MDU112.3 Practical Assignment
(Physics Based Game)

By

Name : Alwin Joshy
Student ID : 5000491

Game Theme.

This is a top-down tank based shooter game I made by combining the task for MUD112.3 and MDU113.3. Game called “Get Wrecked” due to its game play style, combat system and the quickness required to evade the enemy attacks. The game mechanics, controles are made very similar to generic tank control system.

Game Controls.

The utilises generic controls which can be mostly seen in simple 2D games run on pc.

R button = reset the game level,
J button = quit the application,
W button = forward acceleration,
S button = backward acceleration,
A button = turn left,
D button = turn right.
Mouse Button 1 = FIre button.

Things Learned.

On doing this project for MDU112.3 which is targeted towards the development of a fully functioning physics based Top-down shooter game. To achieve it we relied on the lightweight 2D librarie know as SFML .Net librarie.

Through the making of this game I was able to learn the and implement the fundamental as well as the core concepts of object oriented programing like inheritance, polymorphism, overriding and virtual functions.

In this project which is our final project the main goal is to make a top down shooter containing three enemy types a player capable of destroying those enemies by shooting at them which results in earning a score point corresponding to that enemy.

The game contains three types of enemies, they and their properties are given below

Type one :

- Slow moving with a speed of 50
- Does not shoot
- Does a damage of 3 to the player
- Takes a damage of 35
- Carries a score of 10

Type two :

- Fast moving with a speed of 150
- Does not shoot
- Does a damage of 6 to the player
- Takes a damage of 25
- Carries a score of 30

Type three :

- Fast moving with a speed of 150
- Does shoot
- Does a damage of 12 to the player
- Takes a damage of 15
- Carries a score of 60

The player movement is based on simple tank controls. The sprite is rotated on turning and a new forward axis is set. On pressing forward or backward a force is added to the rigidbody of the object which will be considered as motion in the physics section of the game engine.

The Game uses AABB collision to detect collision between sprite sheets. The collision system checks the least penetrated axis of the overlapping sprite and identifies the normal vector and the which further enables in calculating the counter acting of the force which should be applied on the colliding sprite. Physics based factors such as elasticity and mass are taken into consideration in the physics calculation as well.

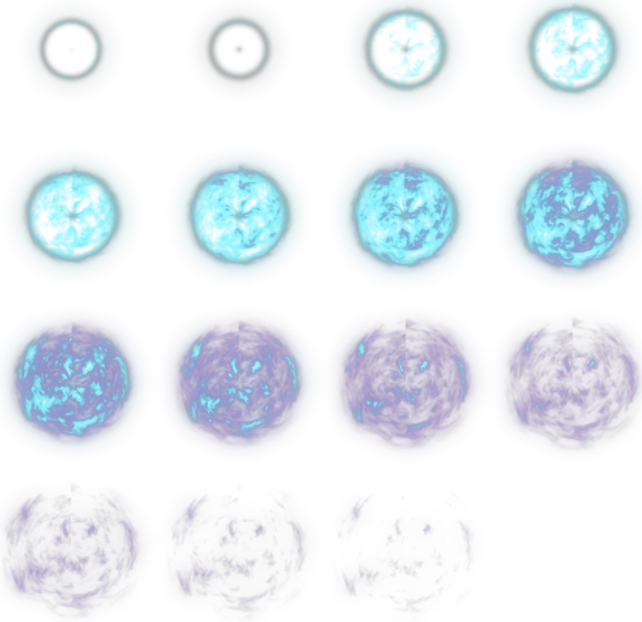
As mentioned earlier as i made my game as a combination of of both MDU112.3 and MDU113.3 assignment which enabled me to achieve A* path finding for the enemy entities. For to conserve the use of memory i have used pooling based bullets an enemy entity creation and destruction system.

Extra Achievements.

