**ASSESSMENT AND INTERNAL VERIFICATION FRONT SHEET (Individual**

**Criteria)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course  Title | **Advanced Diploma** | | |  | **Lecturer Name & Surname** | **NEIL AQUILINA** | | |
| Unit Number & Title | | | **Programming for Computer Games** |  |  |  | | |
| Assignment Number, Title / Type | | | **Simple 2D Car Game - Home** |  |  |  | | |
| Date Set | | | 09/12/2020 | **Deadline Date** | **19/01/2021** |  | | |
| Student  Name | |  | | **ID Number** |  |  | **Class / Group** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | *Student’s declaration prior to handing-in of assignment:*  ❖ *I certify that the work submitted for this assignment is my own and that I have read and understood the respective Plagiarism Policy* | | | |
|  | ***Student’s declaration on assessment special arrangements (Tick only if applicable)***   * *I certify that adequate support was given to me during the assignment through the Institute and/or the Inclusive Education Unit.* * *I declare that I refused the special support offered by the Institute.* | | | |
| Student Signature: | |  | **Date :** |  |

|  |  |  |
| --- | --- | --- |
| Assessment Criteria | Maximum Mark | Mark Achieved |
| *KU2: Describe asset types* | 5 |  |
| *KU5: Identify suitable resolution for images of various types* | 5 |  |
| *KU6: Select asset types and settings for a range of media assets for a game design* | 5 |  |
| *KU7: Idenitfy resolution issues for the development platforms* | 5 |  |
| *KU8: Identify appropriate input devices and methods for development platforms* | 5 |  |
| *KU9: Show and explain how to deploy a game to multiple devices* | 5 |  |
| *AA1: Examine and apply basic programming techniques for a simple game* | 7 |  |
| *AA2: Develop a game with graphical and audio assets* | 7 |  |
| *AA3: Demonstrate ability to resolve issues and input devices in the development environment* | 7 |  |
| *AA4: Apply coroutines for a more interesting gameplay* | 7 |  |
| *AA5: Examine and solve gameplay problems* | 7 |  |
| *SE2: Resolve programming issues related to sound, graphics and gameplay* | 10 |  |
| Total Mark | 75 |  |

|  |
| --- |
| **Assessor’s feedback to student** |
|  |
| ***(If necessary, use reverse side of page for IV feedback on assignment brief / sample of assessment decisions)*** |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Name & Surname** | **Signature** | **Date** |
| **Internal Verifier :** Approval of a*ssignment brief* |  | For approval signature, please refer to electronic audit trail |  |
| **Lecturer / Assessor :** Issue of results and feedback to student |  | For approval signature, please refer to electronic audit trail |  |
| **Internal Verifier :** Approval of *assessment decisions (Sample)* |  | For approval signature, please refer to electronic audit trail |  |
| **Learner’s signature upon collection of corrected assignment.** | |  |  |

|  |
| --- |
| Assessment Criteria |
| *KU2: Describe asset types* |
| *KU5: Identify suitable resolution for images of various types* |
| *KU6: Select asset types and settings for a range of media assets for a game design* |
| *KU7: Idenitfy resolution issues for the development platforms* |
| *KU8: Identify appropriate input devices and methods for development platforms* |
| *KU9: Show and explain how to deploy a game to multiple devices* |
| *AA1: Examine and apply basic programming techniques for a simple game* |
| *AA2: Develop a game with graphical and audio assets* |
| *AA3: Demonstrate ability to resolve issues and input devices in the development environment* |
| *AA4: Apply coroutines for a more interesting gameplay* |
| *AA5: Examine and solve gameplay problems* |
| *SE2: Resolve programming issues related to sound, graphics and gameplay* |

