

PROJECT PROPOSAL: Jump Car Game

Jennie Wei

andrewID: jenniew

Project Description

Jump Car is a game where the player is a car and the goal is to jump to as high of a platform as possible without hitting NPC cars on other platforms. The player car automatically drives back and forth on a platform the width of the screen and each platform has a bot/npc car driving back and forth on it(with different speeds/features).

Similar projects

Jump Car was an IOS game by Ketchapp with the similar gameplay as described above. Unfortunately, this game has not been updated to be compatible with the new IOS updates and there is no online version currently available, so creating this game for TP would finally give me a means to play it again.

My project is similar to Ketchapp's Jump Car in terms of gameplay and structure. Creating this game itself is quite challenging for me, so I don't plan on drastically changing the main game features, however, I do want to add different kinds of special NPC cars and I would like to create/draw my own graphics with a different theme.

Structural Plan

I plan to organize the elements of my project in OOP classes:

- I will have a 'Car' class that creates all the Cars in the game(both NPC and player
 - Some attributes I anticipate adding are car speed, car direction, car image/icon
 - Each type of car will have a subclass inheriting from the Car Class
 - (Tentative) As the cars drive they all leave a smoke trail and I plan to implement that as a function in the car class that creates smoke circles in the car's previous positions
 - the 'Player' Class with inherit from the Car Class
- I will have a 'Platform' Class to create platforms
 - attributes will be platform color(determined by whether the player car has successfully landed on the platform) and platform number
 - I will store all platforms in a static variable list in the Platform class
- (Tentative) create a 'BackgroundElements' Class that contains all the background elements in a static list and subclasses for the elements in the background(i.e. clouds, stars)

Algorithmic Plan

- Player Car/Platform Interaction
 - The player car needs to pass through the platform on first contact as it jumps up then land on the platform on second contact—create attribute to track when bottom of the player car touches top edge of platform
 - if its first contact, solid=False, second contact solid=True, for all jumps afterwards solid=True(because player car can't fall back down)
- Character movement
 - pressing space key makes the car jump upwards(in addition to continued horizontal movement) onStep, letting go of space key makes the car fall onStep until it lands on a platform
- Moving Camera as the Player jumps
 - create a 'map' of platforms much longer than the screen and move through the map as the player jumps higher

Timeline Plan [2.5 pts]: A timeline for when you intend to complete the major features of the project.

I plan to complete the bulk of my project over thanksgiving break:

- implement generating platforms as the player moves up
- create the Car class and different car subclasses
- implement how the Player Car interacts with platforms

After Thanksgiving break:

- create my own graphics(tentative)
- create a starting screen with instructions
- track high score, track average score (for each session)

Version Control Plan

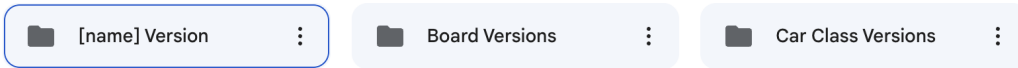
I plan to use Google Drive for Version Control:

- paste versions into google docs and sort them in folders based on step in the process/whats being changed

My Drive > 15112 TP ▾

Type ▾ People ▾ Modified ▾

Folders



(folder names are hypothetical)

- **Module List** [1 pts]: A list of all external modules/hardware/technologies you are planning to use in your project. Note that any such modules must be approved by a tech demo. If you are not planning to use any additional modules, that's okay, just say so!

No additional modules!

TP1 update:

- Add 'AI' cars that are aware of player movements and respond to them
- MVP includes functional gravity(parabolic movement/acceleration changes in jumping
- Add Powerups
- create a separate shortJump function for key presses
 - problem: only having a jump function based on keyhold makes game harder to play as if you don't hold the space key long enough, you don't jump high enough to get over the enemy car
 - short jump makes sure very short key presses still jump a certain height for better UX/easier gameplay

TP2 update:

- side scrolling platforms in addition to vertical scrolling

TP3 update:

- All UI elements hand-drawn(player, platforms, instructions, background, assets, textboxes, etc.)
- Storing high score and high scorer username in a text file + restarts game with same high score stats
- Increasing difficulty as player goes up (less empty platforms, no shield after platform 20)