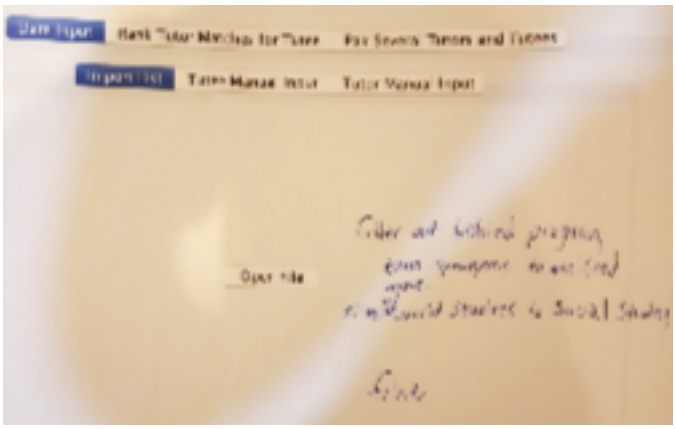
Criterion B - Solution Overview

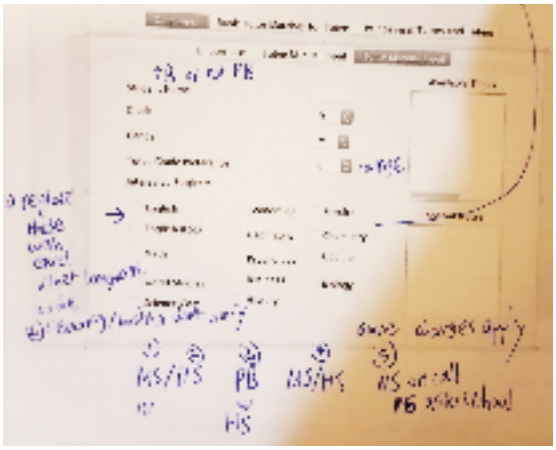
Prototyping Process

After the second interview, several changes were made to the initial prototype. First of all, to improve usability, I made my program better complement Mr.Callahan's spreadsheet, for example, using Mr.Callahan's subject labels. Secondly, I improved the design of the available time input box for visual appeal and thirdly and most importantly, added a 'Change Data' tab to allow Mr.Callahan to edit previously input data, which I forgot in the initial prototype. Lastly, I made some minor but still significant changes, such as adding a 'pair' button in the 'Rank Tutor Match for Tutees' Tab and an 'available' checkbox in the 'Change Data' tab to make the paired or unavailable tutees and tutors not appear when generating future matches.