IICT4016 -

Programming for Computer Games

**Create a Simple 2D Car Game**

**Task 3: Review, Build and Deploy the Game**

|  |  |
| --- | --- |
| Test Case 1 |  |
| Line Error | *FindObjectOfType<GameSession>().ResetGame();* |
| Error  Explanation | The Game cannot find the *GameSession* in the Scene |
| Error  Correction | *GameSession* prefab was moved in the Hierarchy |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 2 |  |
| Line Error | *Read/Write = true* |
| Error  Explanation | The particle system’s texture read/write needed to be enabled |
| Error  Correction | *Read/Write was ticked* |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 3 |  |
| Line Error | GameObject Laser = Instantiate(LaserPrefab, transform.position, Quaternion.identity) as GameObject; |
| Error  Explanation | The enemies that shoot had the same script as the ones which did not |
| Error  Correction | *A new script was created called EnemyShoot* |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 4 |  |
| Line Error | FindObjectOfType<GameSession>().ResetGame(); |
| Error  Explanation | The Game could not find ResetGame(); |
| Error  Correction | Made an if statement so that if GameSession is not equal to gameSession it will restart game |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 5 |  |
| Line Error | healthBar.maxValue = player.GetHealth(); |
| Error  Explanation | was not finding player |
| Error  Correction | Play Canvas was overridden so that it applies to all scenes |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 6 |  |
| Line Error | scoreText.text = gameSession.GetScore().ToString(); |
| Error  Explanation | The Game could not find gameSession.GetScore() |
| Error  Correction | Play Canvas was overridden so that it applies to all scenes |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 7 |  |
| Line Error | deathVFX not assigned |
| Error  Explanation | The Game can not find death VFX of the player when he dies |
| Error  Correction | The explosion was chosen for the VFX |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 8 |  |
| Line Error | Game can not find enemyPrefab variable in waveConfig |
| Error  Explanation | The Game can not find an enemy in the waveConfig |
| Error  Correction | The enemy was added in the empty wave |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 9 |  |
| Line Error | Game can not find Laser in EnemyShoot |
| Error  Explanation | The laser was not assigned on the enemy |
| Error  Correction | The Laser was set on the LaserPrefab |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 10 |  |
| Line Error | private void Hit(DamageDealer dmg) |
| Error  Explanation | The Hit function could not see any enemies to calculate the health |
| Error  Correction | The enemySpawn was set to looping is true |
| Error  Correction  ScreenShot |  |

|  |  |
| --- | --- |
| Test Case 11 |  |
| Line Error |  |
| Error  Explanation | The enemy was not being destroyed on inpact |
| Error  Correction | Polygon collider 2D was applied to the ambulance enemy |
| Error  Correction  ScreenShot |  |

Assignment Rubric:

|  |  |
| --- | --- |
| **Criteria and tasks** | **Marks** |
|  |  |
| **KU2: Describe asset types** |  |
| Arrange the Assets folder into subfolders having a folder for each type of file used in your game | **5** |
|  |  |
| **KU5: Identify suitable resolution for images of various types** |  |
| Use suitable images (Sprites) for: 2D Player Car, Road background, and 5 different types of Obstacles (example: bicycles, cars, trucks, barriers, stones, etc.!). | **2** |
| Set suitable resolutions and scaling for the above images. | **1** |
| Set all images as GameObjects and Prefabs. | **2** |
|  |  |
| **KU6: Select asset types and settings for a range of media assets for a game design** |  |
| Correctly import in the Assets folder different Audio files for:     * Player health reduction * Points gained when obstacle is avoided * Background Music | **1** |
| Setup the Camera to a 10:16 Aspect Ratio resolution | **1** |
| Using ViewPortToWorldPoint(), create a border around your camera | **1** |
| Set the background as repeated scrolling using Quad, Materials and Offset. The scrolling speed should be in line with the racing car speed. | **2** |
|  |  |
| **KU7: Identify resolution issues for the development platforms** |  |
| Upload Project on Github Repository (including Build folders, Game Test  Document) and share the link with your lecturer. You should have at least 12 commits over a span of at least 25 days to avoid development issues. The last commit should be before the deadline date and time, as otherwise it will not be considered. | **5** |
|  |  |
|  |  |
| **KU8: Identify appropriate input devices and methods for development platforms** |  |
| Make the player car move on the x-axis only within the border of the camera. Use Unity built-in methods to make the game frame-rate independent and to stop the car from going out of border. | **3** |
| Give proper Colliders or Triggers to Player Car, Obstacles, Obstacle Bullets and Obstacle Destroyer | **2** |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **KU9: Show and explain how to deploy a game to multiple devices** |  |
| Research how to build the game to:    PC, Mac, Linux StandAlone (.EXE file and other folders). Save the Build in the project **outside the Assets folder**in a folder named ***BuildEXE*** | **1** |
| Android (.apk file). Save the Build in the project **outside the Assets folder**in a folder named ***BuildAndroid*** | **2** |
| iOS. Save the Build in the project **outside the Assets folder**in a folder named ***BuildiOS*** | **2** |
|  |  |
|  |  |
| **AA1: Examine and apply basic programming techniques for a simple game** |  |
| Create a correct Path for each type of Obstacle (5 Paths in total). Paths should be made up of Waypoints. Paths should be set as Prefabs in a *Paths* folder. Use different Lists to save the Waypoints and the Paths for easier access. | **3** |
| Create an Obstacle Wave Scriptable file which contains:     * Obstacle Prefab to spawn * Path Prefab on which to move * Obstacle movement speed * Number of obstacles per wave | **2** |
| Create a DamageDealer class. This class should be implemented with different damage output for different obstacle prefabs as follows:     * Wave 1 Obstacles: 1 damage • Wave 2 Obstacles: 2 damage • Wave 3 Obstacles: 3 damage • Wave 4 Obstacles: 4 damage * Wave 5 Obstacles: 5 damage * Bullets from obstacles: 1 damage | **2** |
|  |  |

