# Design Document

Our game, going by the name of “Metal Dog 3”, will be a Shoot‘em up type of game in which the player controls a dog inside of a spaceship shooting all the evildoers of doghoodness. It’s an open-end arcade game in which the goal is to destroy as many enemies as possible and survive as long as possible. A striving for the highscore.

# Minimal Requirements:

## Player

For the final assignment we are going to make an interpretation of Shoot ‘em up. The player controls a dog and you have to shoot enemies of the canine. The player uses WASD to control the movement of the dog, and uses space to shoot.

## Enemies

The enemies come in various shapes ranging from vacuum cleaners to postmen. They can harm the pattern by colliding into them. Some of them have linear patterns, while some have random movement. Some enemies come in a swarm, and some are solo.

## Randomness

The spawning of the enemies is random.

## Animation

The shooting of the player’s weapon will have animation, as well as destruction of enemies. And if the player dies a death scene will start.

## Pause

The game can be paused by pressing p. And esc pauses the game and gives you a menu.

## Interaction with the file system

Things that will be read from the file system:

* Highscores

## Schedule

|  |  |  |
| --- | --- | --- |
| Week |  |  |
| 1 (22 – 28 October) | -Make a moveable player, which can shoot.  -Implement pausing of the game | -Learn how to draw graphics on screen  -Learn how to convert user input to movement of an object on screen  -Learn how to create projectiles  -Learn how to pause a game. |
| 2 (29 – 4 November) | -Implement enemies  -Implement highscore  -Implement end of game | -Make enemies appear at random intervals and locations  -If an enemy dies, add his reward to the score.  -When player dies, stop the game.  -If the players’ score is higher than the highscore, save the new highscore to a file.  -When the game is started, read a file containing the highscore. |
| 3 (5 – 11 November) | -Implement extra’s  -Bug fixing  -Polishing | TBC |

# Important data types and type classes

## Data types:

## Type classes: