

Developer:
Osama Ibn Khalid
Mahmood

BSc (Hons)
Software Engineering

STREET COMBAT

Supervisor:
Misbahu Zubair

PROJECT OVERVIEW

Fighting games are considered one of the most played genres worldwide. Tekken is an example of it, and there are various features such as animations and user interface that make the game enjoyable in each version. The development of famous action games like Mortal Combat and Tekken 7 is done using Unreal Engine. However, Street Combat is a project developed using the Unity Engine. The outcome is a fighting game that is still exciting even if its main objectives are set to solve its technical difficulties rather than a game design solution.

OBJECTIVES

- Object Oriented based system design.
- Simple User Interface Design as the solution is based on technical aims rather game design objectives.
- Basic Procedural animations.
- Programming the game software in Visual Studio using C# for Unity development.
- Artificial Intelligence implementation for the computer characters using common AI techniques.
- Classic fighting gameplay between a Player and an AI character.
- Free mode gameplay experience.

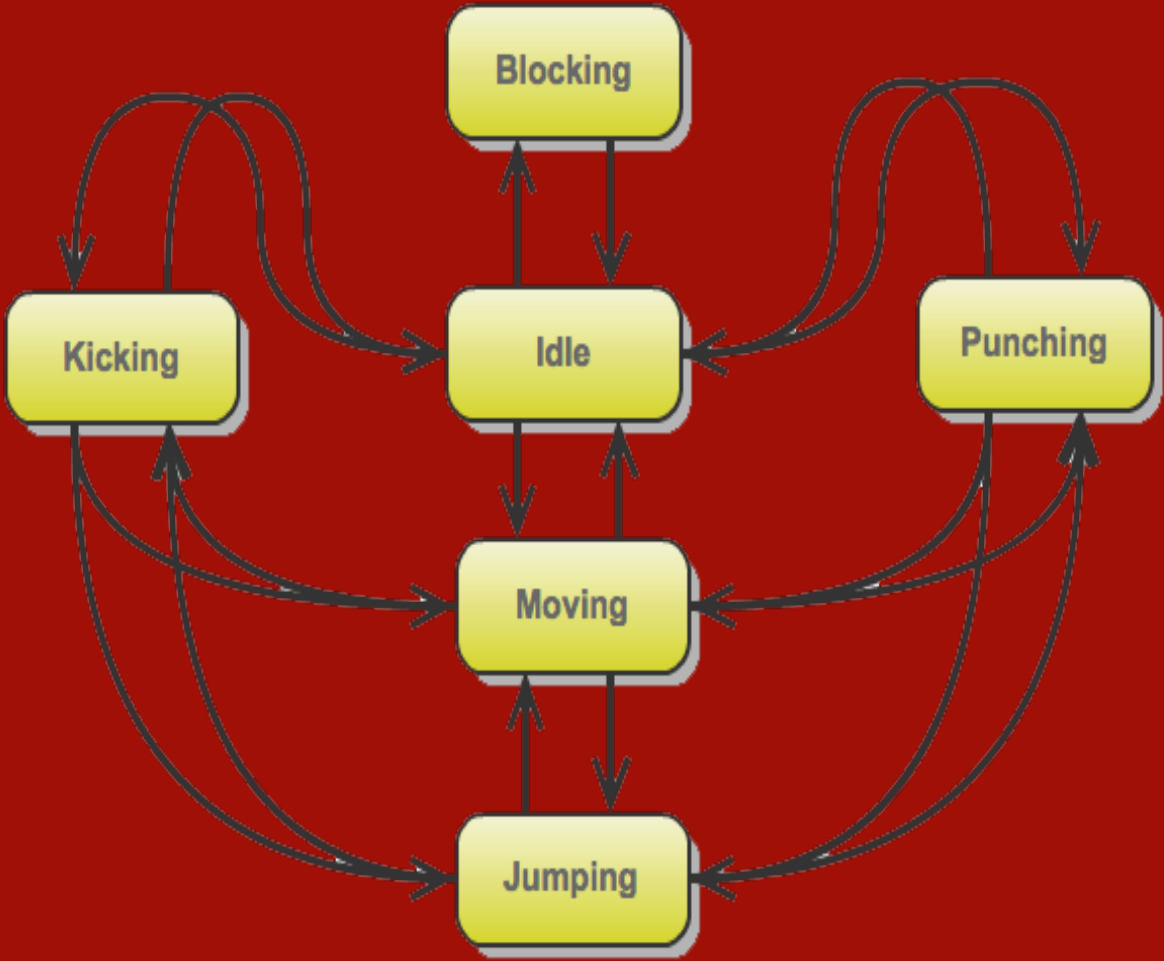
METHODOLOGY

Initially a Waterfall methodology was considered where each development stage is completed before stepping into the next one. This methodology doesn't allow any freedom in making changings to the existing project, so an agile approach has been rather used. MoSCoW is the agile methodology that has been used for the game development. This approach allowed me to prioritise my tasks according to different factors such as features, given timeframe, system requirements etc. Approaching MoSCoW allowed me flexibility in making appropriate updates, changings and improvements in the project in any stage of the lifecycle.

CHALLENGES

- Only basic experience in developing games using TL-Engine.
- Research about specific technical features in action games.
- Used and Learned Unity engine spending a severe amount of time.
- Advanced research and programming practices in C# language for Unity games development.
- No experience and Knowledge about Artificial Intelligence.
- Learning Artificial Intelligence.
- Decide suitable AI technique for my project.

AI FSM DESIGN



TECHNICAL COMPLEXITIES

- Object Oriented implementation approach to organize things in multiple classes assuring better readability and debugging.
- Object Oriented SOLID principles applied to organize better my classes and make the code reusable.
- Implementation of the non-player characters Artificial Intelligence using a Finite State Machine Technique (FSM).
