College Degree

*By: Giovanna Borges Bottino*

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Change Description** |
| 1.0 | 01/21/2019 | Giovanna | Game Idea |
| 1.1 | 01/29/2019 | Giovanna | Document Created |
| 1.2 | 02/04/2019 | Giovanna | User Interface and Technical Specifications added |
| 1.3 | 02/05/2019 | Giovanna | Technical Specifications update and Pre Game added |
| 1.4 | 02/12/2019 | Giovanna | Document has been revised |
| 1.5 | 02/14/2019 | Giovanna | General document updated. |
| 1.6 | 02/15/2019 | Giovanna | General document updated |

**SUMMARY**

[Pre Game Design Document](#_4am13v6r1sdz)

[Assets](#_is1uay9freli)

[Game Overview](#_kuud0iqd2nut)

[Influences & Sources](#_z9gydttb27tu)

[Target Market](#_4mee4jawzv2t)

[Pitch](#_pxlyg1bak0e5)

[Look and Feel](#_8qh94dibpo2)

[Game Mechanics](#_5f6wqnyacp7n)

[Core Game Play](#_34hippmgn0ob)

[Speed](#_vwzk49m58j86)

[Game Flow](#_nb45peobur5y)

[Characters/Units](#_h3us1xr6nnsm)

[Game Play Elements](#_ci7oqrqcikqf)

[Mission and store](#_9x8ci8lzgbtj)

[Technical Specifications](#_lap13t6eukqy)

[Game Mechanics](#_9dn1jmi6szgw)

[Game engine](#_v99ikfkvjx01)

[Platform](#_d6c26x3lbasd)

[Code Objects](#_g4ji0mr1lmjt)

[Player Object](#_bq2ituew59uz)

[Track Object](#_h2qy2ji28mon)

[Obstacle Object](#_3alcqow8lbq2)

[Coin Object](#_2h8mhjvdgnmx)

[Power Up Object](#_mx697o7jxeyg)

[Coffee Object](#_egycig6i5qpt)

[Camera Object](#_yo6u7fanc2xv)

[GameManager](#_5xhd12q5qs14)

[Scripts](#_pzxy1mpecjdo)

[C#](#_y6e9986plsus)

[User Interface](#_qdxecp7u71zf)

[Flowchart](#_xv0siz244vsu)

[Functional Requirements](#_7adrvzjx5sdn)

[Mockups](#_p8y4dgi5k4j1)

## 

## Pre Game Design Document

### **Assets**

The current game was developed using the Assets found in the [Unity Assets Store](https://assetstore.unity.com/packages/essentials/tutorial-projects/endless-runner-sample-game-87901#reviews) and in the [TurboSquid](https://www.turbosquid.com/). These Assets were only used as preview about the game and how it will work. To develop the game according to this document it is necessary to create new Assets.

## Game Overview

College Degree is a 3D third-person endless runner game for mobile devices. The objective is to run as far as you can across three lanes in order to achieve the degree, reach a high score or complete missions. The player can choose to control a university student called Jenifer or Rodrigo and run through the scenario.

While running the player must avoid, by touching the screen, oncoming obstacles and trying not to sleep. If the player doesn’t collect coffee icons, he’ll sleep, by sleeping it is game over. If the player collect too much coffee, the game will become faster. If the player hits an obstacle, it is game over.

### **Influences & Sources**

I have researched and played endless runner games in order to get an idea of the right look and feel for this game as well as to develop ideas for the gameplay and interactions. For example *Subway Surf*, *Temple Run 2* and *Zombie Tsunami*. Beside that, I had help from an Youtube channel, [High Level](https://www.youtube.com/channel/UCetfSquMy40hkTGSh0cRYHQ) and [N3K EN](https://www.youtube.com/channel/UCtQPCnbIB7SP_gM1Xtv8bDQ).

### **Target Market**

The ideal user has the following characteristics:

* 15 to 23 years old;
* Must be a student;
* Has an Smart Device;
* Plays mobile games;
* Universal appeal.

### **Pitch**

Mobile game about a endless run of a student on search for his degree.

### **Look and Feel**

The theme will have a cartoony appearance. Intending to be humorous, some bright colours will help to convey the fun. Due to the fact that the game is set in an university, the list of assets and environmental art will be relevant to a school. Assets such as environmental art will be simple, low-poly and also cartoonish.

## Game Mechanics

### **Core Game Play**

The Playable Character moves in a 3D plane, exploring a runner at school and at house that extend from the top to the end of the screen. The player has to keep attention to not let his character sleep, it will happen when the "eye" goes to zero. “Eye” is the expression to symbolise the student situation. It’s a counter that decrease with time, represent tiredness of being a student. The player have to collect coffee icons that will increase the "eye" score.