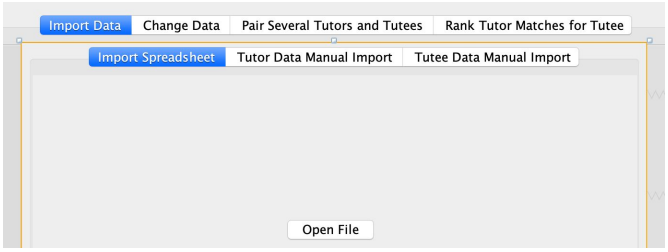
Word Count: 123

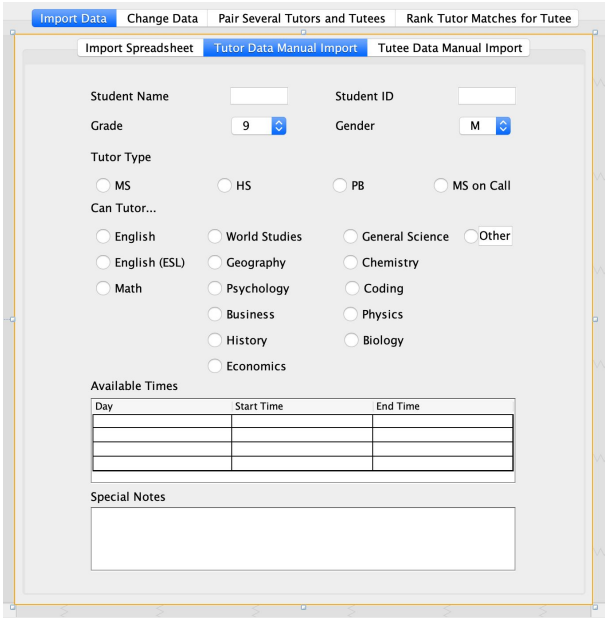
Initial Prototype

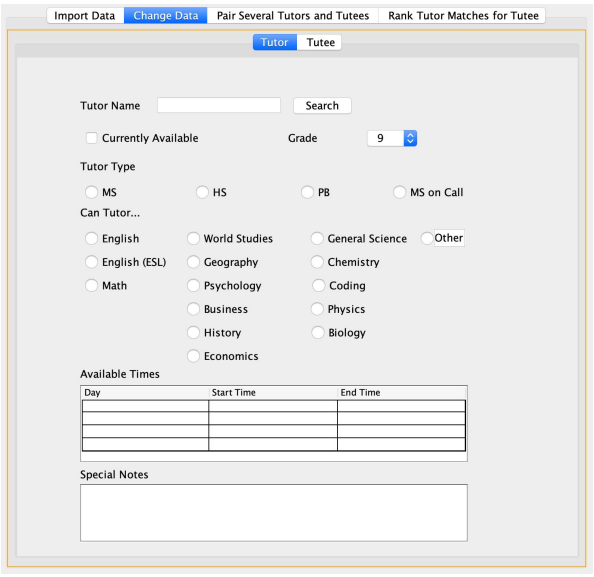


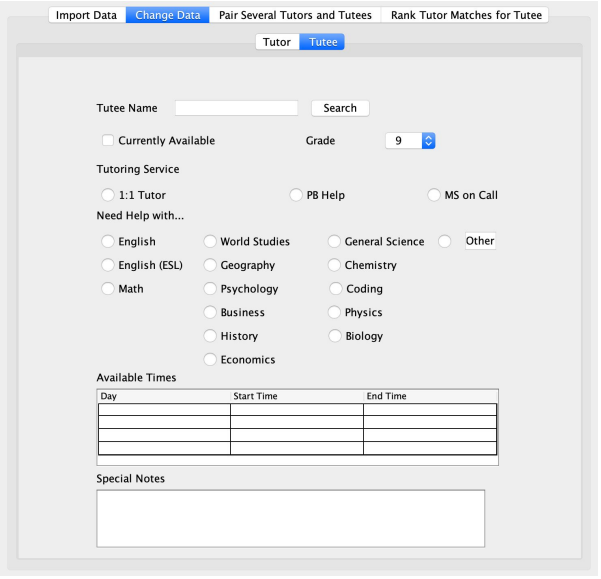


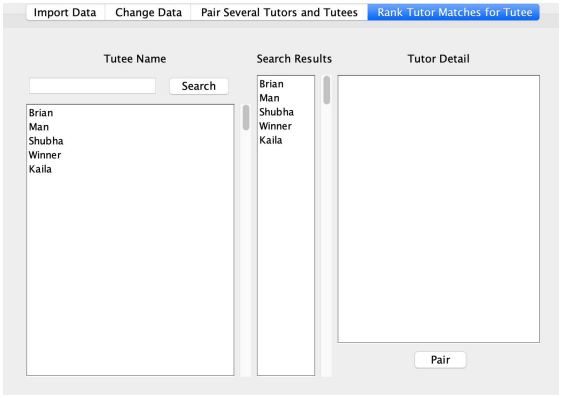
Final Prototype






Import Data
Change Data
Pair Several Tutors and Tutees
Rank Tutor Matches for Tutee
Tutor
Tutee
Tutor Name
Search
Currently Available
Grade
9
Tutor Type
MS
HS
PB
MS on Call
Can Tutor...
English
World Studies
General Science
Other
English (ESL)
Geography
Chemistry
Math
Psychology
Coding
Business
Physics
History
Biology
Economics
Available Times
Day
Start Time
End Time
Special Notes


