

TP Design Proposal - UPDATED

Akintayo Salu

asalu

Friday 22nd April, 2022

Project Description:

1. Project Name: Temple-12 Run
2. Brief Project Description: Temple-112 Run is a 3D endless running game that is inspired by the mobile video game series *Temple Run* developed by Imangi Studios. Temple-12 Run will utilize keyboard presses and/or hand tracking to move a user's player through a 3D ray cast world.

Competitive Analysis:

Within the modern gaming industry, there are a number of 3D endless running games; the most popular ones include *Subway Surfers* by Kiloo and SYBO Games and, of course, *Temple Run*. While most of these games use touch features on smartphones and tablets, Temple-112 Run will use keyboard presses and/or hand tracking for moving avatars within the game. Moreover, another distinction between this project to others is the fact that it is being built with Python, which makes 3D world creation much more harder. Popular endless running games like *Temple Run* are built on Unreal Engine C++, which is optimized for graphics and world generation. For Temple-112 Run, raycasting will be used with drawing functions in CMU 112 Graphics packet will be used to create the 3D world.

Structural Plan:

- The following pseudo-code highlights the most important functions of the program.

rewdrawAll(app,canvas):

handTrack(app)

drawMenu(app,canvas)

drawWorld(app,canvas)

runGame(app,canvas)

If (gameIsOver(app)):

drawExitPage(app,canvas)

Algorithmic Plan:

The most complex parts of the project will be translating the hand tracking feature to moving the avatar within the game and creating an endless, random 3D world for the game.

1. Hand tracking/ Movement: As this is one of the most complex features of the game, I intend to use specific conditionals that translate the x-,y- coordinate of the hand being tracked to the movement of the avatar within the game.
2. World generation: I intend to implement this complex feature by creating an 8-by-8 randomly generated 2D list that has an "entrance" and an "exit". When a player reaches an "exit", a new 2D list is created and that exit becomes the "entrance" to the new 2D list that raycasting is carried out on to create a 3D world.

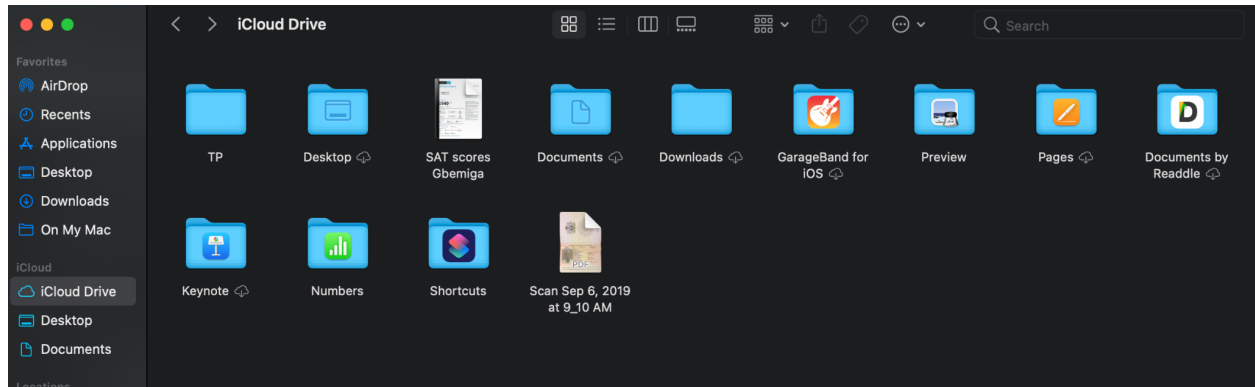
Timeline Plan:

- TP2: Have a working game that has the endless world feature with an avatar that seamlessly runs within it. The world should also autogenerate obstacles.
- TP3: Improve the UI and UX of the game to make the game easier to use.

Version Control Plan :

Code is being saved in the iCloud:

Image evidence:

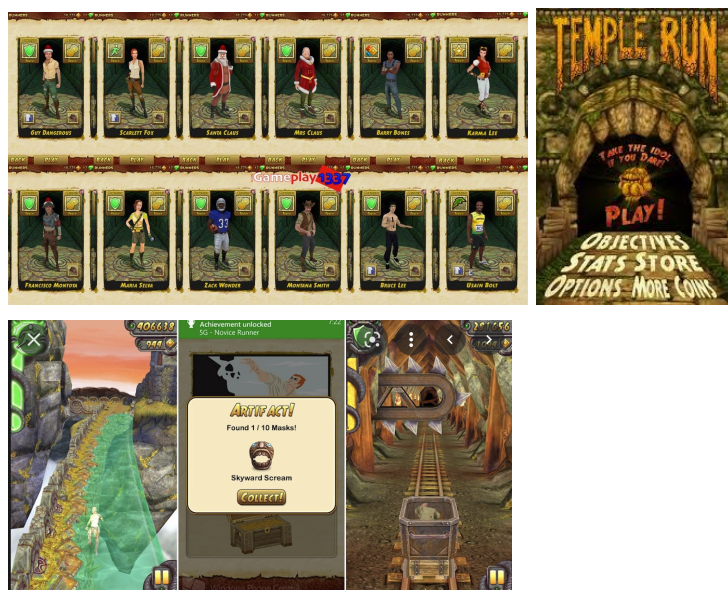


Module List:

- mediapipe for the handtracking
- openCV

Storyboard:

These images are inspirations for the menu pages, opening and closing pages as well as gameplay for the game.



TP2 Update:

- Face Tracking successfully implemented,
- Due to an inability to make obstacles with sprites/regular images, obstacles were made using “ray cast” walls.
- The game looks more like a 3D endless running game in a maze with obstacles, instead of the actual *Temple Run* in real life. Possible new project name: Maze-112 Run
- Raycasting is tasking on the computer, so the number of rays to work has to be a third of the maximum number of possible rays,
- Upon revision, I am doing head tracking instead of hand tracking.

TP3 Update:

- A Home Screen was added,
- An instruction page was added,
- A shortcut to display the menu was added,
- The code was put into classes.
- In general, the UI was somewhat improved.