|  |
| --- |
| PHYS1521  **Math and Physics for Games**  Project Report  Digital Media and IT  School of Applied Sciences and Technology |

|  |  |
| --- | --- |
| **Name(s):** | James Marr |
| **Section:** | DMIT 1521 Math and physics for games |
| **Date:** |  |
| **Instructor:** |  |

Table of Contents

[Introduction 1](#_Toc196397785)

[Overall project setup 1](#_Toc196397786)

[HardCoded 2](#_Toc196397787)

[ForceCoded 2](#_Toc196397788)

[CodePhysics 2](#_Toc196397789)

[Conclusion 2](#_Toc196397790)

[References 3](#_Toc196397791)

**Table of Figures**

# Introduction

For this research project, I have chosen to utilize my knowledge of math and physics in a unity game framework by creating multiple code types on how physics can be handled in programing. The types that will be covered in this report will be:

1: HardCoded: Implementation of all physics on a rudimentary level that is enough to work on close enough to the other coded types.

2: ForcePhysics: Utilizing unity’s physics engine built in and applying forces to the player object.

3: CodePhysics: Also utilizing unity’s physics engine but applying different methods of movement that aren’t applying forces.

These names for each type of physics used are self-named to keep track for the type of physics that will be talked about during this report.

The reason why I’ve chosen this as my research project is that I’ve always been interested in the various types of ways that movement is handled in various games. There are always advantages and disadvantages to each individual type a developer decides to choose for their game. I wanted to explore in depth for some of the shortcomings that each style has, well also learning some unique ways that can be done to overcome these challenges.

# Overall project setup

Here you will have several paragraphs outlining the concept(s) of your project. You will need to go into depth on your project, i.e. outline all the key points relevant to Game Programming. For each key point below be sure to include diagrams/figures along with any math/code that is relevant to the key point. The diagrams and figures will need to be updated in the Table of Figures.

The setup of the project will be using Unity 6000.041f1 for the game engine used for this project, the base project template will be using the universal core 2D template, most game objects used in the scene will contain an appropriate collider (e.g. box collider2D), rigidbody2D and a capsule represented as a player. In the GameState object it will contain the state of which physics type/modes that are being used at any given time. (HardCoded, ForcePhysics, CodePhysics) Each of the individual modes will influence some physics objects by changing their body type so that each of the individual physics will minimally affect the other modes, for example the left and right wall will switch on its trigger on their respective colliders well in HardCoded mode but turn it off well in any other mode.

The setup of the game scene will have 2 walls, a ground, a platform, a red trigger box that will work as a force to push left labeled “ConveyorBelt”, the scene camera and the Player capsule. By using unity’s new InputSystem, the player can move left, right and jump, this will allow the player to interact with the scene and the various objects. The majority of all the code that will affect the player will be in the “PlayerController” script to reduce the need of applying a lot more code across the setup project as a whole. Notable exceptions of what objects also have code are the walls to switch the collider triggers on and off based on the current mode.

## HardCoded

Section for Key Point 1 (you need to rename the heading for this sub-section).

## ForceCoded

Section for Key Point 2 (you need to rename the heading for this sub-section). Repeat this as many times as required for your report

## CodePhysics

# Conclusion

Summarize the report by restating the reason for this topic and how the key points (covered above) make this topic relevant to Game Programming.

# References

Put all your references here.