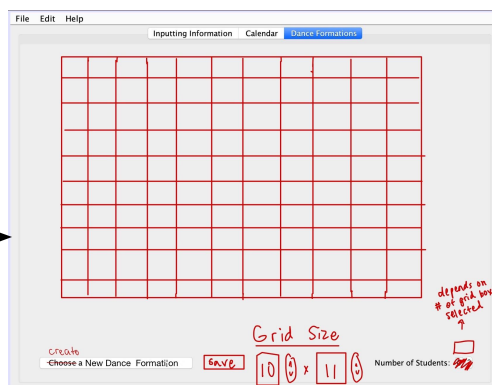
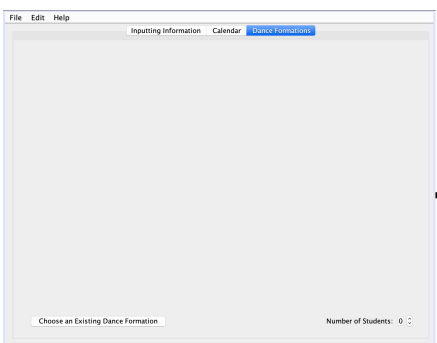
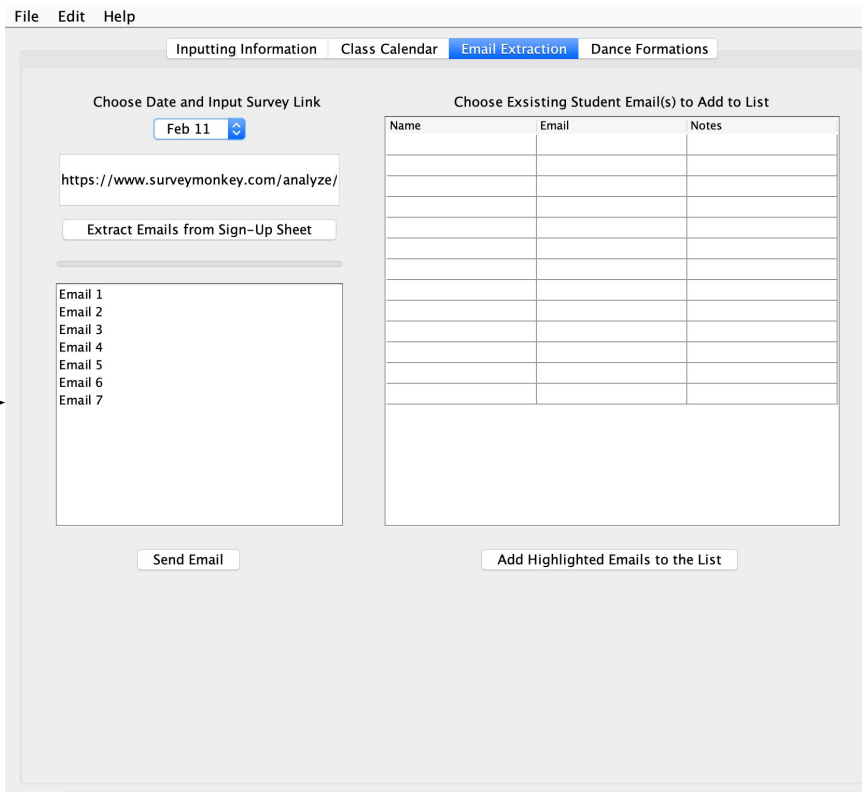
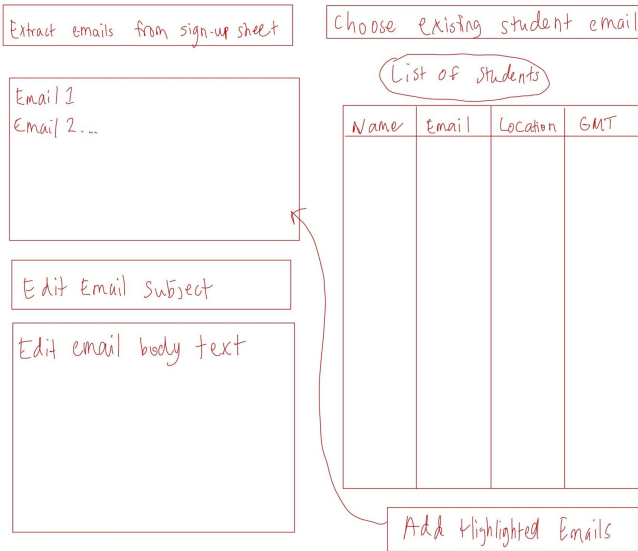
## First Prototype



