

Criteria A: Planning

Scenario

Ms. Richardson is a teacher in a school with a point system called “House Leagues” in highschool. Currently, there are 4 houses, and each student and teacher is assigned to a house based on their advisories. Houses can earn points by winning event activities or when a teacher gives a house points based on a student’s commendable actions.

During the meeting with the client, she said she was worried that some teachers might be assigning too many points, so she wants to monitor which teachers add points (Appendix 1). She thinks if one teacher is assigning all the points, it’s unfair. After consulting with her, I suggested that I make a program that accesses data from a database and is able to give reports of how many points teachers assign. The program will be shown through a graphical user interface to make using the program easier and more straightforward for the client.

Rationale

In order to solve her problem, I decided to use a database for storing information and a graphical user interface. Using a database can update or modify information and organize the data in different tables neatly and clearly, so it is easy to only show the information about house points through SQL statements within my code. Using a database will be suitable than using files because databases are more structured than files. In addition, since there will be a relationship between houses and teachers, using a database to relate the two would be much easier.

Also, I will be making a graphical user interface instead of a console-based program because although a console-based program would be easier to code, a graphical user interface would be much easier for my client to use. I will use Python because it contains third-party modules that can be implemented in the code such as having the graphical user interface using Tkinter.

Benefits of using database and python:

Database

- Data integrity
- Can perform query statements
- Reduces data redundancy
- Independent from the applications programs

Python

- High-level language
- Can access and interact with a database
- Can create a graphical user interface using tkinter
- Can construct the graphical interface and the structure of the program in one file

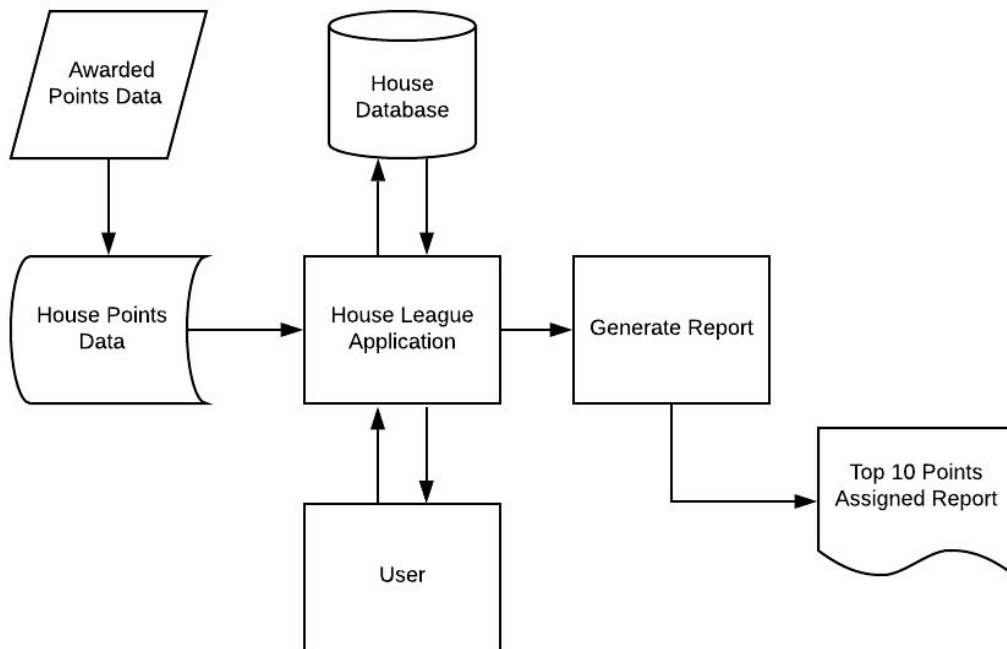
Success Criteria

1. The client gets a report of top 10 teachers assigning the most points by clicking a button
2. The program presents the reports in a table format
3. The client can add a new house and set its house color
4. The client can click a button to reset house points each year
5. The program shows a message to check if the client really wants to reset points
6. The program shows a message when there are data entry errors
7. The client can exit/cancel the program when the client clicks an exit button

Word Count: 362

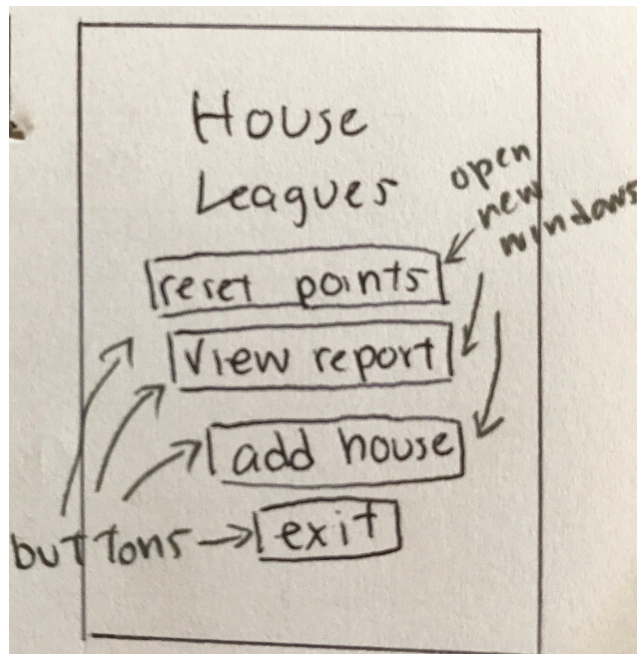
Criteria B: Designing

System Flow Chart

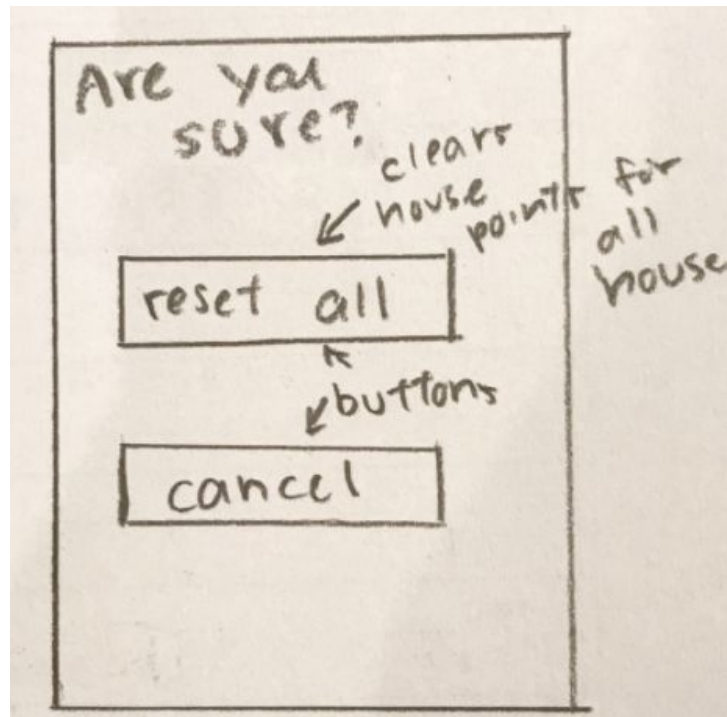


User Interface Design

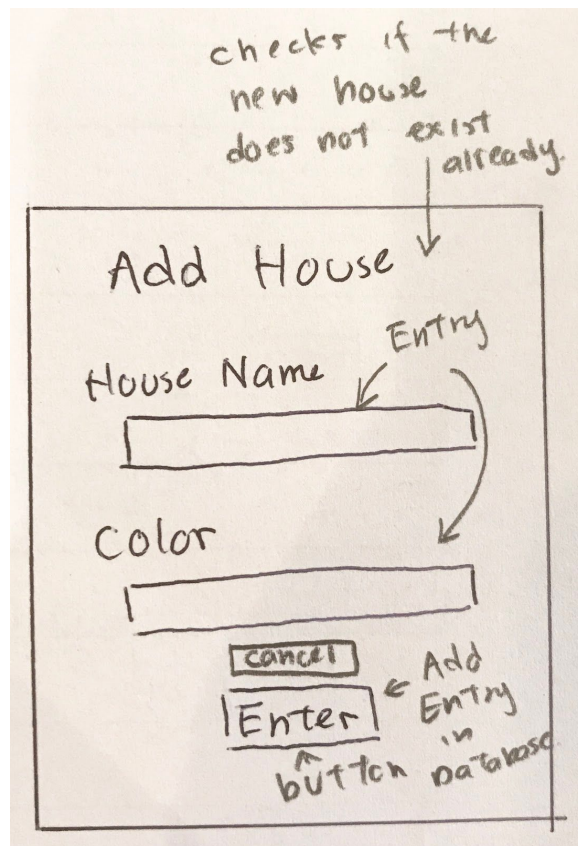
Home Page:



Resetting Points Windows



Add house Window



Report of top 10 teachers:

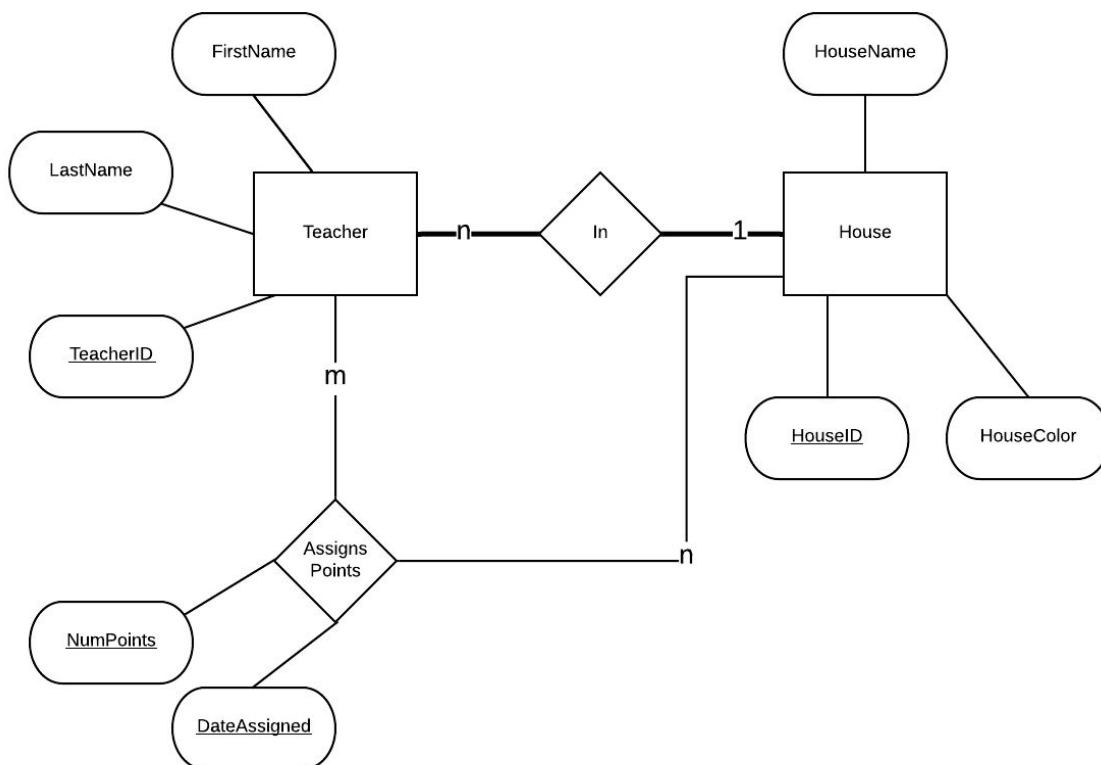
Top 10 Teachers Assigning House Points.

