Oregon State University
College of Engineering

# CS Capstone Requirements Document

December 3, 2019

# DreamZBox 2.0

Prepared for

## Oregon State University EECS

Megan McCormick
Corinna Brown

Prepared by

## Group29
## CS29 Dreamz Catcherz

Haofeng Tian
Cory Hayes
Maxwell Evdemon
Tristan Hilbert

**Abstract**

The *CS29 Dreamz Catcherz* figured out the required functional necessities for the software of a video game console. They attach the required features to a game titled, *Maze of Dreamz*, and a dashboard titled, *DreamZDash*. For an entertaining experience, the *DreamZDash* must launch any game and modify system settings. Additionally, the *Maze of Dreamz* must use all of the buttons on the controller. These features will be tested through code and documentation. The goal is to finish these features within about 20 weeks of the project start date.

## CONTENTS

## List of Figures

# 1 INTRODUCTION

The *DreamZBox 2.0* is a hardware system with portable components. It is a home use video game console. Additionally, it remains battery powered, allowing individuals to transport the hardware while in use. The users are given controllers with a joy-stick, buttons, and accelerometer to interact with the consolidated hardware. A software system is desired to provide the entertainment aspects for the *DreamZBox 2.0*. This software system must adhere to certain goals and functionalities listed in this document. The overview, requirements, and metrics are provided herein.

## 1.1 Purpose

The *DreamZBox 2.0* uses cost-efficient hardware to provide an achievable entertainment experience for a largely diverse audience. The software must allow the *DreamZBox 2.0* to remain accessible to a wide audience. Consumers will want to retain a simple and entertaining experience when working with the hardware. Stakeholders seek use of a video game on the system. Providing a portable video game console with proprietary controllers, the console needs an innovating game to demonstrate the hardware. The software seeks to support *DreamZBox 2.0* in marketing, usability, and distribution.

## 1.2 Scope

The software system will provide a user dashboard internally called *DreamZDash*. This will allow usability of the console hardware itself for technicality. It will also provide a platform for a set of video-game applications. Additionally, the software will be packaged with a game built into the user's library. This game will be called *Maze of Dreamz*. The software will seek to provide usability for the largest possible extents of the system. It will provide use of all peripheral inputs to the system. The software will not provide systems level accessibility based on security and time constraints. The software will not provide usability for any audience beyond the non-technical users. The software will provide a user-tested entertaining experience.

## 1.3 System Overview

### *1.3.1 System Context*

*DreamZDash* will represent the first experience users will see and hear upon powering on the system. It will abstract the system. It will provide launching of applications like *Maze of Dreamz*. Alternatively, *Maze of Dreamz* provides graphical animation based on user input. The game will reflect a top-down maze adventure game. The user will play as a singular entity and attempt to reach a conceivable end while avoiding obstacles within the game. The singular entity will have virtual actions that will help progress it forward, including movement. Each map for the top-down maze game will be procedurally generated, in order to try and give a unique experience every time the game is played. After completion of the maze, the player will achieve victory.

### *1.3.2 System Functions*

The dashboard should provide ample user interface to reflect possible user questions about system status. The dashboard should remain fully accessible through the controller. The joy-stick, being a push-in joy-stick, should provide full traversal and selection. The dashboard will look aesthetically pleasing and denote system state, including battery charge. When *Maze of Dreamz* is launched, the joy-stick will facilitate movement while the buttons facilitate different assigned abilities. The abilities and movement will allow surpassing of the objectives within the maze. Each ability will light up

the lights on the controller a different color. The user will have to use the abilities to achieve victory. Additionally, *Maze of Dreamz* is a multiplayer game that will work with two controllers, one for each player. In the case of multiplayer the second player will function similarly to the first player, as they have full movement and abilities. The two players will share the screen view between their entities. Finally, the user may pause the game at any time with a possible menu button.

#### *1.3.3 User Characteristics*

The only user supported by this software is end-users or players of the device. The possible users per console is two. These users will have to use the system without immediate oversight or help. The portable nature of the system does not restrict it to a specific location, but most likely it will reside in an entertainment system or livable space.

### 1.4 Definitions

#### *1.4.1 top-down*

A game style that is meant to feel as if the player is looking down upon the playable map and controllable player characters. When viewing a top down map is typically a two dimensional area that where all the game-play takes place.

#### *1.4.2 procedurally generated*

Components within a application that generate when the process is initiated. These components use forms of patterns and statistics to provide randomized experiences.