Import Data
Change Data
Pair Several Tutors and Tutees
Rank Tutor Matches for Tutee
Tutor
Tutee
Tutee Name
Search
Currently Available
Grade
9
Tutoring Service
1:1 Tutor
PB Help
MS on Call
Need Help with...
English
World Studies
General Science
Other
English (ESL)
Geography
Chemistry
Math
Psychology
Coding
Business
Physics
History
Biology
Economics
Available Times
Day
Start Time
End Time
Special Notes


Import Data
Change Data
Pair Several Tutors and Tutees
Rank Tutor Matches for Tutee
Tutee Name
Search Results
Tutor Detail
Search
Brian
Man
Shubha
Winner
Kaila
Brian
Man
Shubha
Winner
Kaila
Pair

Input and Output List

*Due to redundancy, variables from other tabs treated effectively the same were put in the same column

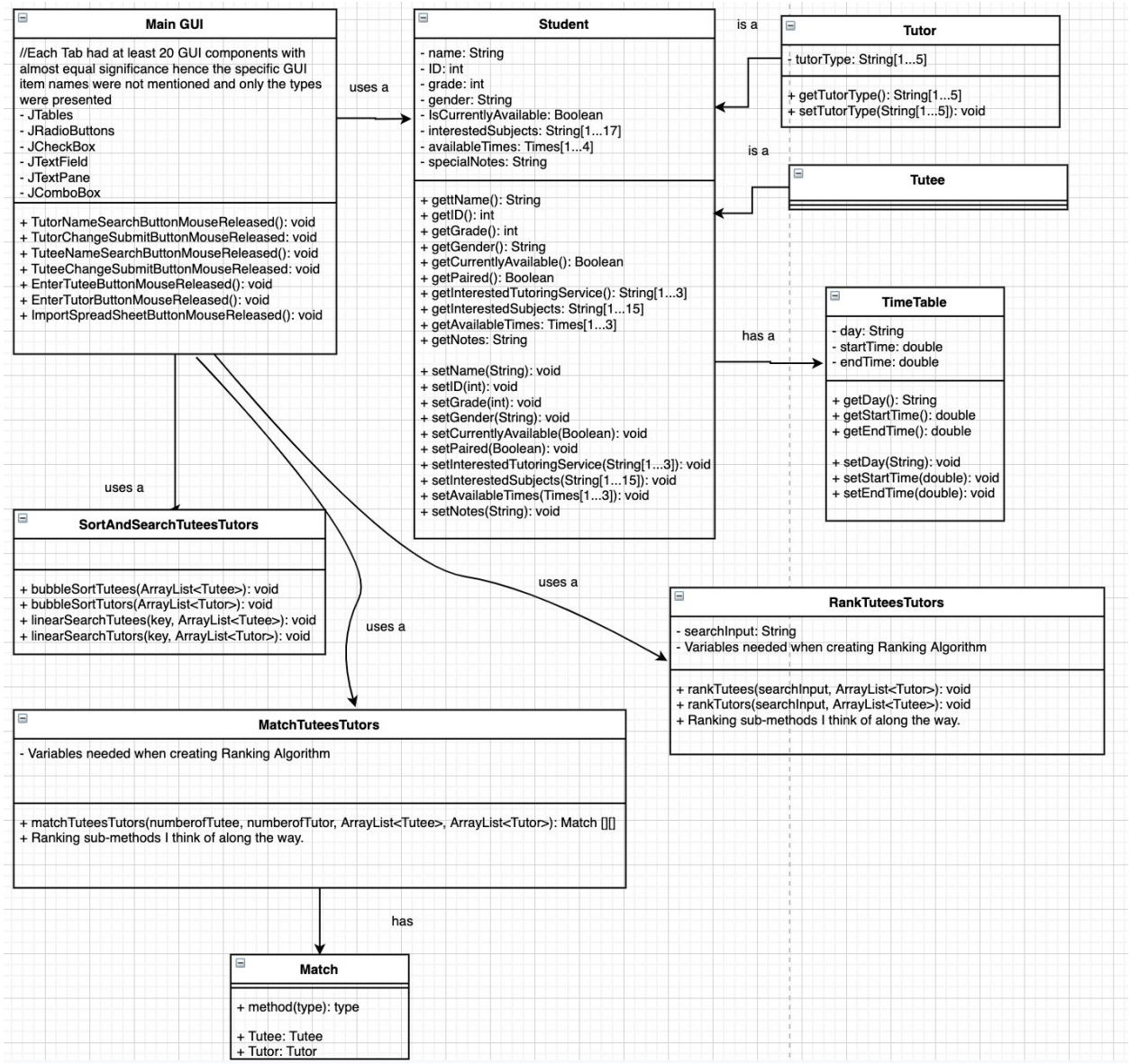
Input	Tab	Data Type	Normal Range etc.	Example
Student Name	Tutor/Tutee Data Manual Input Tutor/Tutee Change Data Rank Tutor Match	String	N/A	Brian
Student ID	Tutor/Tutee Data Manual Input	Int	10000-30000	19285
Grade	Tutor/Tutee Data Manual Input Tutee/Tutor Change Data	Int	9-12 6-12	7
Gender	Tutor/Tutee Data Manual Input	String	M or F	M
Tutor Type	Tutor Data Manual Input Tutee/Tutor Change Data	String Array	N/A	PB, MS on Call
Tutor/Tutee Interested Subjects	Tutor/Tutee Data Manual Input Tutee/Tutor Change Data	String Array	N/A	English, Math
Available Times	Tutor/Tutee Data Manual Input Tutee/Tutor Change Data	TimeTable Array	Mon, Tue, Wed, Thu, or Fri + 00.00~23.59 - 0.00~23.59	Tue 13.00-14.00
Special Notes	Tutor/Tutee Data Manual Input Tutee/Tutor Change Data	String	N/A	N/A
Availability	Tutee/Tutor Change Data	boolean	True or False	True
Number of Tutors	Pair Tutor Tutee	Int	N/A - reasonable max at 100	40
Number of Tutees	Pair Tutor Tutee	Int	N/A- reasonable max at 100	20

Output	Tab	Data Type	Normal Range etc.	Example
Availability	Tutee/Tutor Change Data	Boolean	True or False	True
Student ID	Tutee/Tutor Change Data	Int	10000-30000	19285
Grade	Tutee/Tutor Change Data	Int	9-12 6-12	7
Tutor Type	Tutor Change Data	String Array	N/A	PB, MS on Call
Tutor/Tutee Interested Subjects	Tutor/Tutee Data Manual Input	String Array	N/A	English, Math
Available Times	Tutor/Tutee Data Manual Input	Time Array	Mon, Tue, Wed, Thu, or Fri + 00.00~23.59 - 0.00~23.59	Tue 13.00-14.00
Special Notes	Tutor/Tutee Data Manual Input	String	N/A	N/A
Tutee Name	Pair Tutor Tutee	String	N/A	N/A
Tutor Name	Pair Tutor Tutee	String	N/A	N/A
Meeting Time	Pair Tutor Tutee	Time Array	Mon, Tue, Wed, Thu, or Fri + 00.00~23.59 - 0.00~23.59	Tue 13.00-14.00
Tutor Detail	Rank Tutor Tutee	String	N/A	Name: Brian Grade: 12 Gender: M Tutoring Service: PB Available Times: Fri 2.00-3.00 Interested Subjects: Math

				Special Notes: Experienced Tutor

Word Count: 17 (Table excluded)