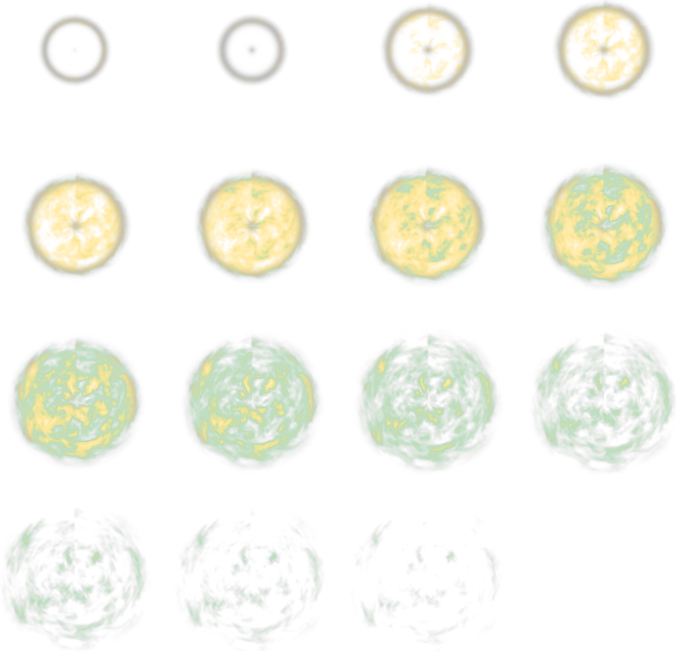
For to earn extra credits points I have achieved A* pathfinding for the enemy entities, use of costume sprites and have used frame based animated sprite based particle effects. The game play

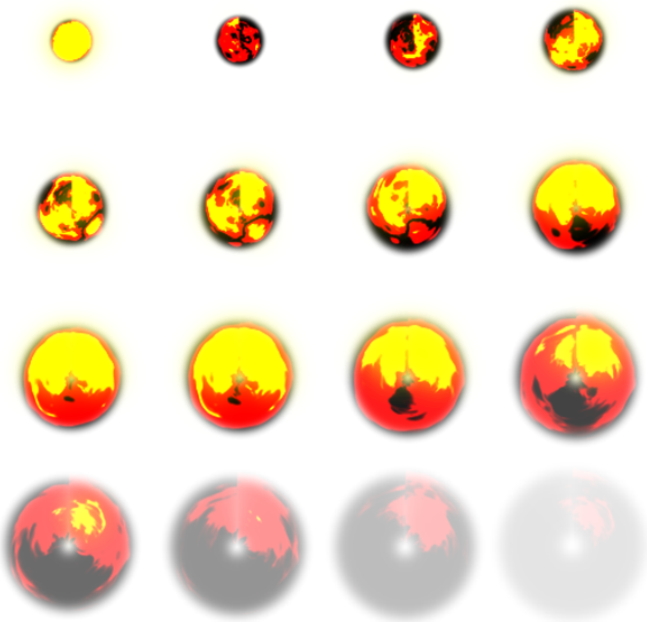
is accompanied by a free music as well which is called ‘"8 Bit Scrap!" Fast Fun Battle Music by HeatleyBros’

Player fire blast :

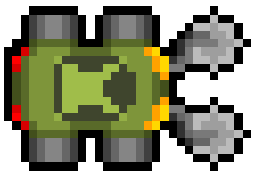


Enemy fire blast :

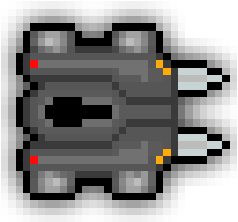




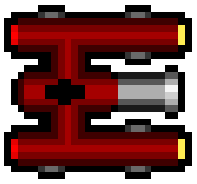
Vehicle explosion :



enemy one sprite :



Enemy two sprite :



Enemy three sprite :

References.

Adobe Inc.(2012). Photoshop CS6[Windows PC]. California : United States.

Adobe Inc.(2018). After Effects CS6[Windows PC]. California : United States.

Tissot, B. (n.d). 8 Bit Scrap.[Audio file]. Retrieved from https://www.youtube.com /watch?v=ay-ccm07XmU