Erica Li - 101179636
Preethi Nagabandi -101187515
Judy Wu - 101184803

COMP 1406 - Final Project Proposal

Project Name: Good Skill!

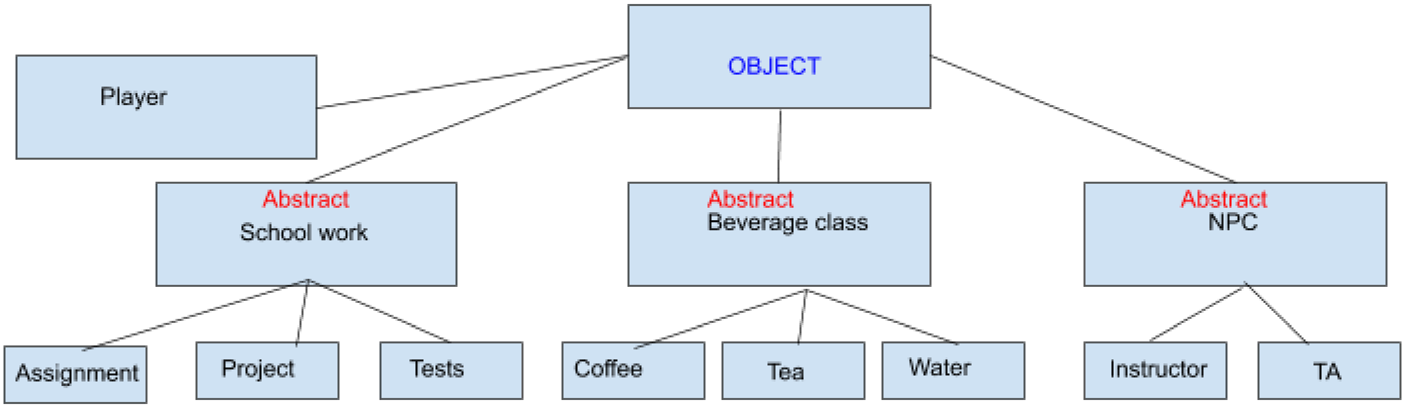
November 2020

For our project we are making a student themed platformer game using javaFx. There would be a student sprite running and jumping through platforms, and controlled by either WASD or arrow keys. There would be beverages such as coffee and tea that the character can drink to increase their speed and jump height. The enemy sprites will be assignments, midterms, and projects. The player can either defeat them by shooting “words” from their pencil or jumping over them. Jumping over them would affect the player’s score/gpa. The landmark at the end of the game would be winter break. The player’s score will be displayed at the end of the game as their GPA.

Necessary data:

- Getting user input in order to provide output
  - This would be a listener (event handling method) for a user clicking a certain key on their keyboard for the player to perform actions
  - For example, jumping when the spacebar is hit

Class hierarchy model:



Algorithmic solutions:

- Basic physics such as jumping or acceleration
- Checking to see what is touching
  - Can be useful for determining if player is hit by a projectile or enemy
    - Which would decrease their player’s HP

Course concepts:

- OOP Principles

Erica Li - 101179636
Preethi Nagabandi -101187515
Judy Wu - 101184803

- Abstraction
    - Implementation of abstract classes: School work, beverage, and NPC classes
- Encapsulation
    - private/protected variables and methods
    - setter/getter methods
    - Separate view from model
- Inheritance/interfaces
    - Abstract School Work class:
            - Paper with words
            - Jumped over assignment/skip will reduce gpa
            - Projects
            - Tests
            - Different assignment types
                - Different assignments can have different difficulty levels
                - Make them behave differently as an extra
    - Abstract Beverage class:
        - Sub classes representing tea/coffee/water, etc
        - Different beverages have different effects of the player's health/performance
        - Makes player faster, jump higher/further, shield for 5 sec
        - Coffee
            - After certain amount of cups within X seconds, it will hurt player's performance
        - Tea
            - Less intense than coffee, less potential for getting hurt
        - Alcohol
            - Depressant, make you slower
    - General player class:
        - Can move up and down, jump etc
        - Can "die" from exhaustion
        - If we have time to have different player type subclasses for the user to choose from
            - Different players have different tools/choose your tools (potential)
    - Abstract NPC class
        - Be able to communicate to player throughout the game
        - Instructor (Dave will be an instance of Instructor)
            - Introduce objective of the game and the rules on how to play
                - "I laid out this work for you that you must complete by the end of the term to pass the course"
            - Communicate random quotes throughout the game

      - TAs
        - 4 instances of TA class to represent the TAs in the course
        - Give tips if player is having trouble completing an assignment
        - Warn player if they are running low on health
        - Tell player how much progress they have made
      - Give a brief tutorial at the beginning
    - Platforms
      - Player cant be inside the platform but on top of them
    - Hud class
      - Shows user their HP
  - Polymorphism
    - Treat all enemies the same in certain scenarios
    - ArrayList of enemies
- JavaFX
  - GUI
    - Implementing User Interface
      - Adding components
        - Buttons, TextFields, ChoiceBox, etc.
          - Health Bar/Exhaustion/HUD
          - Messages from NPCs at the top of the screen (tips, explanation of rules, etc.)
          - Icon of powerups you have (potential)
          - Display number of assignments/tests they still have to complete
          - Display GPA at the end of the game
        - Key pressed
        - Containers
      - Different panes
      - Methods to improve the layout of the window
        - setText, relocate, setStyle, etc.
    - Event Handling methods
      - Get user input and provide the required output
      - Button pressed, keypressed
        - Ask user to enter their choice of player, tools they want to use, powerup they want to use, etc.

Next Steps:

- Make different assignment types have specific ways to be defeated
  - Essentially more variety in their behavior
- Allow user to store a powerup to use later
- Implement different player types for user to choose from at the start of the game
  - Different players can have different tool/advantages in defeating certain assignments

Erica Li - 101179636
Preethi Nagabandi -101187515
Judy Wu - 101184803

- Add different levels to represent different months in the semester
    - As you process the levels will be more difficult, have more assignments to complete
- Different players
    - Maybe having different players you can choose from the start, you can have different playstyle
- Save user progress in the game
    - Save player name, game stats to a file
    - Read saved info from a file to start the game again at a specific point
    - Remove info from file once they completed the game
- Different attacks moves a player can do (may be different for each player type)
    - Need to have a certain amount of energy to perform a more powerful move