Plan of Attack  
ISGPKBS

**Group 3**  
Wendy Neijenhuis  
Janita Veenstra  
Dominik Lewy  
Erik Peters  
Job van der Toom

Contents

[Introduction 3](#_Toc254083887)

[Assignment 4](#_Toc254083888)

[Phasing 5](#_Toc254083889)

[Concept phase 5](#_Toc254083890)

[Design phase 5](#_Toc254083891)

[Implementation phase 5](#_Toc254083892)

[Testing phase 5](#_Toc254083893)

[Milestones 6](#_Toc254083894)

[Tasks 7](#_Toc254083895)

[Planning 8](#_Toc254083896)

# Introduction

This document is our Plan of Attack for the KBS of the Games Programming minor. In this document we shall mainly describe in which phases we shall create our game and a planning. This planning will contain what will be done when, and by whom.

# Assignment

Our assignment for this period is creating a 2d side scrolling game. The game must have at least three levels which are created with tiles. The main character has to move around the levels, getting from the left to the right and we have to make it fun to play.

In the first period of this minor, we have to create a 2D side scrolling game. Our game has to exist of at least three different themed levels, which we must create with tiles, which we must ‘import’ from a large bitmap, the ’level map’. One of the things we can do, is animating the tiles, by using different bitmaps. Each level has to be three screens wide (we thought with a resolution of 800x600px, this would be at least 2400px) and one screen height (600px then).

The main character of our game must walk around in some directions, like left and right. Also he must be able to jump. Besides that we have to implement some physics so our character, and the other objects, will fall down when they are not supported by objects where they can stand on. The falling must happen natural. Besides this, the character must face in the direction he is walking and the movement must be handles by the arrows of the keyboard. Also, when hitting things, we have to detect the collision, so you can’t move further (boundary collision) , get some points when hitting a gadget, die when hitting a ghost or something changes in the world. When alive and not entering or leaving a world, the main character has to be in the centre of the screen. When he dies and his lives are depleted, the game has to end.

For the background we must create depth with the use of parallax scrolling, using at least two background layers. Before each level and before the game starts or ends, there must be a splash screen showing some of the theme which comes. After the splash screen of entering the game, there must be a menu where the user can load this last game and see his highscores, when in game the user also can save his game here.

In the levels there must be some kind of gadgets around; these gadgets will make up the game play of our game. With different gadgets for different purposes we can change how the game is played. Besides, with this use of gadgets, we can make the game appealing to play. The enemies in the game will walk around in an adjustable range and they will be aware of their environment (example).

The game has to use a state machine pattern. The game constants are saved in a configuration file called ‘params.ini’. Besides, the enemies must have some AI behaviours.

# Phasing

We will divide our project in the following phases:

* Concept phase
* Design phase
* Implementation phase
* Testing phase

## Concept phase

In this phase we will create a game design and a little prototype. The game design will be a document that describes the story of the game, the levels, the enemies and the gadgets/items in the game. We will also create some designs for the characters, items and levels.

## Design phase

In this phase we create the architecture of our game and continue with designing the characters, items and levels in the game. After creating the architecture we will decide what we want to build for our first iteration and this part will be designed in a class diagram.

## Implementation phase

In this phase we will demonstrate an iteration of our game weekly. This will be done by determining what we want to create in the next iteration, create and update the design and implement.

## Testing phase

At the end of our project we will test our game if there are any bugs.

# Milestones

This project contains the following milestones:

|  |  |
| --- | --- |
| **What?** | **Finished when** |
| Plan of Attack, Functional Design, State Diagrams and Class Diagram | End of week 3 |
| Level and Character rendering + Sounds | End of week 4 |
| Moving character and gadgets rendering + picking up | End of week 5 |
| Enemies: rendering + behaviour | End of week 6 |
| Menu, settings, highscores, multilevel support | End of week 7 |
| Intro screens, game intro, saving and loading | End of week 8 |
| Installation CD + Testing and bugfixing | End of week 9 |
| Testing and bugfixing | End of week 10 |

# Tasks

|  |  |
| --- | --- |
| **Task** | **By whom** |
| Plan of Attack | Job & Janita |
| Functional Design | Erik & Wendy |
| State Diagrams | Everyone |
| Class Diagram | Everyone |
| Level Rendering | Erik & Wendy |
| Character Rendering | Job & Janita |
| Sounds | Dominik |
| Moving character | Job & Janita |
| Gadget rendering | Erik & Wendy |
| Gadget pickup | Dominik |
| Enemies rendering | Everyone |
| Enemies behaviour | Eveyone |
| Menu | Janita |
| Settings | Job |
| Highscores | Dominik |
| Multilevel support | Erik & Wendy |
| Intro screens | Janita |
| Game intro | Job |
| Saving and loading | Erik & Wendy |
| Installation CD | Job & Janita |
| Testing and bugfixing | Everyone |

# Planning