Obstacles, power-up and grades appear along the way. There’s two type of obstacles, the first kill you with only one hit, the second let you vulnerable to the next obstacle. Obstacles will vary between teacher, chairs and tables, bed, ruler, black/whiteboard.

There’s two type of power up, the first “Books”, multiply your punctuation but you “eye” is decrease. The second, “Friend”, attract better grades.

Grades are considered coins,they are divide as “A”, “B”, “C”, “D” and “F”.

On “College Degree” the player is going to have differents kinds of missions and objectives.

#### **Speed**

The speed is controlled by the “eye” level, the score start with 17. Eye has four different levels.

1. 0 - the character is sleeping.
2. 1 to 10 - the character is sleepy, in the second level the speed decrease and start to be slowly, if the speed arrive to 0, the player dies.
3. 11 to 20 - in this level the speed start at 10 and increase into 30.
4. 21 to 30 - the character is to awake, in this level the speed increase infinity.

### **Game Flow**

Actions that the Player Character can perform are:

1. Move left, and right, and move left, or right ,while jumping or sliding;

2. Jump up;

3. Slide;

4. Stop a jump or a slide to do a slide or a jump respectively.

### **Characters/Units**

The main character, Jenifer or Rodrigo, is youthful and skinny wearing stereotypical student clothes (jeans, sneakers, flannelette shirt, bag). The model will have a low-polygon count

to maximise the effectiveness of the game given that it is for mobile devices. This will give the

game and our playable character a more cartoonish look. Both characters does not have concept art yet.

### **Game Play Elements**

These are elements present in the game level that the PC can interact with, none does have concept art.

**Coffee item:** Cup of coffee that recover +5 from the eye score and add +2 to the speed.

**Book item:** Book modify your score by “Time.deltaTime \* speed \* 8” but your eye score is decreased by 1.1.

**Friend item:** It’s a NPC, called Emma, she have the same student stereotypical and she walk on the front of the PC attracting better grades;

**Grade item:** Grades are considered coins,they are divide as “A”, “B”, “C”, “D” and “F”.

* Collecting “A” you win 10 coins,
* Collecting “B” you win 8 coins,
* Collecting “C” you win 6 coins,
* Collecting “D” you lose 12 coins and
* Collecting “F” you lose 20 coins.

**Teacher obstacle:** It’s a NPC, that stays still, and appears randomly as the player runs down the lanes. If the player hit it, the character will be vulnerable to the next obstacle. To avoid the teacher is necessary to change lanes.

**Chairs and tables obstacle:** It’s an obstacle that appear randomly as the player runs down the lanes. If the player hit it, the character will be vulnerable to the next obstacle. To avoid is necessary to change lanes or jump.

**Bed obstacle:** It’s an obstacle that appear randomly as the player runs down the lanes. If the player hit it, the character will be vulnerable to the next obstacle. To avoid it is necessary to change lanes or jump.

**Ruler obstacle:** It’s an obstacle that occupy the three lanes and appear randomly as the player runs down the lanes. If the player hit it, the character will die. To avoid it is necessary to jump.

**Black/whiteboard obstacle:** It’s an obstacle that occupy the three lanes and appear randomly as the player runs down the lanes. If the player hit it, the character will die. To avoid it is necessary to slide.

Additionally we have elements that communicate the PC’s game state:

**Score Count:** Text that represents the PC’s score, which increases differently depending on the speed and how far of the character.

**Coin Count**: Text that represents the PC’s number of coins, which increases/decreases differently depending of the collected grades.

**Coin Value:** Text that represents the value of the last coin collected.

**Eye Score:** Image that according to eye level changes animation. Represent at what level of the eye it is.

**Book icon:** Text that appears only when the book item is collected. Represent that the power up is activated.

### **Mission and store**

The game will be developed around the subjects of the college. For a matter to be completed it is necessary to complete an N+2 number of missions. Each mission will fill the academic history with completed lessons. There will be 18 subjects to be made in all. After completing the academic history, the character graduates in college and gives the player the opportunity to continue playing to achieve masters and PhD.

Missions and store items will be connected. With the missions fulfilled, the subject will be completed. The more lessons more items and upgrade for power ups are unlocked in the store.

## Technical Specifications

### **Game Mechanics**

#### **Game engine**

I have used a game engine to develop the game called Unity3D developed by Unity Technologies. A game engine is a system designed to develop games for various platforms like consoles, computers and handheld devices like smartphones.

