

Quest for the Crown 2

Game Concept & Design Document

By

Diogo Muller de Miranda

João Vitor Pietsiaki Moraes

Melanie Young Yee

**Document Contents**

[1 Introduction 1](#_Toc362106078)

[2 Game Concept 2](#_Toc362106079)

[2.1 Introduction 2](#_Toc362106080)

[2.2 Game Analysis 2](#_Toc362106081)

[2.3 Game Atmosphere 4](#_Toc362106082)

[2.4 Game Play 4](#_Toc362106083)

[2.5 Key Features 5](#_Toc362106084)

[2.6 Selling Features 5](#_Toc362106085)

[3 Design Document 6](#_Toc362106086)

[3.1 Design Version 6](#_Toc362106087)

[3.2 Design Guidelines 6](#_Toc362106088)

[3.3 Game Design Definitions 6](#_Toc362106089)

[3.4 Game Matrix 6](#_Toc362106090)

[3.5 Game Flow Chart 7](#_Toc362106091)

[3.6 Player Elements 7](#_Toc362106092)

[3.6.1 Player Definition 7](#_Toc362106093)

[3.6.2 Player Properties 7](#_Toc362106094)

[3.6.3 Player Rewards (Power-ups & Pick-ups) 8](#_Toc362106095)

[3.6.4 User Interface (UI) 8](#_Toc362106096)

[3.6.5 Heads up Display (HUD) 8](#_Toc362106097)

[3.6.6 Player View 8](#_Toc362106098)

[3.7 Antagonistic Elements 8](#_Toc362106099)

[3.7.1 Antagonistic Definitions 9](#_Toc362106100)

[3.7.2 Antagonistic Properties 9](#_Toc362106101)

[3.7.3 Antagonistic List 9](#_Toc362106102)

[3.7.4 Artificial Intelligence (AI) 9](#_Toc362106103)

[3.8 Global Game Elements 9](#_Toc362106104)

[3.9 The Story 9](#_Toc362106105)

[3.9.1 The Story Copy 9](#_Toc362106106)

[3.10 Concept Art 10](#_Toc362106107)

[3.11 Level Design 10](#_Toc362106108)

[3.11.1 Level Copy 10](#_Toc362106109)

[3.12 Audio & Sound F/X 10](#_Toc362106110)

[3.13 Game Architecture 10](#_Toc362106111)

[3.13.1 Game Architecture Overview 11](#_Toc362106112)

[3.13.2 Architecture Copy 11](#_Toc362106113)

[3.13.3 How to play Copy 11](#_Toc362106114)

[4 Technical Document 12](#_Toc362106115)

[4.1 System Requirements 12](#_Toc362106116)

[4.2 Visual Content 12](#_Toc362106117)

[4.3 Audio Content 13](#_Toc362106118)

[4.4 Programming Content 13](#_Toc362106119)

[4.5 Code Structure 13](#_Toc362106120)

[4.6 Concerns and Alternatives 14](#_Toc362106121)

[4.7 Resources 14](#_Toc362106122)

[4.8 Technical Matrix 14](#_Toc362106123)

[5 Appendix A 15](#_Toc362106124)

[5.1 Taxonomy 15](#_Toc362106125)

[5.2 Genre 15](#_Toc362106126)

[5.3 Game Elements 16](#_Toc362106127)

[5.4 Content 16](#_Toc362106128)

[5.5 Theme 16](#_Toc362106129)

[5.6 Style 16](#_Toc362106130)

[6 Appendix B 17](#_Toc362106131)

[7 Appendix C 1](#_Toc362106132)

[8 Appendix D 1](#_Toc362106133)

# Introduction

This document is based on the Game Design Document template by [ACHAR INFORMAÇÕES SOBRE O TEMPLATE NO EUREKA].

It describes the Game Quest for The Crown 2, and is divided in three parts: the game concept document, game mechanics document and game tech document. The original writer did this with the intention of making a modular document that can be updated and split into different versions easily. We simplified this division as chapters, so we can keep the document concise and direct.

[ADICIONAR MAIS?]

# Game Concept

## Introduction

Quest for the Crown 2 is a 2D adventure/exploration medieval open world game, with dual-stick controls. It takes elements from older exploration games, with dungeon exploration, different items with varied effects on enemies and objects, puzzles based on those items, and adds the second stick aiming, changing the combat present on this style other games.

