

# Game Design Document of “The Fantastic Quest”

## Contents

---

<b>Outline of game concept</b> .....	1
<b>Game Design discussion</b> .....	2
<b>Game’s mechanics Description</b> .....	3
<b>Player interface</b> .....	6
<b>Advanced features</b> .....	6
<b>Audio Choice</b> .....	8
<b>Game challenge</b> .....	8
<b>Test and evaluation</b> .....	9
<b>Planning and organisation</b> .....	10
<b>Conclusion</b> .....	11
<b>Reference list</b> .....	11

## Outline of game concept

---

The principal idea of this game is to blend two different types of game genre altogether, the space shooter and RPG. This could be quite hard as both genres belong to two completely different play styles and this could result in a confused game where you have different systems within the game. The concepts were born from the inspiration of a retro game called King’s Knight (Square 1986), a game that pursue this hazardous path. Attracted to this curious game, I created something that uses its core dynamics and added my personal twist and vision.



Fig.1 Box art of King's Knight



Fig.2 Gameplay example

The game follows a simple storyline, but the player is not really exposed to it. The entertainment merits can be identified in the gameplay where we have, for example, a progress bar in the first level and a score system. These two elements and the gameplay itself can be identified as the main entertainment features.



Fig.3 Progress bar



Fig.4 Score

The main objective from the player's perspective will be to level up the stats, reach the end of the level and defeat the boss.



Fig.5 Stats

## Game Design discussion

The visual style of the game is pixel art used in the 8 to 16-bit era. It is a vibrant and simplistic style. The theme of the game is fantasy/medieval.

The level design has two structures, in the first level we progress through a forest and the gameplay is about progression and survival. At the boss level, the main purpose is the destruction of the dragon.

There are three playable avatars, depending on what we choose we have different spells and different stats.

- **Warrior:** this character has l2 attack and defence as warriors are usually physically stronger than other classes. In addition, we have the spinning sword and shield abilities.



Fig.6 Warrior Sprite

- **Wizard:** this character has all the stats at l1 but in return for this sacrifice we have many powerful spells. A magic explosion and water blast.



Fig.7 Wizard Sprite

- **Thief:** this character has a l3 dexterity which will make him move quicker. As special abilities, we can set a trap or throw a bomb.



Fig.8 Thief Sprite

Enemies types:

- **Basic:** that goes towards.



Fig.9 Basic sprite

- **Shooter:** it shoots bullets.



Fig.10 Shooter sprite

- **Follower:** it points to the player's direction.



Fig.11 Follower sprite

- **Side-walker:** this will appear from the right side and will move horizontally.



Fig.12 Side-walker sprite

## Game's mechanics Description

- **Stats and levelling:** the stats represent the level of growth of a character into a certain skill, we have Attack is represented by the rate of shooting the bullets. The defence is how many hits a character can get before death. We have Dexterity indicates how quickly the character moves in order to avoid any hazards.

System	attack = 1	Add action
Mouse	Left button is down	System
System	Every 0.4 seconds	Audio
		Create object <b>shoot</b> on layer 1 at (player.X, player.Y -30 )
		Play <b>shoot1.0</b> not looping at volume 0 dB at object <b>player</b> ( inner angle 360, outer angle 360, outer gain 360 dB) with tag ""
		Add action
System	attack = 2	Add action
Mouse	Left button is down	System
System	Every 0.3 seconds	Audio
		Create object <b>shoot</b> on layer 1 at (player.X, player.Y -30 )
		Play <b>shoot1.0</b> not looping at volume 0 dB at object <b>player</b> ( inner angle 360, outer angle 360, outer gain 360 dB) with tag ""
		Add action
System	attack = 3	Add action
Mouse	Left button is down	System
System	Every 0.2 seconds	Audio
		Create object <b>shoot</b> on layer 1 at (player.X, player.Y -30 )
		Play <b>shoot1.0</b> not looping at volume 0 dB at object <b>player</b> ( inner angle 360, outer angle 360, outer gain 360 dB) with tag ""
		Add action

Fig.13 Stats application

- In order to level up a stat the player must pick a determined power up.



Fig.14 power ups

- In addition to that we also have health points that report the vitality and the magic/ability points that represent the usage left of those. These are rechargeable by pick-ups.