## Revisions on First Prototype



File Edit Help



### Initial Prototyping Process Reflection (Crit A)

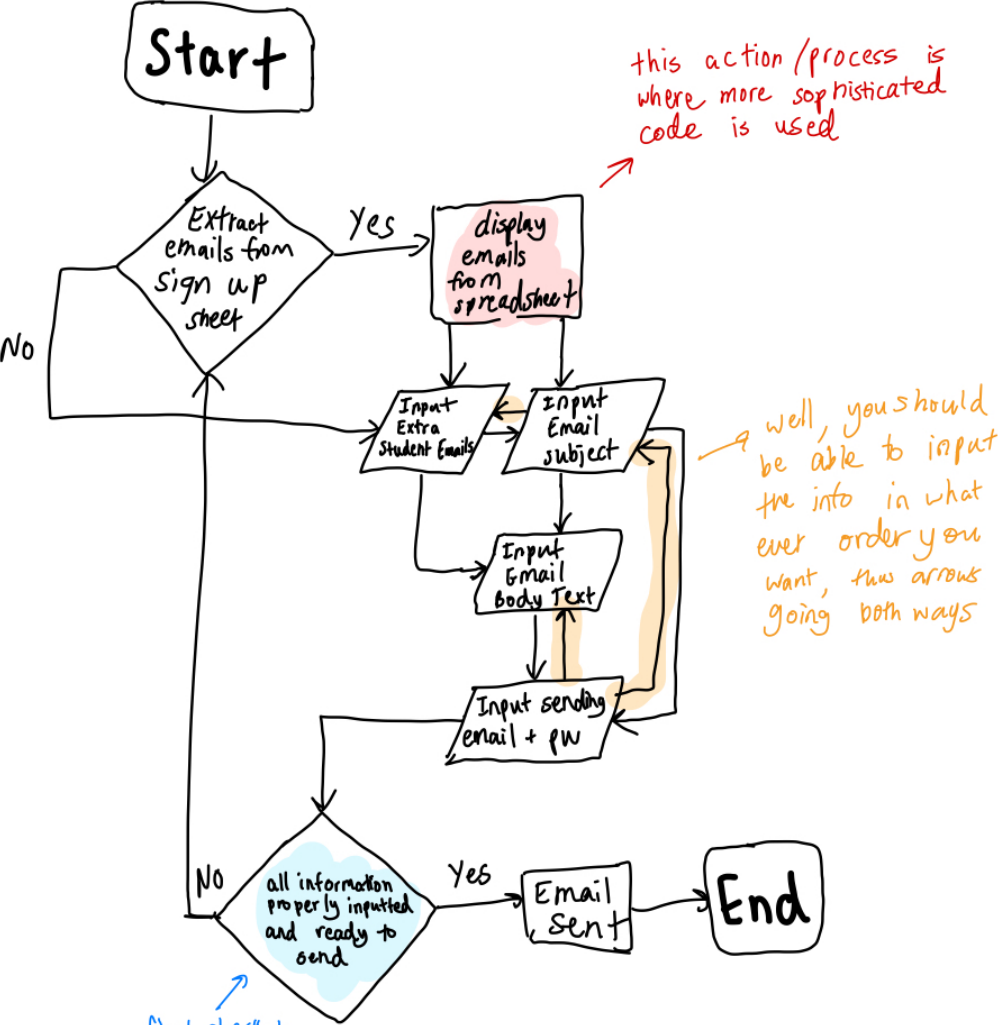
### Final Prototyping Process Reflection (Crit B)

The first edits on the GUI prototype were essential to get a better understanding of the relevant information my client wanted to be stored within the database, the intended look and feel of the GUI, and the actions and tasks that can be accomplished. The main changes included:

The final prototype shows off all the missing usability features in the other prototype, expanding on the requests and feedback from the client. Primarily, the first two tabs have a much more sophisticated, cleaner, and uniform look with the inputs being on top of the table and each element having a clearly outlined purpose. The option to change the class date and survey link (used for email extraction) has been added, as well as a progress bar since the Selenium web automation may take time and that would help with notifying the client of the progress of that task. Finally, the dance formation tab now has a resizable grid with each gridbox being a selectable toggle button- almost identical to the edits proposed by the client on the initial prototype

1. Moving the inputs of the information above the table rather than to the side, allowing wider and more spacious displaying columns.
2. Adding a sorting and searching option within the first tabbed panel
3. Creating a more detailed version of the way in which the user will be able to create and access dance formations from the respective tab, as well as extract emails from an online sign-up sheet.

# Appendix 6: Flow Chart Sketch



this action/process is where more sophisticated code is used

well, you should be able to input the info in what ever order you want, thus arrows going both ways

final check to see there is at the very least either body/subject of email  
↳ has to have senders email and password in order to log-in.