HOMEWORK WEEK 2
Project Plan

Group Name: Globetrotters
Group Members: Shayli Patel, Cat Phillips, Michaela D'Mello, Nosheen Masud, Lana Moroney

Project Type: Text Adventure Game in Python
Game Name: Where is Ada Lovelace? (*name yet to be confirmed*)

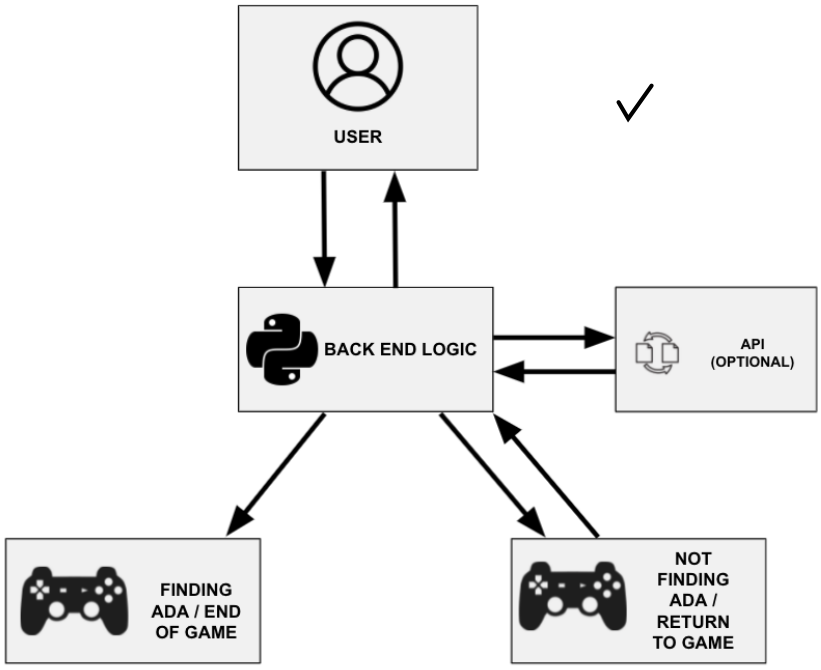
About the game:
We will be building a walk-through text adventure game that requires the user to travel to multiple cities where they are given clues to figure out in which city Ada Lovelace is hiding. In each city the user will be given a series of clues, which they have to use to progress onto the next city. There will be 3-6 cities in total. If the user guesses the wrong city, they will not be able to progress to the next city. Only when they go through the series of cities will they be able to solve where Ada Lovelace is hiding.

For game flowchart, see appendix i.

api part sounds really cool, excited to see how it turns out!

Key features of game:
- Dictionaries for clues
- Itertools, Collections for clues
- APIs (possibly) for tasks (spotify for clues, currency convertor for money option)
- Classes and Decorators within code
- Testing file

Architecture diagram of game:



Building Phases:

Phase 1: Build one location, set up the tasks and process flow to the next location or back to start depending on success. Run Testing.
Phase 2: Extend to 3 locations. Run Testing.
Phase 3: (time permitting): Extend to 6 locations, add in currency convertor task, add in additional character of villain. Run Testing.

Team approach:

- Weekly scrum style meetings to monitor progress and build, phase on phase
- Slack and Trello for ideas and to stay on track for all elements of project
- GitHub for version control to store and share code
- Shared Google Drive for all non-coding documents
- Work on locations either individually or in groups of 2
- Testing to be done at the end of each phase
- Implementing DRY (Don't Repeat Yourself) method in program

how would it be coded? is it going to be x-number of programmers working on y-number of features e.g. 2 people per feature?

what may code-logic look like? is it entirely python code for the back-end / to operate it e.g. if conditions to determine answer is correct or not, etc

comments to consider, not actual marking ^

Appendix i: Flowchart for game: