# Games Programming (AC31009)

# Game Name: Kid Dash

# by Bernard Aihevba – 2400960

Table of Contents

* Overview of the game
* Changes made to the Game and Why
* Design Patterns
* Game Concepts
* Ethical and Security Considerations
* Things I Would do Differently
* Game Testing

**Overview of the Game**

Kid Dash is a 3d endless runner game. It was inspired by games like Sonic Dash, Temple Run and Subway Surfers – popular games played by many children all around the world. The game was created and coded in the Godot engine and with GDScript which is a custom scripting language for the engine. The game features a runner character controlled by the player. The aim of the game is to avoid all obstacles by jumping or moving left and right through the game. The appeal of the game is to make the player, with each run of the game, try to beat the high score of the previous run.

**Changes made to the Game and Why**

There were some changes made to the game plan. I initially intended to and started creating the game using C# language as it is one, I’m familiar with. Upon beginning and making my research I did discover GD Script which is said to be more convenient when it comes to game development in the Godot engine. For this I had to make use of various YouTube tutorials and coding forums to learn and get acquainted with GDScript.

**Design Patterns**

For the game coding, the design patters I used include ;

* Command (Method Call wrapped in object):

In my code I used the rand\_range() function in Trees.gd to position different instances of the tree rotation in various parts of the game environment. Can be found in lines 20-22 in Trees.gd.

* Prototype:

In my code, for the game environment, this is used to spawn various instances of the game. There is the instance with no obstacles, the instances with obstacles (in the middle, left and right lanes respectively). This can be found in the Chunks.gd code of my game

* State:

The runner character of my game has states which are running and jumping states. The running (in the center) state is the default state of the character. The other states are; running on left lane (triggered by the left arrow button or “A” key), running on the right (triggered by the right arrow button or “D” key) and jumping (triggered by the up arrow button or “W” key). The implementation of this can be found in the \_physics\_process() function of Player.gd

* Game Loop:

The game listens for user input and moves the game runner in accordance with player’s input.

**Game Concepts**

* Physics:

Since the main aim of the game is to dodge obstacles, the questions of what happened if there was collision and how to determine if there was one, were one of the most important ones that came up during the thought process of the game play. In the game, the runner character and obstacles were all given collision shapes. The \_on\_Obstacle\_body\_entered() function in the Obstacle.gd code determines if the runner character collision shape collides with obstacle collision shapes. This will then prompt the game over screen.

* Level Creator:

The game is just one infinite level that only ends when collision between the runner character and obstacles is detected.

* Procedurally Generated Content:

As mentioned in the above point, the game consists of one infinite level that keeps going as long as no collision is detected. As the game is still active, more parts of the level are spawned hence the game being generated procedurally as it is active.

**Ethical and Security Considerations**

1. The age range for the game.

The game is made for all ages and therefore pose no moral or explicit threats to players of the younger generation.

1. Discrimination in the game:

There is no occurrence of discrimination in the game and therefore would not be found offensive by any group of people. It is made to be enjoyed by people of different races, religions, sexual orientations and walks of life.

1. Violence:

This is violence free game and therefore cannot introduce violent behavior to players.

1. User Data:

The game does not require any data from the player and as a result, there is no risk of data leaks, and the question of what user data can be kept is not relevant in the analysis of this game.

1. Copyright:

Although the game is inspired by popular games; Sonic Dash, Subway Surfers and Temple Run, it does not include any item from any of these games that may raise a copyright issue.

**Things I Would do Differently**

If I were to tackle this game again, there are a few things I would do differently and add to the game.

1. I would add a better game startup and include a menu where users can select different runner characters to start their run
2. I would add a pause button so players can pause and play the game at their own convenience
3. I would make a version of the game adaptable to mobile devices and other devices that are not laptops or pcs.
4. I would create a file to save the high score of the player and update that if it is ever passed
5. I would add collectable coins in the game which player can accumulate and use to purchase more characters and power ups
6. I would add a social media sector where you can connect with friends and compare high scores on a leaderboard and also post to social media like Instagram and Facebook
7. I would add music to the game to make the gameplay more enjoyable
8. I would add provisions for the disabled e.g Those with sight or hearing impediments and those tremor diseases like Parkson’s disease.

**Game Testing**

For the testing I had different friends test the game on different operating systems and it worked on Windows, MacOS and Linux. I also had people of different ages test the game and the results where positive as all testers where able to understand the gameplay easily and had no issues navigating through it. The Godot engine is needed to run the game presently but if tackled again, would be made to run without the engine being present and just the game file.

References

[Godot beginner tutorial - endless runner or infinite runner game (generalistprogrammer.com)](https://generalistprogrammer.com/godot/godot-beginner-tutorial-endless-runner-or-infinite-runner-game/)

[(44) How to make a 3D Player - Godot Tutorial - YouTube](https://www.youtube.com/watch?v=k3QZrtbK2_w)

[(44) Godot 3 - Subway Surfers Tutorial - Part 1 - YouTube](https://www.youtube.com/watch?v=D8LnTRXnVF8&t=42s)

[Episode 1: Make a 3D endless runner game with Godot C# - YouTube](https://www.youtube.com/watch?v=6jUWEKFk9oI)

I also made use of a number of stackoverlow solutions.