

***SCHOOL OF ELECTRICAL, ELECTRONIC AND COMPUTER ENGINEERING***

***JetMan 2D Game***

***Computer Methods 3 Project***

**Semester 1: 2015**

**Ebot-Monchey Ojong-Mo’Ebot 213512989**

**Vuyolwethu Ngqaqhu 213571439**

**Langelihle Ngobese 213571644**

**Nontokozo Ndlovu 211509006**

**Sabelo Gabela 213551986**

# Introduction

A robot with a jet is passing through a ruined city with endless obstacles. Guide it safely through the obstacles and see how far you can go!

# Software development methodology

An agile approach was taken towards the development of the game because it is difficult to determine the full requirements of the game upfront. Changes to the game’s mechanics were likely to happen and an agile approach easily incorporated those changes. Git was used for version control and to allow collaboration among the team members. The repo for the project was hosted on Github.com and all the team members where added as collaborators.

# Software architecture

The code is separated into three namespaces named JetMan::Utils, JetMan::Graphics, and JetMan::Game.

The Utils namespace contains utility classes that provide useful functionality. This includes the SoundManager class which handles loading and playing of sound files, and the ImageManager class which handles the loading of images.

The Graphics namespace contains the classes that deal with the visual aspect of the game. The Rectangle class is one of the most important as it describes a way to simply represent the location of a 2D object on the screen and determine whether such objects intersect. The abstract Displayable class uses this notion of location and adds a method for objects to render themselves to the screen. This is the base class of all graphical components which are able to be displayed. The abstract class Widget extends the Displayable class and allows for a visual object to respond to mouse interactions. More classes such as Label and Button are derived from the Widget class to present a way to more easily create graphical user interfaces containing menus. The Sprite class adds the ability for a visual object to move about the screen. It inherits from the Widget class simply to allow derived sprites to be added to containers for widgets like buttons and labels (actual widgets). Game related classes (JetManSprite, Wall) are derived from the Sprite class to allow them to be displayed and moved about the screen.

The Game namespace contains the main Game class which instantiates all the utilities required, creates the graphical user interface, registers event handlers and runs the game loop. The game loop is a loop that runs continuously receiving events from the window (to close), mouse, keyboard, timer, and then updating the state of the objects in the game. The logic of the game is implemented after the state is updated: bounds checking, collision detection, re-spawning walls, score updating etc. Finally all the visual components are rendered and the cycle repeats.

# Discussion of team member’s contribution

* Langelihle (213571644) created the ImageManager and InformationBox classes and was involved in the design and acquisition of resources to produce the user interface.
* Nontokozo (211509006) created the SoundManager class and was involved in the design and creation of visual artwork resources to produce the user interface.
* Sabelo (213551986) created the classes Rectangle, Sprite, and JetManSprite.
* Vuyolwethu (213571439) created most of the graphical components that were used to build the user interfaces.
* Ebot-Monchey (213512989) implemented most of the Game class.

# Instructions to play the game

The instructions are straightforward. Guide the robot passed the obstacles by tapping the spacebar (key which boosts the robot by a little amount), holding the spacebar key for about half a second (which boosts the robot up by a lot more), or allowing gravity to pull the robot downwards.

# Key resources consulted

The Allegro 5.0.11 reference manual and the Allegro 5 API tutorials from the Allegro wiki site were invaluable resources in developing this game.

# Game URL

The game code and resources can be found in the repo at:

<https://github.com/Ebot-Monchey/JetMan>

# Credits

1. City background by Gage Herrmann accessed at <http://blog.gageh.us/sun-gun-city-background/>
2. Music available in the public domain