#	First Name	Last Name	Teacher House	receiver house	Points	Date
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						

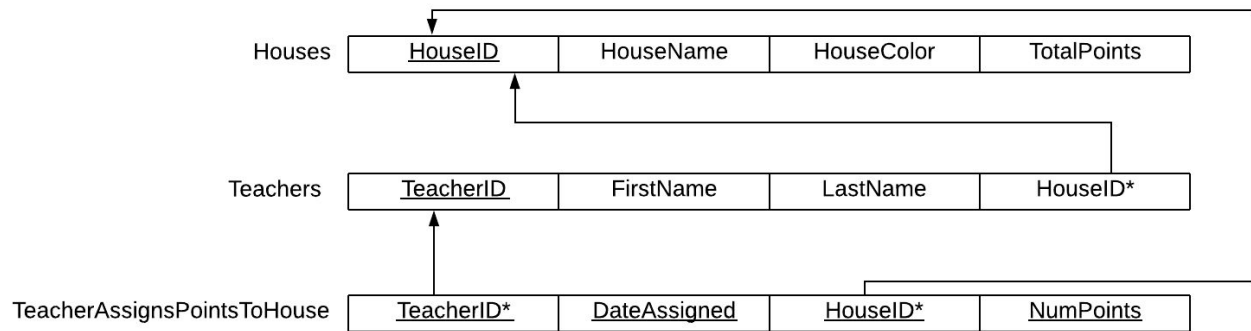
scroll bar
if window gets smaller

Data Design

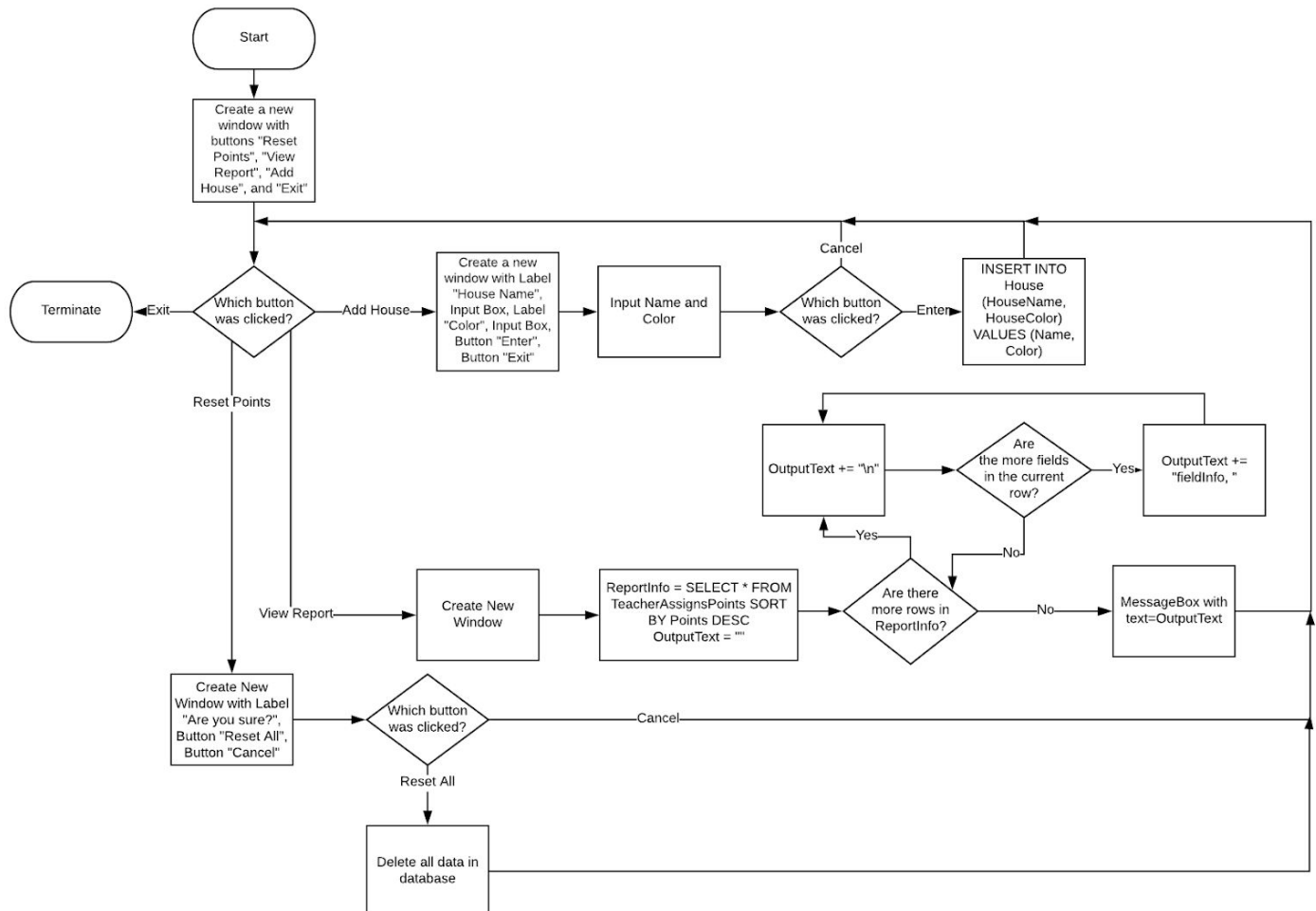
ERD



Relational Mapping



Algorithm Design



Test Plan

Actions to be Tested	Test Method
The client gets a report of top 10 teachers assigning the most points by clicking a button	Run the program and check if the report is visible for the client in a new window by clicking the button on the home page.
The program presents the reports in a table format	Click on the "get report" button on the home page and check if the report is presented as a table and is easy to understand.
The client can add a new house and set its house color	Enter a new house name and house color in the "add house" window and check if the new values are added in the database by running the program to return all the house in the "House" tables in the database.
The client can click a button to reset house points each year	Open the "reset points" button on the home page and click the button to reset all points for all houses. Then check if the total points are set to 0 for each house in the "TeacherAssignsPointsToHouse" table via the "Get Report" button.
The program shows a message to check if the client really wants to reset points	Click the reset all button, and check if the message box pops up. Make sure that the "Reset" button and "Cancel" button reset the points or cancel the reset respectively.
The program shows a message when there are data entry errors	Enter numerical values in the "add house" window for the new house name and house color then see if a message shows that there are errors in the entered values.
The client can exit/cancel the program when the client clicks an exit button	Click on the "exit" button on the home page, and "cancel" button in the resetting points and adding house windows to see if those windows close. The "cancel" button should close the window and return to the home page. The "exit" button should close the program without any messages and interference.

Criteria B: Record of Task

Task Number	Planned Action	Planned Outcome	Time Estimate	Target Completion Date	Criterion
1	Brainstorm for Ideas	Talking to teachers around the school for project ideas	2 days	April 7th, 2020	A
2	Meet with client	The client gives information on her scenario and problem	1 day	April 9th, 2020	A
3	Think of possible solutions	Look at the information given and write possible applications	1 day	April 9th, 2020	A
4	Meet with the client again about possible solutions	Decide on the solution that would best fit the client, and would be doable	1 day	April 9th, 2020	A
5	Start documenting Criterion A	Finish the scenario, rationale and success criteria	2 days	April 14th, 2020	A
6	Check success criteria with the client	Have defined success criteria to base the project	30 minutes	April 16th, 2020	A
7	Draw system flow chart	Finish the system flow chart to start drawing ERDs, Relation Maps and Algorithm Designs	1 hour	April 17th, 2020	B
8	Draw ERDs and Relational Diagrams for the Database	Finish all database related designs	2 hours		B
9	Draw flow chart for the algorithm design	Finish all designs for Criterion B	2 hours		B

10	Start documenting Criterion B	Finish documentation of Criterion B	2 days		B
11	Begin writing the code for the GUI	Have a functioning GUI written to start working on the d	2 days		C
12	Begin coding the algorithms for the program	Get a rough draft of the program finished	2 week		C
13	Show the client the rough draft of the program	Get feedback on fixes to the program	2 days		C
14	Modify program based on the client's feedback		2 week		C
15	Start debugging the program for any minor mistakes	Get a final draft of the program	4 days		C
16	Give the user the program to use	The client gives any final feedback on the program	1 week		E
17	Take the feedback into account for a possible updated version of the program		1 day		E
18	Make a video demonstration of the program		1 week		D
19	Finalize documentations		1 week		D