- A Finite State Machine technique allowed me to define the characters' behaviours in different states of the gameplay.
- Implementation of different game difficulties. This means that a player can decide to play against either a difficult AI or an easier one.
- Implementation of AI simulations game modes where two AI characters competes each other and their difficulty is based on what difficulty a user selects when the game starts (default is standard difficulty).

TESTING

Due to the nature of the project, most of the project's testing was a case of applying a script to the scene in some way and seeing if the output was as expected. Therefore, most testing was performed while the project was being built. The testing strategy used is the Black box testing technique which doesn't require internal knowledge of the implementation. The main objective of this approach is that incorrect behaviours are identified, and performance errors are corrected accordingly. In my project, even after making minor changes in the program, the Game was debugged or compiled to assure that the expected behaviour was achieved.

MOBILE VERSION

The Android version still comes with serious bugs that are planned to be fixed in Summer. However, a basic gameplay can still be experienced.



FUTURE WORK

However, there is still a lot of work to do before deploying the game. Some important tasks as part of future work and personal interest are:

- More advanced AI Techniques implementation.
- More game modes such as survival mode and adventure mode.
- User login access.
- Mobile version of the game (due to time frame available only advanced task partially attempted).
- Improve design of the game and user interface by adding Particle effects and sounds on hit.
- More characters in the selection screen and different special attacks for each of them.

CONCLUSIONS

Most of the objectives that seemed reasonable to meet before the project's deadline were met; all of the 'Must', 'Should' and 'Could' objectives described in the planning of the project were met and tested. A basic android version has been made also allowing me to develop a significant project for multiple platforms. The overall game is about different game scenes. The first scene consists of a main menu where the user can select various options. There are mainly 3 game modes:

- Player one vs AI character.
- Player one vs a character in practice mode that can be moved by own preference.
- AI simulations that are demonstrations of two AI fighting each other.

For these modes the difficulty of the AI can be selected by accessing the settings screen from the main menu.