Class Diagrams

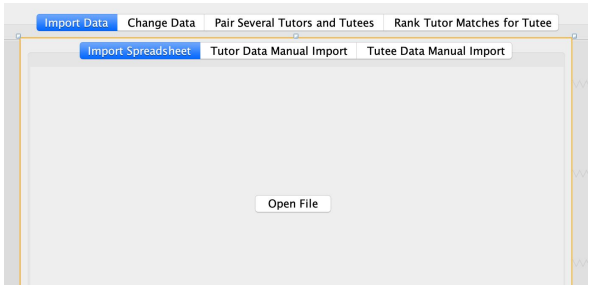


Chronological Development Plan

1. Create all necessary objects and add all their attributes - 1 day
2. Write code for manual input and change data tabs - 3-6 days
    a. Writing method in...
        i. Manual input tab which creates instance of tutors/tutees
        ii. Change data tab which sets values of GUI components recognizing search input
        iii. Change data tab which sets attributes of tutors/tutees
    b. While...
        i. Testing if each GUI component inputs data correctly
        ii. Testing if search class works
        iii. Testing if each tab works correctly
3. Write code for csv file import tab (through 3 versions)- 2-4 days
    a. Version 1: Basic file reading program that prints certain attributes of tutors/tutees read
    b. Version 2: Program that prints out all attributes of tutors/tutees read
    c. Version 3: Program which creates instances of tutors/tutees from file
4. Write code for Ranking tab (through 5 versions) - 7-10 days
    a. Version 1: Program that filters available tutors with the same subject
    b. Version 2: Version 1 which also recognizes similar subjects
    c. Version 3: Program that filters tutors with same available times
    d. Version 4: Version 1 which also recognizes times that overlap even if not exactly the same
    e. Version 5: Program that ranks tutors based on time available for tutoring and subject relevancy
5. Write code for Mass pairing tab - 8 days +
    a. Version 1: Program that pairs tutees with their best choice (allows duplicates of tutors to achieve this)
    b. Development of prioritization algorithm which determines which tutee gets their top tutor choice first
    c. Version 2: Version 1 that does not allow duplicates using a prioritization algorithm.
6. Testing the interactions of all tabs and resolving of inter-tab errors - 3-4 days

Word Count: 0 (Bulleted List)

Testing Plan



Import Sheet Data Tab

Things done in this tab:

- Csv file import and creation of tutee and tutor instances from available data

Testing Plan:

- All attributes of Tutee[0] and Tutor[0] are compared to the attributes of the first tutee and tutor in the csv file

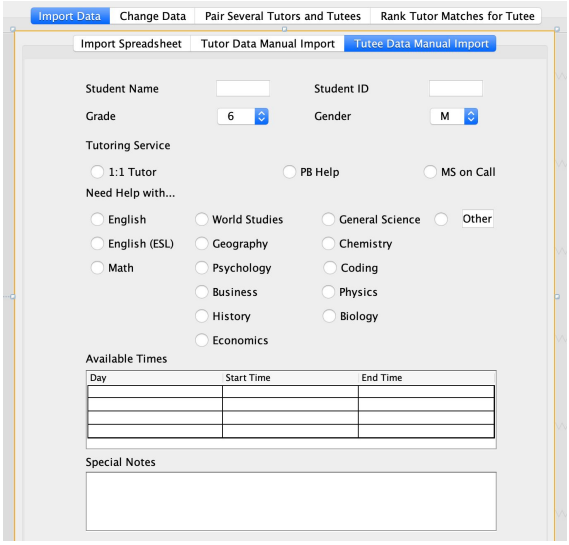
Tutor/Tutee Data Manual Input Tabs

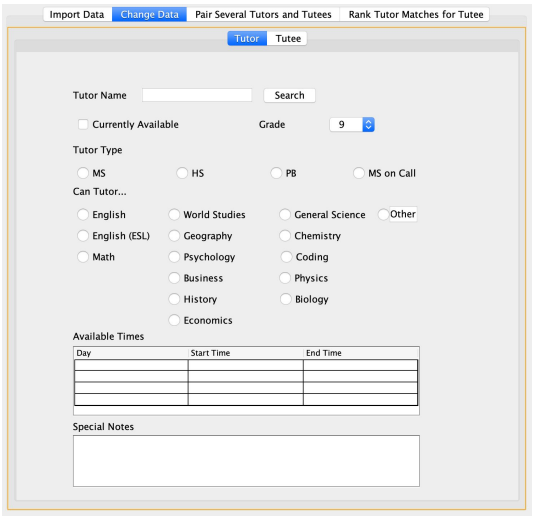
Things done in these tabs:

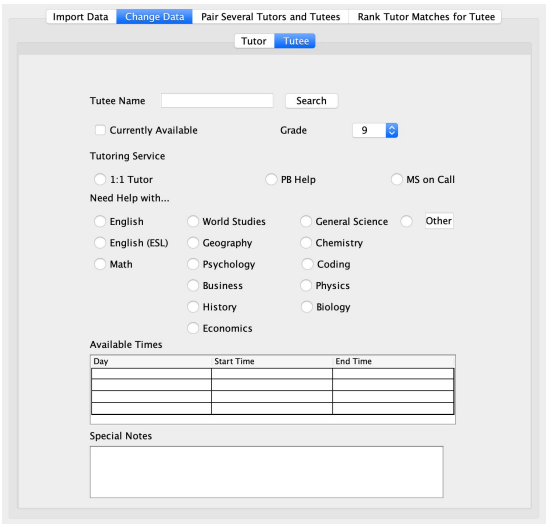
- Inputting of Tutor/Tutee instances

Testing Plan:

- Each GUI component is tested separately by inputting values and printing them as outputs
- 2 tutees/tutors are inputted to the tab with different button combinations/text inputs for all components in the tab and all their attributes are printed







Tutor/Tutee Change Data Tab

Things done in these tabs:

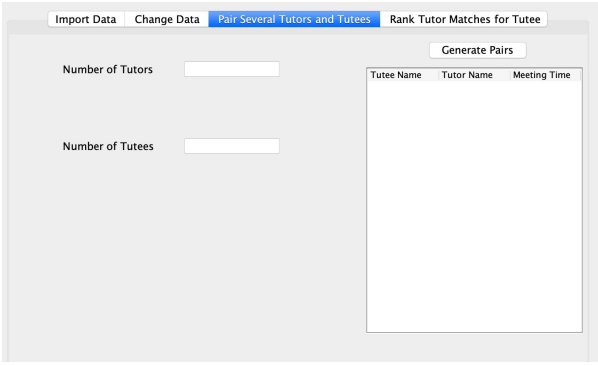
- Inputting of tutor/tutee name
- Display of Tutor/Tutee info
- Saving of edited attributes of Tutor/Tutee instances

Internal functioning

- Search for student input
- Update of tutor/tutee attribute following submission of change

Testing Plan:

- Name of Tutee/Tutor already input is searched, and values shown in the tab is compared with previously printed Tutee attribute
- Change in all components for Tutor/Tutee are made one at a time and the name of Tutor/Tutee is searched each time to see changes



Pair Tutor Tutee Tab

Things done in this tab:

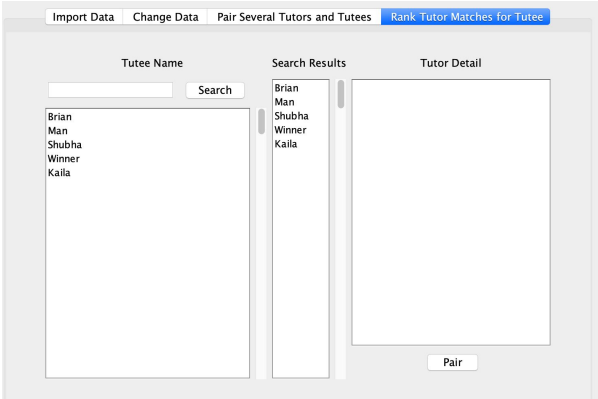
- Inputting of number of tutors and tutees
- Generation of Tutor and Tutee pairs and their meeting times

Internal functioning:

- Run match Function
- Generation of match instances

Testing Plan:

- Import the csv file for a large data set, to...
  - test if numbers of pairs generated match number of tutees
  - test if all pairs have meeting times when the tutor and tutee is available.



Rank Tutor Tutee Tab

Things done in this tab:

- Input tutee name
- Rank tutor names
- Display of selected tutor details
- Resetting of text pane when pair button is clicked

Internal functioning:

- Running of rank function and presentation of tutors in the order which they best meet the needs of the tutee
- Marking of tutor and tutee as unavailable after pairing

Testing Plan:

- Tutee attributes and attributes of tutor presented are compared to judge whether program successfully ranked tutees and tutors

Word Count: 162 (Bulleted list excluded, complete sentences in list included)
Criterion B Word Count: 302