Fig.15 bars

- **Why did I choose that:** as the player is progressing the game she will be motivated to play as there is an actual character progression and meaningful choices (evaluative) in the game that affect the gameplay.
- **Combat mechanic:** the player can move left and right and shoot bullets in front of her, depending on the character chosen we have two different special attacks.
  - **Why did I choose that:** the player will have dynamic and responsive gameplay driven by a descriptive meaningful play which responds in real time to the player's actions. (Salen K. Zimmerman E. 2004)



Fig.16 Shooting

One of the most remarkable aspects of the event sheet can be identified in the state machine which plays a major part in this game, for example, we have a state machine that indicates the parameter of the player depending on the stats. If the attack is level 2 the bullet will be shot every 0.3 seconds and so on. There is also a state machine for the boss fight that will tell the boss what action to take in a determinate time.

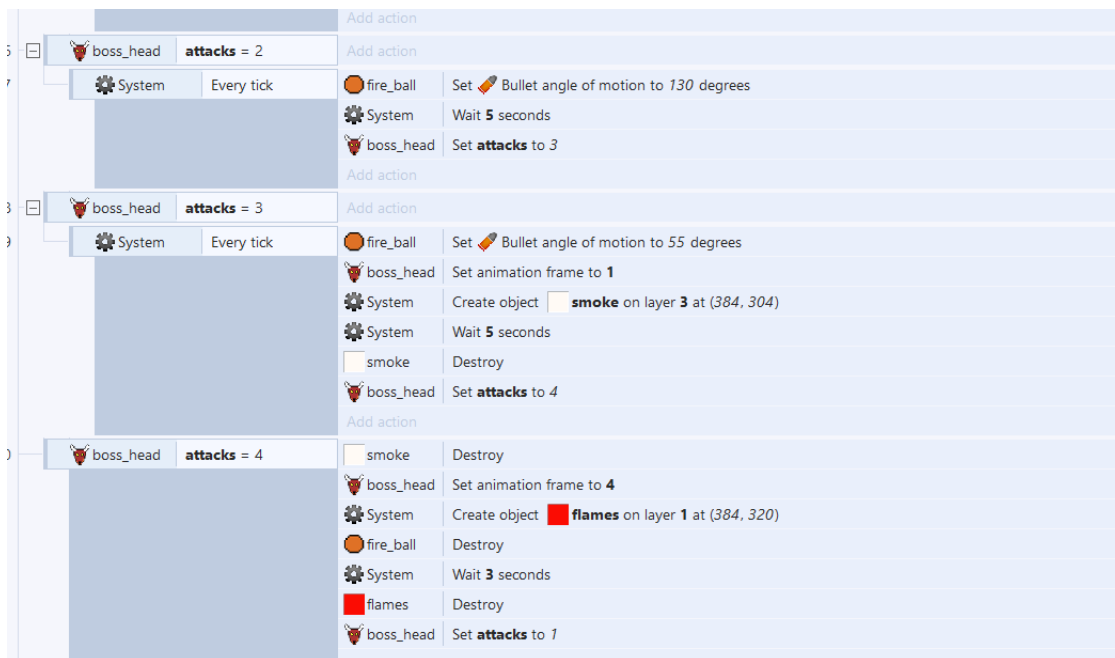


Fig.17 state machine

## Player interface

The menus are kept to a minimum in order to keep the player focused on the gameplay. We have the initial menu and the game over pop-up screen that gives you the possibility to restart.



Fig.18 menu

Fig.19 death screen

- **Heads-up display:** while playing there always be a constant feedback present in the bottom of the screen where are kept all the useful information constantly refreshed such as health points, lives left and current score.



Fig.20 HUD

All the user interfaces are designed to follow the general visual style of the game so they are made appropriate with the theme.

Also, the controls are kept to the minimum, this will allow the player to understand and master them immediately.

We have the left hand for the movements and the right hand for the actions with the mouse, so the player doesn't need to move the hands around.

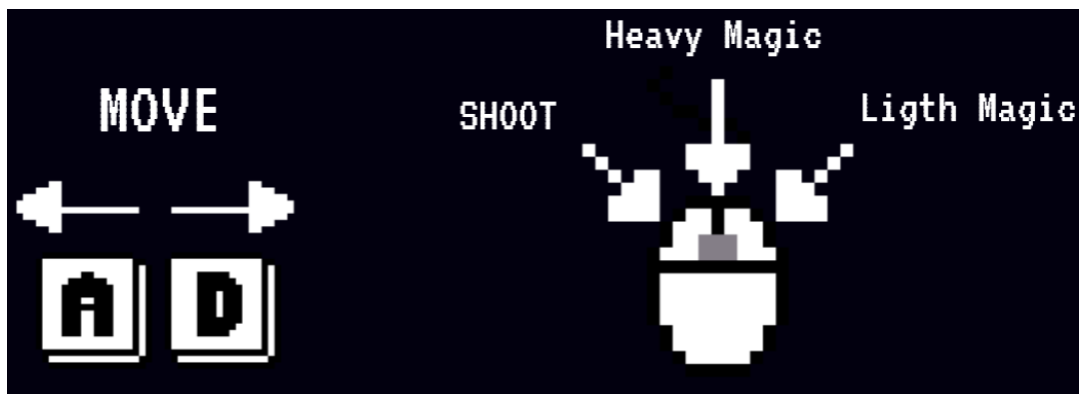


Fig.21 Control scheme

## Advanced features

One of the bits where there is artificial intelligence can be identified is as the follower enemy. This enemy will spot and point the player slightly before appearing in the layout. A slightly similar behaviour happens with the shooter's enemy.



Fig.22 follower



Fig.23 bullet

Another piece of AI can be seen in the boss fight where we have these flames enemies that will respawn if they are destroyed unless the player uses the water spell that can be only used with the wizard.



Fig.24 Flame-enemy

Particles effects: we have an explosion with a 360 radius where we have different sizes depending on the object. Then we have the abilities where different results by changing the properties; the fire spell which works similarly to an explosion with a 180 radius etc.



Fig.25/26 Spells examples

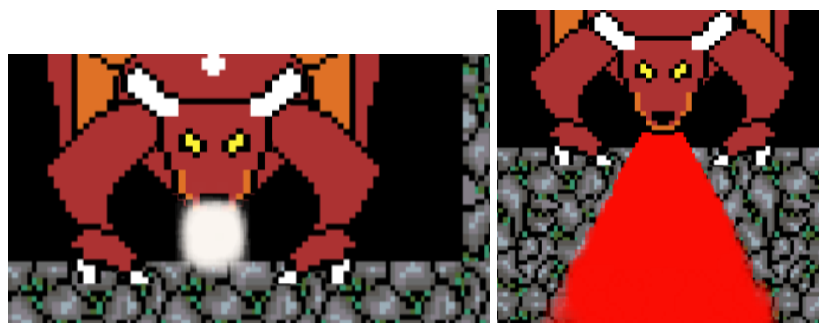


Fig.27 Dragon's Breath



## Audio Choice

---

All the sounds implemented into the game were designed or chosen to follow the visual style, in this case it has been chosen something that has to go alongside with pixel art. Nothing can be more suitable than 8 to 16-bit music in this case.

For level one we have one battle theme, dynamic and with a quick pace that encourages the player to battle.

Then, the boss fight theme has an initial escalation which gives more pressure as we are in front of a very dangerous creature and the player must feel that.

The sound effects present in the game are used to enhance feedback and transmit an appropriate feeling depending on the situation. For example, the explosion sound is used to increase the satisfactory feel of defeating an enemy.

In the case of the boss fight, the dragon's actions are enhanced by the sounds, this gives major importance to the character from the player's perspective.

## Game challenge

---

The score is shown in the hub and increases depending on what you destroy/hit. It will be also shown at the end of the game as a reminder of how well the player did throughout the gameplay.

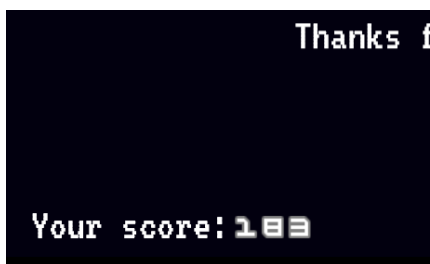


Fig.28 End game screen

In the first level, the building up of the character represents a training phase where he starts with minimum levels and easy enemies to handle, in fact, most of the enemies go down with few shots.

Then we have the boss stage, the final stage where all the skills learnt previously and the stats acquired are needed to defeat the dragon, the difficulty increases exponentially as we have now a massive threat which is much more health than anyone else in the game.

By scoring 1000 points the player is awarded one life. In this way, the player is encouraged to defeat more enemies.

## Test and evaluation

---

The game has been tested with the iteration method several times by trying to meet all the possible cases that the player could find while playing, in this way I could guarantee a pretty solid player experience. In a nutshell, the development was creating, testing and adjusting.

## Feedbacks

Tester	Problem	Fix
Paulo Faria	Score bug, it doesn't reset when the player dies.	When a player has 0 life score goes to 0.
Paulo Faria	The HP and MP bars will increase without a limitation.	When those points surpass 200, they will be reset to 200 again.
Daniel King	When press esc and replay the game won't reset the characters.	When the player presses "esc" all global variables are reset.

The game was analysed with also the debugger in order to check if the framerate was kept over 60, as there were some cases where there was not proper optimization of the game which caused an overload of the system; this was generated by some spare instances that weren't destroyed after the use in the layout.







Frames per second	60 (16.7 ms/frame)	
Est. CPU utilisation	10%	
Est. image memory	8.2 mb	
Renderer	webgl	
Object count	46	
Collision checks/sec	0 (~0/tick)	

Fig.29 Debugging

## Planning and Organisation

The major part of the game design was to combine well the characteristics of these two genres, in order to succeed, there were different designs that simulated different approaches.

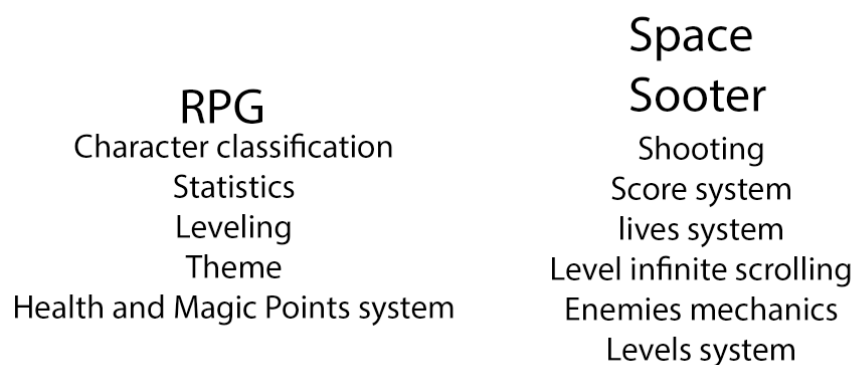


Fig.30 Genres qualities

The first design considered had one character with all these different abilities altogether, and he had also to “learn” through the game. But this would result in confusion as there would be lots of controls to learn.

All the designs had to follow a few different focal points in order to pass into the creation. One of them is certainly the immediacy as the main idea is to give a quick pace and the learning curve of the game.

The organization of the implementation was separated into stages that follow a repeated cycle likely to an agile model (Ambler S. 2002) used in the software development life cycle, we have the design of the idea, and then we transform this idea into a mechanic and finally test this mechanic, if it goes well with the rest of the game.



Fig.31 Agile diagram

## Conclusion

Along with the making of the assignment I have learnt how to create a sprite in pixel art and animate it accordingly, how to produce and modify soundtracks in order to fit the circumstances, how to approach an object-orientated language better and how simplification of a game sometimes is better in order to have a better playability.

With more time I would insert more depth into the game by implementing more stages, levels and special abilities, a world map with different biomes, a dialogue system, special items and an efficacy system depending on what character is used.

## Reference list

---

Agile diagram - associazioneprism.eu - project-management.png [Online Image]. Available from: <http://www.associazioneprism.eu/methodologies/> [accessed 10/12/2018].

Ambler S., 2002 *Agile modeling: effective practices for extreme programming and the unified process*. 1st Edition. New York: John Wiley & Sons

EliteFerrex 2010, 8-bit Boss Battle 4: The Rival, Available From: <https://www.newgrounds.com/audio/listen/367084>, [Accessed 10/11/2018] **Boss theme**

HLG 2017, Serious Battle!!, Available From: <https://www.newgrounds.com/audio/listen/759452>, [Accessed 10/11/2018] **Forest theme**

Jeti 2018, Alkhemikal, Available From: <https://www.dafont.com/FR/alkhemikal.font>, [Accessed 9/11/2018] **Font in the screen title**

Pushbar 2010, Victory (8-bit tune), Available From: <https://www.newgrounds.com/audio/listen/330100>, [Accessed 13/11/2018] **Victory theme**

Salen k. Zimmerman e., 2004, *Rules of play: game design fundamentals*. 1st edition. London: the MIT press.

Workss, 1986, King's knight [video game], Nintendo entertainment system, Japan: square