|  |  |
| --- | --- |
| **AA2: Develop a game with graphical and audio assets** |  |
| Start the Game from a Menu Scene with 2 Options, Play and Quit using  UI. When clicking *Quit* the Game ends. When clicking *Play*, go to the  Game Scene and start the Game | **1** |
| Background Music should start playing and keeps on going till the game quits. | **1** |
| Player starts with 50 Health Points. If an Obstacle or a bullet collide with the Player Car, the Player should have its Health reduced depending on the Damage done by the Obstacle | **2** |
| Obstacle collision Sound Effect is played. | **1** |
| Health Points and Game Points should be shown and updated in a UI Text object | **1** |
| Both the Winner and Game Over Scene should have options to go to Start Menu Scene. | **1** |
|  |  |
| **AA3: Demonstrate ability to resolve issues and input devices in the development environment** |  |
| An interview is held with your lecturer where you are tasked to answer any practical questions related to your work | **7** |
|  |  |
| **AA4: Apply coroutines for a more interesting gameplay** |  |
| Using coroutines and a timer your game should: Spawn all Obstacles in wave | **2** |
| 2 Obstacle Waves should fire a bullet | **2** |
| Spawn all Obstacle waves using Lists and foreach loop | **2** |
| Make Obstacle Waves restart from the beginning using Lists and foreach loop | **1** |
|  |  |
| **AA5: Examine and solve gameplay problems** |  |
| During Development you will encounter programming issues and gameplay problems. You are to document 14 **different** errors shown by the Console, the line error where you had the problems and a short explanation on how you solved the problem. Use the Game Test Document provided to document such information. Save the Game Test Document as a PDF and upload on Github. | **7** |
|  |  |

|  |  |
| --- | --- |
| **SE2: Resolve programming issues related to sound, graphics and gameplay** |  |
| Implement a *Level* script which takes care to load levels | **1** |
| When an Obstacle hits the Player Car, the Obstacle should be destroyed, an Explosion Particle effect applied and a proper Sound Effect is played. | **2** |
| Use Layers and Layer Collision Matrix to reduce Collision issues | **1** |
| If Player Health <= 0 and Player Game Points < 100, Player Car should be destroyed, an Explosion Particle Effect and a proper Sound Effect applied and load Game Over Scene with total points obtained shown. | **3** |
| Obstacles which are avoided by the Player, should be destroyed when they hit an Obstacle Destroyer at the bottom of the Scene. For every Obstacle avoided, the Player is awarded 5 Game Points. Avoiding bullets does not gain the Player any points. | **2** |
| If Player Game Points >= 100, the Game should stop and a Winner Scene should be loaded | **1** |
|  |  |
| **TOTAL MARKS:** | **75** |
|  |  |