The basic story of the game is that a group of cultists kidnapped the king, and with him was the magic crown from the kingdom. Our hero is the king’s bodyguard, a knight that could not save the king, and now departs on his quest to get stronger and rescue the king and the crown.

The game exploration consists on different screens, using a top-down perspective, visited by the player. It is based on older Zelda games, with a bigger number of enemies and changed combat, consisting of one attack button (for the equipped weapon) and one stick/group of keys that represents the weapon direction. The behavior of the weapon direction stick/buttons can change, depending of the weapon. On some weapons, it will act as aim, on others it will do an attack on that direction, or even act as a way to control the weapon.

## Game Analysis

This is a general overview of the game.

|  |  |
| --- | --- |
| **Game Description** |  |
| Genre: | * + Role-play   + Adventure   + Puzzle   + Action |
| Game Elements: | * + Shooting   + Collecting   + Combat   + Dodging   + Solving Puzzles   + Exploring |

|  |  |
| --- | --- |
| Game Content: | * + Adventure   + Exploration   + Action |
| Theme: | * + Fantasy   + Medieval |
| Style: | * + Old School   + Cartoon |
| Game Sequence: | * + Non-Linear Storylines   + Open world   + Free exploration |
| Player: | * Initially, one. Local multiplayer is still a possibility. |
| **Game Reference** |  |
| Game Taxonomy: | * Fictional Game/Narrative. |
| Player Immersion: | * + Narrative   + Physical   + Mental |
| Reference: | * Classic Zelda-style overworld and dungeon exploration, with different items that have different effects. * Dual-stick attack controls, based on games such as Geometry Wars, Super Stardust, and so on. * Open world exploration, with independent dungeon order, such as Elder Scrolls. |

|  |  |
| --- | --- |
| **Game Technical** |  |
| Technical From: | * 2D |
| View: | * Top-Down |
| Platform: | * C#, Monogame |
| Device: | * PC, MAC, Linux, Ouya |

## Game Atmosphere

The game world is an open and vast territory, with varied kinds of areas, each with its own mood. The basic game areas, for now, consist of:

* Hero Fields/Hero Shine: Open field areas with weak enemies, bright graphics and open maps.
* Goldensand Beach/Beach Cave: Open areas with water and long range enemies.
* Stoneman Caves: Irregular, dark areas with fast, but weak enemies, like bats.
* Orc Forest: Full of enemies and narrow passages. Lush, green maps.
* Haunted Woods: Dark forest, more open than Orc forest, but full of undead enemies such as zombies and skeletons.
* Fort Avalanche: Full of enemies, a mix of open areas and narrow passages, consists of gray walls and stone floors. Some parts of the fort where invaded by the beach.
* Wizard’s Tomb: Great open, dark areas, full of undead enemies, and elements that represent the wizard’s power and wealth, such as prisons, treasures, weapons and so on. Think of a mix between a castle dungeon and a Pharaoh Pyramid.
* Stoneman Mountains: Mountain pathway, with bright light, but dirty colors. Big number of ranged enemies.
* Dark Valley: Dark area, with few, but strong enemies. Mostly rocks and darkness.
* Cultist Tower: A dark tower, with lots of cultists – medium range, teleporting enemies – a mix between narrow and open areas, lots of traps.

A more detailed map of the game can be seen on Appendix A.

## Game Play

The initial game screen has a few options that the player can choose: New Game, Load Game, Credits, How to Play, Options and Quit Game.

Credits shows the game Credits. How To Play show the game basic controls. Quit Game quits the game.

Options shows the game options: Game resolution, if the game plays full screen or not, Inverted Aim for the bow and an option to delete all the saved data.

New Game starts a new game, while Load Game starts the game from a save point.

When the player starts a new game, the game shows the story to the player player: The king was kidnapped with his magic crown, and you, his bodyguard, is searching for him.

The first map contains no enemies, only a save point, a dungeon entrance and another exit to the map, distant from the player. The dungeon contains only weak enemies, one weapon (the sword) and the weakest boss, the slimeworm (that’s actually a line of slimes that can divide themselves in smaller slimeworms). The first dungeon was planned this way so the player could learn how to play, by playing. The sword is activated by pressing the Attack Button, or by pressing a direction on the attack stick/keys.

If the player ventures outside the dungeon, south of the initial area, he’s presented with the hero fields, and is free to go anywhere from there. Some areas have stronger enemies, other areas have faster enemies, and some have a bigger number of enemies. The player is free to decide what he considers the easiest way to proceed.

Other weapons that the player can get are:

* Bow and Arrows: The player aims with the attack stick/arrows, shoot with attack button. Costs Arrows.
* Fire Wand: The player shoots a fireball to the direction he is looking at, using the attack button. Costs Magic.
* Boomerang: The player can shoot the boomerang with the attack button, and control it for a short period, using the attack stick/arrows.

Each dungeon contain one item and one boss, and after the first dungeon, the item is always guarded by a sub-boss that attacks you with the it. The dungeon boss is weak to the dungeon item, and it’s easier (but not necessarily obligatory) to defeat him with it. The dungeon puzzles can be solved using only the dungeon item.

The player has a few status: Health, the remaining life points the player has, and magic, that shows the basic player magic status and recharges one point per second. Also, some weapons can have their own ammo. The bow, for example, uses arrows.

The game ends if the player loses all of their health points. In this case, the player is taken to a game over screen and can return to the main menu, where it can load a previous saved game.

The player wins the game if he defeats the cultist leader and rescues the king and the crown. In this case, the player is shown a cutscene, the game’s ending.

## Key Features

* Five Dungeons.
* More than 50 different maps.
* Four types of weapons.
* 6 types of enemies.
* 5 different bosses.
* 3 different sub-bosses.
* About 2-3 hours on the first gameplay.
* Can be replayed on different dungeon orders, or not using one or more kinds of weapon.
* One Player.
* PC/MAC/Linux/Ouya.

## Selling Features

* Old-school exploration game
* Different Items, Enemies, Dungeons
* Vast open world to explore
* Great for old-school game fans

The game uses only free, open or creative commons free for commercial use assets.

# Design Document

This document describes how game objects behave, controlled and properties they have.

This documentation is primarily concerned with the game it’s self.

This part of the document is meant to be modular. Meaning you could have several Game Mechanic documents attached to the Concept Document.

## Design Version

A version can single out a certain series of devices that may have limitations, different OS or more advanced features.

A code convention for different versions would be advisable.

*Example:* Such as **J1.1**

(J): (JAVA) Developed for a particular Technology

(1.): Concept Update

(.1): Content Update

## Design Guidelines

This is an important statement about any creative restrictions that need to be regarded and a general over all goal of the design.

## Game Design Definitions

This is a section where the definition of the game play is established.

Definitions should include how a player wins, loses, passes levels and the main focus of the game play.

Issues that should be addressed here are:

* Menu
* Synopsis
* Game Play
* Player Control
* Game Over (Winning & Losing)

## Game Matrix

This is a spreadsheet containing the generic names of the player and antagonistic elements and their game properties.

This should allow an easy cross reference for an elements in the game that a value.

Consult with the programmer about the properties a game object.

*See Appendix B*

## Game Flow Chart

This is where a visual of how the different game elements and their properties interact.

Game Flow chart should represent Objects, Properties and Actions that are present in the game.

Flow chart objects, properties and actions should have a number reference to where they exsist with in the game mechanics document.

*See Appendix C*

## Player Elements

List all the elements that are directly related to or to the benefit of the player.

Devise two sets of names for player elements. One set is a generic name (or code) and the other is its game name.

Describe the terminology that you use to describe the player’s properties.

This is a good place to interact with a graphic designer to get graphics to match to game names.

Graphics that will be seen during game play should be exhibited here.

Multi-player issues should also be mentioned here.

### Player Definition

Make quick descriptions that define the player.

What are the default settings for the player at the beginning of the game or level.

A suggested list of player definitions:

* Actions: What can the player do?
* Information (Status): What information about the game is available for the player?
* Default Properties: How does the player begin the game?
* Winning: How can the player win?
* Loosing: How does the player lose?

### Player Properties

Make a list that defines the properties that a player has.

Player properties can be affected by player’s action or interaction with other game elements.

Define the properties and how they affect the player’s current game.

A suggested list of player definitions:

* Health
* Weapons
* Actions
* Etc.

Each property should mention a feedback as a result of the property changing

### Player Rewards (Power-ups & Pick-ups)

Make a list of all objects that affect the player in a positive way. (i.e. health replenished)

Define these objects by describing what affect they cause and how the player can use the object.

### User Interface (UI)

This is where a description of the user’s control of the game can be placed.

It is also recommended to think about which buttons on a device would be best suited for the game.

Consider what the worst layout is, then ask you self if your UI is it still playable?

A visual representation can be added, where we relate the physical controls to the actions in the game.

When designing the UI, make use of the expertise of someone from quality control.

### Heads up Display (HUD)

This is where a description of any graphics that will represent information during game play should be described.

A visual representation (mock-up screenshot) here would be useful.

This is a good place to interact with a graphic designer.

### Player View

A screen shot is very necessary here.

A definition of how the camera moves for the player.

A mock-up of an overview of the level relative to the screen size will help create a perspective of a levels size compared to what is actually seen.

## Antagonistic Elements

This is where a list of antagonistic (enemies, opponent) objects should be listed with graphics and written description.

Describe the terminology that you used to describe antagonistic properties.

Devise two sets of names for player elements. One set is a generic name (or code) and the other is its game name.

This is a good place to interact with a graphic designer to get graphics to match to game names.

### Antagonistic Definitions

This where a description of what makes a antagonistic element.

### Antagonistic Properties

This is a list of properties that antagonistic elements have in common.

### Antagonistic List

This is where a list of all the antagonistic elements goes.

### Artificial Intelligence (AI)

This is where visuals and written description(s) of the antagonistic element’s behaviors.

These should be labeled in such a way that they can be used in level design with out having to describe them again.

Devise generic names for repetitive behaviors.

This is how an AI action could be broken down:

* Normal State: What is the object doing if it has not come in contact with the player?
* Detection State: What does it take for this object to detect the player?
* Reaction State: What does the object do as an action after passing the reaction state?
* End State: What happens to the object after player has reacted correctly or incorrectly to object?

## Global Game Elements

In this section it is important to describe the boundaries, neutral objects, camera views and scale of the world.

Neutral game world objects can be things like a static background, objects that do not interact with the player or antagonistic elements.

## The Story

This is where the story can be described in detail.

A story board can be used to tie in graphics to the text.

This can later be used for splash screen concepts.

### The Story Copy

A shorter version of the story (The in game version) should also be written here.

This is where the script for in game characters or story information during the cut scenes would be placed.

This category does not always pertain to the current Game Design.

## Concept Art

Sketches that are used for the concept can go into this section as visual reference.

In the case of a brand, certain creative restrictions should be noted here.

This is a good place to interact with a graphic designer to get graphics to match to game names.

## Level Design

This is where information pertaining to level design and visuals of the level design goes.

Level design can best be shown as a flow chart.

Use generic names to create level design.

### Level Copy

This is where the script for in game characters or story information during the cut scenes would be placed.

## Audio & Sound F/X

This is where game ambient and Sound F/X should be listed with generic names then described.

This section also includes deciding if you will use a device’s vibration ring mode.

## [Game Architecture](file:///C:\Users\Current%20Game%20Ideas%20&%20Designs\Game%20Doc%20Template\options.htm)

This is best done by a flow chart to represent the overall game.

Number each screen.

* Title Screen
* Option Screens
* Game Modes
* End Screens

*See Appendix D*

### Game Architecture Overview

The splash screens or video clip need to be in accordance to game story and style.

If cut scenes use video then story boards should be created.

This will need to be created with the graphic designer.

Menu should be designed with the most important options easily accessible.

Be aware how many clicks it takes to accomplish a task.

It would be wise to get together with someone from quality control and a programmer.

The Game Instructions should be written so that the player understands how to play the game.

Mock-ups should be made so that the programmers get the correct layout of the menu.

Mention and describe high score screen here as well.

### Architecture Copy

All text fro the game can be complied here.

Review Game Architecture Overview

### How to play Copy

This will be game copy.

Information for the player, clearly describing how to play the game.

# Technical Document

The information concerning the technical aspects of the game should be placed here.

The technical document is best achieved with consensus from the people responsible for the Visual, Programming and Audio aspects.

This part of the document is meant to be modular. Meaning you could have several Game Technical documents attached to the Concept Document and Game Mechanic Document.

## System Requirements

This is a list of system requirements that a device will have to meet to run the game.

This also represents the restrictions that may apply to the end product.

## Visual Content

A list of technical requirements from those in concerned with the visual aspects of the game.

This is a section that will require extensive meetings with a graphic designer.

All objects should be listed with their generic names.

* General
  + File Size Restrictions
  + File Format Type
  + File Quality Type
  + Visual Scale
* Player Elements
  + Type of States (Default, Damage, Destroyed, ect.)
    - Amount Animation Frames
* Heads Up Display (HUD)
  + Type Icons
  + States
  + Font Type
* Antagonistic Elements
  + Type of States (Default, Damage, Destroyed, ect.)
  + Amount Animation Frames
* Global Elements
  + Background/Texture/Tiles
  + Font Type

## Audio Content

It is very important to communicate with the audio designer before and while the audio content is being developed.

* General
  + File Size Restrictions
  + File Format Type
  + File Quality Type
* Player Elements
  + Type of Sound f/x
  + Device Vibration
* Antagonistic Elements
  + Type of Sound f/x
  + Device Vibration
* Global Elements
  + Ambient Music
* Splash Screens
  + Ambient Music
* Menus
  + Type of Sound f/x

## Programming Content

Contents should be collaboration with the programmer.

The object here is to try to organize and modulate as much as possible.

* General
  + Requirements
  + File Size Restrictions
  + File Format Type
  + Specify Coding Conventions
  + Language/Device Restrictions
  + Screen Type (Small, Medium, Large)
* Player Elements
  + Type of Event
* Antagonistic Elements
  + Type Event
* Global Elements
  + Type of Event
* Splash Screens
  + Type of Event
* Menus
  + Type of Event
  + Type of Options

## Code Structure

This is where an overview of how objects/functions/data interact, a list of what specified functions/routines do and a list of what order modules will be written.

## Concerns and Alternatives

If there are concerns about something technical they should be stated here and what will be alternatives to the concern.

## Resources

A list of applications and equipment, that is acceptable for use on development of this game.

This is mostly a legal issue that development members must be aware of.

## Technical Matrix

The Matrix will be split into the different device series for each content category.

Technical Matrix includes the content lists of Audio, Visual and Programming.

# Appendix A

[PUT GAME MAPS HERE.]

In this appendix there are lists for a games Taxonomy, Genre, Elements, Contents, Theme and Style. These lists are far from complete but are useful for further examples.

## Taxonomy

* Simulation
  + Story
  + Play
  + Chance
    - Fiction
    - Non Fiction
* Story
  + Simulation
  + Play
  + Chance
    - Fiction
    - Non Fiction
* Play
  + Story
  + Chance
  + Simulation
    - Fiction
    - Non Fiction

## Genre

* Basic
  + Adventure
  + Arcade (any "twitch" element)
  + Construction & Management
  + Puzzle
  + Role-play
  + Simulator
  + Strategy
* More Specific
  + Abstract (i.e., Arkanoid)
  + Adventure
  + Arcade (very generalised category)
  + Beat-em-up
  + Flight Sim/Space Sim
  + FPS (first-person shooter)
  + MMOG(Massive Multi-player Online)
    - MMORPG (role playing game)
    - MMORTS (real time strategy)
    - MMOTBS (turn base strategy)
  + Platform
  + Puzzle
  + Racing
  + RPG (role-playing game)
  + RTS (real-time strategy)
  + Shoot-em-up (scrolling shooter)
  + TBS (turn based strategy)
  + Trading

## Game Elements

* Alignment
* Catch
* Chase
* Collecting
* Combat
* Cooperation
* Dodging
* Escape
* Fighting
* Forbidden Act
* Hiding
* Jump
* Luck
* Maze
* Nursing
* Obstacles
* Puzzle
* React
* Resource Management
* Seeking
* Shooting
* Story Comprehension
* Target
* Trading
* Trivia
* Timing

## Content

* Action
* Drama
* Erotic
* Horror
* Humor
* Pure Play
* Realism
* Thriller

## Theme

* Abstract
* Crime
* Fantasy
* Noir
* Porn
* Sci-Fi
* Spy
* War
* Western

## Style

* Abstract
* Cartoon
* Manga
* Old School
* Realism

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Appendix B  |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Object** | **Properties** | |  |  | | *Unit type* | *Hit Points* | *Damage* | *Bullet speed* | *Movement* | | *Basic Enemies* |  |  |  |  | | Cannon -01 | 30 | 100 | Medium | NA | | Cannon -02 | 60 | 150 | Slow | NA | | Static rocket launcher | 70 | 700 | Medium | NA | | Flock 1 | 15 | 500 | NA | Medium | | Flock 2 | 15 | 35 | Slow | Medium | | Medium 1 | 80 | 350 | Fast | Slow | | Medium 2 | 90 | 400 | Medium | Slow | | Rail | 120 | 600 | NA | Slow | | Small 1 | 25 | 200 | Fast | Fast | | Small 2 | 30 | 210 | Fast | Fast | |  |

|  |
| --- |
| Appendix C image004 image003 |

# Appendix D

