Game Project I

GAME2050 Game Programming I

Evaluation : 20% of final grade
Due Date : Mar 10

Assignment Description

The scenario: You are a game developer for a small game company. The company has received a small contract to produce a race betting style game to be deployed on a client's website. Another developer had put together a partial mock up of the game graphics for testing purposes, but at the last minute was re-assigned to another project before coding any JavaScript. It is your job to bring this game to life.

The requirements are:

- The provided assets.fla flash movie contains all the finished animations and is ready for conversion into a single sprite sheet using ZOE. Your job is to bring these sprites to life in a game.
- Gameplay is as follows:
    - Bet Screen: The user places their bet (quarter, half, or full amount of their purse).
    - Choose Screen: The user chooses which spinning wheel to bet on to win.
    - Race Screen: The traffic light comes up and animates, eventually turning to green. When it turns green the spinning wheels start moving towards the right of the stage. On every frame the distance each wheel moves to the right is randomly calculated within some range. This creates the effect of some wheels pulling ahead while others fall behind in the race. Which ever wheel makes it over the yellow line first wins and the game immediately proceeds to the summary screen.
    - Summary Screen: Performs calculations and adds winnings to the purse if any. For example, if the user has a purse of 100 and bets a quarter, the wager becomes 25 and their remaining purse is 75. If the wheel the user bets on wins the race, the winnings is 50 (double of whatever was bet) and added to the purse giving a total purse of 125. Likewise, if the wheel the user bets on loses the race, the winnings is 0 and the total purse will remain at 75. A "Continue" button allows the user to proceed to the Bet Screen again to place another bet providing the user has more than 0 in the purse.
    - Game Over Screen: This screen is displayed if the user has 0 left in the purse.
- See the AVI demo on the functionality of the finished game. Be sure to study this carefully as your game's functionality will need to match it perfectly.
- A user's wager amount must always be whole numbers (Hint: Math Object of JS)
- All output should be done using CreateJS Text objects
- It is required that you develop custom classes for each of the four screens (excluding Game Over) to control the functionality of each screen. The lesson on game screen management is a good reference on how to do this effectively. Follow this structure!
- You are free to use any of the useful custom classes we have developed for this project (AssetManager, etc.) to speed up your development
- Be sure to include internal comments throughout your code

> See Other Side

Requirements (Marks breakdown)

Game Project	
Game.js (Main JS file) *Screen Object Instantiation, Screen management, etc.*	4
Bet Screen / Choose Screen Classes *User wager in whole numbers, Updated text, Saving data, etc.*	7
Race Screen Class *Random Wheel Movement Motion, Traffic light implemented, Winner Detection, etc.*	6
Summary Screen Class *Calculations, Update text, play again or game over, etc.*	5
Comments included	2
TOTAL MARK	24

Other Notes

As you know, it is required that each screen have a custom class to control it. But player information (bet amount, purse amount, etc.) needs to be monitored throughout the game, so a Player custom class might be useful as well.

The graphics for this game are boring at best! Feel free to add your own design to them, but remember that they will not be marked – only the game functionality is being marked for this project.

Remember to divide and conquer this project! It WILL seem overwhelming if you try to tackle all the functionality of the project at once. An example on how to complete might be as follows:

- Add the default starting template for an HTML5 game (game.htm / Game.js)
- Add the screen management code to get the four screens appearing at the correct times
- Add the functionality to the race screen to get your spinning wheels to move to the right of the stage
- Add functionality to get the traffic light to start the race
- Add the functionality to get the wheels to randomly adjust their movement each frame
- Add functionality to detect the winner
- Etc…

This project will be marked in person on the due date through code review. Be prepared to explain your game code when questions are asked. Your mark will be affected if you do not understand your own code.