

# [Insert Game Name]

# Technical Design Document

Version [Insert Version Number]

## Version History

[illegible]

## Table of Contents

<b>Game Overview .....</b>	<b>3</b>
Game Summary.....	3
Platform .....	3
<b>Development Overview .....</b>	<b>3</b>
Development Team.....	3
Development Environment.....	3
<i>Development Hardware</i> .....	3
<i>Development Software</i> .....	3
<i>External Code</i> .....	3
<b>Game Mechanics .....</b>	<b>4</b>
Main Technical Requirements.....	4
Architecture .....	4
Game Flow .....	4
Graphics .....	4
Audio .....	4
Artificial Intelligence.....	4
Networking (If Applicable).....	5
Physics (If Applicable).....	5
Game Objects and Logic .....	5
Data Management and Flow.....	5
<b>User Interface.....</b>	<b>5</b>
Game Shell.....	5
Play Screen .....	6
<b>Technical Risk .....</b>	<b>6</b>

## Game Overview

This section provides an overview of the game, suitable to someone new to the project.

### Game Summary

Provide a one or two paragraph overview of what the game is about. This material can come from your pitch, or can be something completely new for this.

### Platform

Describe the hardware/software platform that the game is going to be played on. This can come from the corresponding section of your pitch.

## Development Overview

This section discusses key aspects of the development of the game, as opposed to the game itself.

### Development Team

List the team members and their roles during the development process.

### Development Environment

This describes the environment used to develop the game. In theory, someone could set up an environment like what is described here and everything needed to work on the game.

#### Development Hardware

Provide a brief description of the hardware used during development. Provide hardware specifications as well as operating systems, and anything relevant that way.

#### Development Software

Provide a description of any tools used during development, including IDEs, compilers, editors, debuggers, and so on. If you are using any version control software, backup software, and so on, mention that here too.

#### External Code

Provide a description of all of the code used but not development by the team. This includes engine code, libraries, 3D APIs and drivers, and so on. A discussion of the content pipelined used to bring art and sound assets into the game would be useful as well.

## Game Mechanics

This is the main part of the document. This goes through the architecture and implementation notes on the main subsystems and control elements of the game.

## Main Technical Requirements

This provides a description of the key technical requirements imposed by the game's design. This will include critical functionality as well as constraints on this functionality (like performance, and so on). Some of these issues can be discussed in the other subsections under Game Mechanics.

## Architecture

In this section, you provide a description of the game's architecture, including all of its modules and the interactions between them. This should be done in text, as well as using one or more diagrams to depict the architecture.

## Game Flow

This section provides a discussion of how control flows through the game. This includes a description of the main control loop of the game and how it interacts with the various parts of the game, as well as the various different states that the game can be in (title screen, menus, playing, paused, and so on, perhaps with sub-states). Again, diagrams should be provided.

## Graphics

This section discusses the 3D/2D graphics elements of the game. Focus on the core aspects as opposed to interface elements, as these will be discussed later. (This would include rendering, sprite management, and so on.) If there are any constraints on the type of content that can be used here, it should be discussed.

## Audio

This section discusses the sound elements of the game, including music playback, sound effect playback, and so on. If there are any constraints on the type of content that can be used here, it should be discussed.

## Artificial Intelligence

This section discusses the control of all non-player active entities in the game. Identify how this control is done (scripting, state machines, rules, and so on) and provide details for each entity.

## Networking (If Applicable)

If there are online features to the game, they should be discussed here. This includes a description of protocols used, and other technical details.

## Physics (If Applicable)

If physical simulation is used, for collision detection, object interactions, and so on, a discussion of what is done and how is provided here.

## Game Objects and Logic

This is a discussion of all gameplay objects and logic used to support the game, as opposed to the various core subsystems discussed above. (One way of thinking about it is that these are all the game specific elements to your game ... the other aspects described in the subsections above could theoretically be provided by an engine, and might be, depending on your game.)

This should include subsections for each main game object or set of objects.

## Data Management and Flow

Game data will need to be loaded and saved at various points in time. This section should describe the procedures used to do loading, saving, and any processing of this data (such as compression/decompression, encryption/decryption, parsing, and so on), as well as the various different types of files used to contain this information.

## User Interface

This section describes how information is presented to the player, and how input is received in return. This includes display layout, feedback, controls, and so on.

## Game Shell

This discusses all screens presented to the player outside of game play. If there will be sound, music, or tactile feedback in any of these screens, it should be discussed. Controls should also be discussed to describe how screens are changed or how interface elements (buttons, etc.) are manipulated by the player.

The game will likely have menus given to the player to control the flow of out-of-game activities. This includes setting game options, controlling game flow (starting, stopping, pausing, and so on), and a variety of other tasks. The various menus must be identified, as well as the flow between them, and circumstances under which they appear and disappear.

Additionally, other screens may be presented to the player. This includes splash screens with logos and/or legal notices, credit screens, and so on. These should also be discussed.

Diagrams should be provided both as concept art and to illustrate screen flow.

## Play Screen

This is a discussion of what is presented to the player during game-play. This includes the depiction of all game elements, as well as other information presented in the game's heads up display (HUD). Any audio and tactile feedback provided should also be discussed. Diagrams should be provided as concept art to illustrate how the game screen appears during play.

In addition, a discussion of all controls should be provided. Not only should this describe what each control is and how it impacts gameplay, but this should also include how control events are processed and disseminated to the various subsystems and game objects as described above.

## Technical Risk

In this section, you discuss the various elements of technical risk associated with the development of your game. This can include things such as unfamiliarity with the platform or development tools being used, complex aspects of the game, other constraints on your schedules or resources, and so on. Each risk should be accompanied by a strategy to minimize, reduce, or mitigate the risk, to ensure that you will be able to handle the challenge appropriately.

Each risk (or set or related risks) can be discussed in their own subsection.