#### **Platform**

Unity gives the possibility to export the game to various platforms, these includes: iOS, Mac Standalone, Windows Standalone, Web, Nintendo Wii, Xbox 360 PS3 and Android. This game specifically is developed for Android.

#### **Code Objects**

##### **Player Object**

The player object can be identified by the "Player" tag, the player will be the character manipulated by the player. The object has the Player.cs script, which is used to act according to the user input. If this object collides with an object with the tag:

* Obstacle1: The character will lose 2 point of 2 from the life and will die.
* Obstacle2: The character will lose 1 point of 2 from the life and will start to blink. The obstacle will be deactivated.
* Coin[]: The number of the coins will be increased or decreased. The obstacle will be deactivated.
* Coffee: The speed and the eye score will be increased. The obstacle will be deactivated.
* Book: The score will be increased and the eye score will be decreased. The obstacle will be deactivated.

##### **Track Object**

Track object can be identified by the "Track" tag. This object has the Tracker.cs script attached to it. Through this script the object can install different objects, such as: obstacles, coins, power ups and coffees. If this object collides with the Player object it will be reallocated towards the Z axis.

##### **Obstacle Object**

Obstacle object can be identified by the tag "Obstacle1" or "Obstacle2". Objects labeled "Obstacle2" have the ChangeLane.cs script attached to it. Through this script the object can be allocated in 3 different positions in the X axis.

##### **Coin Object**

Coin object can be identified by the tag "CoinA" or "CoinB" or "CoinC" or "CoinD" or "CoinE". These objects have the ChangeLane.cs script attached to it. Through this script the object can be allocated in 3 different positions in the X axis.

##### **Power Up Object**

Power Up object can be identified by the tag "Book". This objects have the ChangeLane.cs script attached to it. Through this script the object can be allocated in 3 different positions in the X axis.

##### **Coffee Object**

Coffee object can be identified by the tag "Coffee". This objects have the ChangeLane.cs script attached to it. Through this script the object can be allocated in 3 different positions in the X axis.

##### **Camera Object**

The camera Object is an orthographic camera that exists in every scene of the game. This object has have the MoveCamera.cs scripts attached to it. This script allow the camera to move in the Z axis.

##### **GameManager**

This object is named GameManage and has the GameManager.cs script attached to it.

### **Scripts**

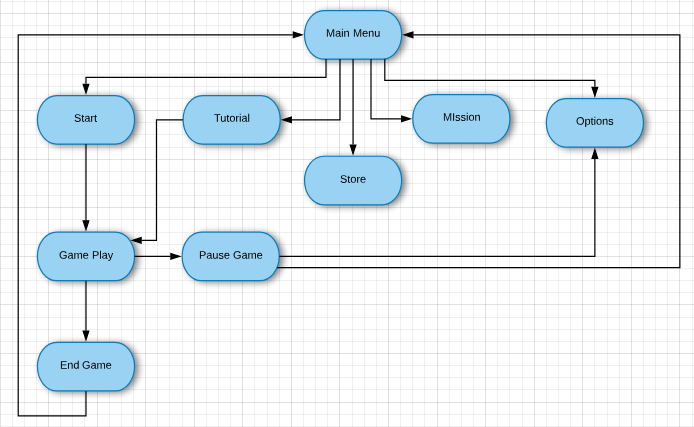
#### **C#**

The game was all developed in the programming language C#. It’s a modern object-oriented programming language. C# is very popular for creating games applications and was chosen because it is powerful, flexible, and well-supported.

## 

## User Interface

### **Flowchart**



#### **Functional Requirements**

**Main Menu:** Presents all of the possible choices for the user as well as a captivating graphic image that introduces the look and feel of the game and of the series.

**Start:** Button to select to start the game.

**Gameplay**: Where the game happens, have a text to count the number of coins and another to count the score. There is a button to pause the game.

**End Game:** Painel that indicates the Game Over.

**Pause Game:** Can be accessed during gameplay from this page, the user can see which mission he is currently in or he can quit the game or go to Options.

**Options**: Gives the user the ability to control some of the game’s attributes, such as sound volume, music volume and the credits.

**Tutorial:** Just like the GamePlay, but with a easy difficult.

**Store:** Where the Player can buy new skins and change character.

**Mission:** Presentes 3 possibles missions.

#### **Mockups**

1. **Main Menu**

****

1. **Options**

****

1. **Mission**

****

1. **Store**

****

1. **GamePlay**

****

1. **Tutorial GamePlay**

****

1. **Pause Game**

****

1. **EndGame**

