

Project 1 Design

Monday, March 19, 2018 4:46 PM

① Analysis : Use cases

- Start menu :

- ① new game : Starts a new game. Tells player starting attributes and brings up game menu.
- ② High scores : Displays up to 10 Highscores, prompts user to either return to main menu or quit.
- ③ Quit : Ends program

- Game menu :

- ① move : move one space, lose one time, chance of encounter/puzzle
- ② read tech papers : Increase intelligence by random amount (1-5) and lose fixed amount (2) of time.
- ③ Search for loose change : Boost money by random amount (\$0.25-\$3.50) lose fixed amount (2) of time.
- ④ view character : show stats
- ⑤ Quit : Lose game, return to main menu

Possible Encounters :

- Encounter Professor : lose time, chance to gain intelligence
- Encounter Grad student : Lose time, chance to gain or lose intelligence
- Find Papers to grade : Lose time, gain money
- Slip on banana : Lose int, move extra step
- Run into Ex : Lose time, move back space
- Find watch : Gain time

Possible Puzzles :

- When was the Shelby center first opened?

Answer : 2007

- Who is bigger: Mr. Bigger, Mrs. Bigger, or their Baby?

Answer: The baby because he is a little bigger

- A farmer has 17 sheep and all but nine die. How many are left?

Answer: 9

- How many months have 28 days?

Answer: 12

- What are the next 3 letters of following: J, F, M, A, M, J, J, A

Answer: S, O, N (first letter of months)

- Jimmy's mom had 4 children. She named the first Monday, the 2nd Tuesday, 3rd Wednesday. What is the name of the 4th?

Answer: Jimmy

② Design

Menu: Holds game menu and handles most errors from user input.

- Variables:

- String PlayerChoice: holds player's menu choice

- Functions:

- Public String getInput()

- Displays 5 menu options and asks for choice, stores choice in playerChoice

- Public getChoice(): returns choice

- Class relation: This class doesn't rely on other classes.

- Other notes: This class is called by System

System: Controls most of the game and calls to other classes

- Variables:

- int fixedTimeLoss: lost time for collecting change or reading

- int maxStartInt, minStartInt: max/min starting intelligence

- double maxStartMoney, minStartMoney: max/min start money
- int maxStartTime, minStartTime: max/min start time
- int hallwayLength: Length of hallway
- Functions:
 - Public void newGame()
 - Instantiate char variables (call character class)
 - call menu.getInput
 - Process input and call other functions based on it.
 - Public void setFixedTimeLoss(int timeLoss)
 - sets fixedTimeLoss
 - Public void setStartIntRange(int min, int max)
 - sets min/max start Intelligence range
 - Public void setStartMoneyRange(double min, double max)
 - Sets min/max start money range
 - Public void setStartTimeRange(int min, int max)
 - set min/max start time
 - Displays warning if less than hallway length.
 - Public void setHallwayLength(int length)
 - set Hallway length
 - Private void readPapers()
 - read papers (lose fixedTime, gain rand int amount)
 - Private void changeSearch()
 - search for change (lose fixedTime, gain rand \$\$)
 - Public void displayHighScore()
 - display highscore
 - Public void saveHighScore()
 - Saves highscore

- Public void endgame()
 - End game scenarios
- Class relation: This class calls basically every other class.
- Other notes: This is the backbone of the game. By allowing users to change ranges, it makes it more portable.

Encounter: determines encounters

Variables: There will be variables for % chance of each encounter, but not yet decided all encounters and chances.

Functions:

- Public void randomEncounter()
 - create rand # then call corresponding encounter
- Possibly private functions for each encounter.
 - Have to see if it will enhance the code's readability or hurt it.

Class relations: calls to change character stats, if puzzle, calls puzzle class.

Puzzle: Determines Puzzles

Variables: Probably no major variables

Functions:

- public void randomPuzzle()
 - Create rand # then call puzzle.

Class relations: Is called by Encounter, calls character class to change character attributes

Character: Holds character attributes

Variables:

int time: time player has

int intelligence: player intelligence

int money: player money

Functions:

- public void displayStats()
 - display stats
- public void setTime(int newTime)
 - sets time
- public void setInt(int newInt)
 - sets intelligence
- public void setMoney(double newMoney)
 - sets money
- public int getTime()
 - gets time
- public int getInt()
 - gets intelligence
- public double getMoney()
 - gets money
- public void loseInt(int amount)
 - lose amount of intelligence
- public void gainInt(int amount)
 - gain amount of intelligence
- public void loseTime(int amount)
 - lose set amount of time
- public void gainTime(int amount)
 - gain amount of time
- public void loseMoney(double amount)
 - lose amount of money
- public void gainMoney(double amount)

- gain amount of money

Class relations: Any stat change requires this class to be called.

Other notes: After writing this out, it seems excessive.

I might just set the attributes to public, so classes can directly change them instead of using so functions.

③ Testing scenarios

Normal usage:

- Typing a number (1, 2, 3, 4) should call the corresponding menu option's function.
- During a puzzle, typing the correct answer should give the "positive" outcome. Wrong answer should give "negative" outcome
- Any movement down the hallway should move one step closer and use one time unless the encounter changes this.
- Winning the game should prompt for a name for high score.
- Quit option for mainmenu should end program
- Quit option for game menu should return to main menu (and taunt user for losing)
- Show scores option should show up to 10 high scores.

abnormal usage:

- input should always be checked.
 - main menu and game menu should check for integer input for options. If wrong, prompt user for

valid input

- Puzzles also must check.
- If max time is set less than halfway length, the user must be warned that the game will most likely be impossible to beat.

Other ideas (only implement if time allows)

- Save/Load game
- Difficulty settings
- Hints for puzzles