Initial Game Concept & Design Document Template

## Introduction

A short sentence or two about the game, its genre, player type, technical form, references and theme.

Everyone that reads this should be able to understand what the basic idea of this game is.

## Game Analysis

This is a general overview of the game. Fill out the right column.

|  |  |
| --- | --- |
| **Game Description** |  |
| Genre: | *FPS* |
| Player: | Single |
| Game Elements: | Game elements are the basic activities the player will be doing for fun during the game. Example: shooting, combat, dodging, chase, etc |
| Game Content: | *Example: H*orror , Thriller , Humor , Drama |
| Theme: | *Example :* Western, Sci-Fi, War, Fantasy |
| Style: | *Example:* Real, Manga, |
| Game Sequence: | *Example:* Linear- Storylines, Hyper- Storylines that the player can influence, Simulation |
| Player Immersion: | This is an attempt to understand what kind of enjoyment the player will receive from the game. *Example:* Tactical, Strategy, Narrative, Physical, Emotional, Mental |
| Technical From: | Basically there is 2D graphics (Flat) and 3D graphics |
| View: | Camera view the player will experience the game from |
| SW Platform: | *Example:* Java, C++, Windows |
| HW Platform: | *Example* : PC, Mobile, Console |
| Audience | *Example* : General, PG, PG13, High school students, adults only |

## Game Atmosphere

This is where it is best to have a mood board or a clear description of the game’s style.

* Atmosphere Mood Board
* Character/ Units Sketch & Description
* A Level(Locations) Sketch & Description
* Audio Description

## Game Play

Provide descriptive paragraphs about how the game is played. The idea is that you want the person imagine they are actually playing the game.

Do not use Generic names when writing about the game play.

*Example:* No one wants to here that enemy\_1 will have more hit points than enemy\_2. Instead we should talk about how the Lazarus Fighter has more armour than Apollo Fighter.

This outline will vary according to the type of game.

* Opening the game application
* Game Options
* Story Synopsis
* Modes
* Game Elements
* Game Levels
* Player’s Controls
* Winning
* Losing
* End

**5. Key Features**

Key features are a list of game elements that are attractive to the player.

* Number of Levels
* Number of Enemies/ Characters *(Example: 12 characters or amount of enemies, end bosses)*
* Time of Game Play *(Example: 2 hours of fun)*
* Replay ability
* Audio Specifications
* Graphic Specifications
* Device Compatibility
* Number of Players
* Online Activities (high scores, etc.)
* Number/Type Modes

## Details of Game Design

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.

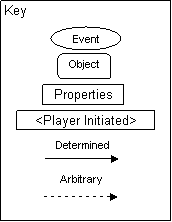
Issues that should be addressed here are:

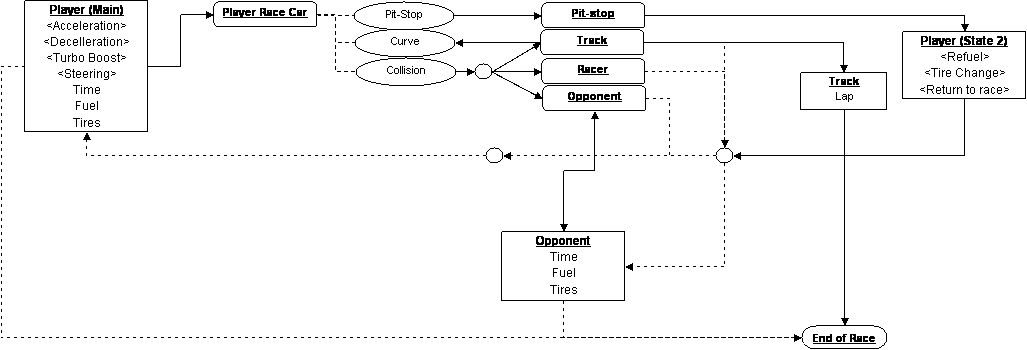
* Menu
* Synopsis
* Game Play
* Player Control
* Game Over (Winning & Losing)

## Game 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 exist with in the game mechanics document.





## Player Elements

### 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 properties:

* 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)

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.

Game_Design_Figure14

In general it displays Scores, Resource levels, Mini Map, Chat, Alerts and Level.

### Player View

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.

### Key map or control table

Provide a diagram showing control input, action, and context. An example is below:

Game_Design_Figure12

## Antagonistic Elements

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

### Antagonistic Definitions

A description of what makes an antagonistic element.

### Antagonistic Properties

A list of properties that antagonistic elements have in common.

### Antagonistic List

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.

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?

### An example describing the player/antagonistic element

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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 |

## Global Game Elements

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 if you have one. A story board can be used to tie in graphics to the text.

## 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.

## 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.

## Audio & Sound F/X

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

## [Game Architecture](file:///C:\Documents%20and%20Settings\rvtg-hrst-plg-c\Mijn%20documenten\Main%20Documents\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

## arch_flow_full

## 7. Technical Document

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

## 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. 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

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

## 8. Development timeline

**Timeline for the entire game**

Decompose the game development into sub-tasks and provide weekly schedule. You team can propose a grand game idea but specify what and how much your team can complete by the end of the quarter.

**Task distribution**

Describe who is in charge of which task.