#### *1.4.3 joy-stick*

A joy-stick or joy stick is an analog form of input which supplies data across a two dimensional plane. Depending on the direction this component is pushed, it provides a vector in that given direction.

#### *1.4.4 throbber*

A graphical swirl or animation that denotes operations occurring in the background. Necessary for large amounts of data pipe-lining.

#### *1.4.5 frame*

A singular render or expression of graphical change. A still image.

#### *1.4.6 LED*

Light Emitting Diode: these devices are tiny lights that work with current.

#### *1.4.7 USB*

Universal Serial Bus: In this document the term is used in place of a memory stick or ”USB stick”. These are portable devices that can plug into accessible ports allowing data transfer and storage.

## 2 System Requirements

The following sections list the necessary requirements and necessary attributes for the software of the *DreamZBox 2.0*.

## 2.1 Functional Requirements

### *2.1.1 For* DreamZDash

- Shall interface through the Bluetooth controller.
- Shall launch a game
- Shall show a battery life
- Shall display images and graphics
- Shall show changeable settings
- Shall modify settings based on user input
- Shall provide a software power switch

### *2.1.2 For* Maze of Dreamz

- Shall use 5 buttons, a push-in joy-stick, a trigger button, a scroll wheel, LED lights, and an accelerometer
- Shall display images and graphics
- Shall modify displayed images based on user input
- Shall light up LED lights based on game state
- Shall provide a different action or ability for each button and joy-stick
- Shall provide menu transversal and ability usage with the accelerometer

## 2.2 Usability Requirements

### *2.2.1 For all of the* DreamZBox 2.0

- A player should be able to use the *DreamZBox 2.0* controller without discomfort.

### *2.2.2 For* DreamZDash

- Shall look aesthetically pleasing
- Shall use readable text
- Should remain simple and intuitive
- Should have use without explanation
- Should answer all potential user questions
- May include a throbber for loading content

### *2.2.3 For* Maze of Dreamz

- Will not require the player to contort their hand in an unnatural way
- Should engage the player
- Should encourage the player to use all controls

## 2.3 Performance Requirements

### *2.3.1 For* DreamZDash

- Shall pre-load images for games
- Shall display at correct resolution for potential devices
- May protect against setting changes harming battery life

### *2.3.2 For* Maze of Dreamz

- Shall operate at more than a frame per second
- Will operate at the same rate with multiplayer

## 2.4 System Interface

- *DreamZBox 2.0* shall connect to devices through Bluetooth wireless
- *DreamZDash* may provide games through wireless internet also known as WiFi
- *DreamZDash* and *Maze of Dreamz* alongside other games will have an initial function to start their processes.

## 2.5 System Modes

- *DreamZBox 2.0* shall have 3 modes
- *DreamZBox 2.0* will have an unreachable terminal mode
- *DreamZBox 2.0* shall have a startup mode consisting of *DreamZDash*
- *DreamZBox 2.0* shall have a Game Mode consisting of games like *Maze of Dreamz*
- Upon failure the Game Mode shall fall back to *DreamZDash*

## 2.6 Information Management

### *2.6.1 For* DreamZDash

- Shall store configurations and chosen settings in non-volatile memory
- Shall store user downloads in non-volatile memory
- Shall keep all memory stored on this system
- Should not allow other devices to access this data

### *2.6.2 For* Maze of Dreamz

- Shall generate data onto volatile memory
- May store player progress onto non-volatile memory which may include:
  - Entity State
  - Level Generation
  - Player Collected Items
- Should not allow other applications to access this data

### 2.7 Packaging

- *Maze of Dreamz* and *DreamZDash* shall be compiled and compressed into a singular file
- The file above may be loaded onto the console through the internet
- The file above may be loaded onto the console serially
- Other games may be compressed and provided over the internet

## 3 Verification and Metrics

### 3.1 Functional Requirements

ALL 5 buttons, the joy-stick, and the accelerometer on the controller and is used during the game play. The *DreamZBox 2.0* can use HDMI port to connect with other devices.

#### *3.1.1 Use Cases*

- Use Case 1: The user can launch *Maze of Dreamz* with the joy-stick of the controller.
- Use Case 2: The user can use each of the peripherals on the controller, causing something different in *Maze of Dreamz* or on the LED lights of the controller.

### 3.2 Usability Requirements

The user should feel satisfied with game settings.

#### *3.2.1 Use Cases*

- Use Case 1: A new user want to experience the *DreamZBox 2.0*. They can understand the control of the console, select and play the game without others' help.
- Use Case 2: The user start playing a game on *DreamZBox 2.0*. While playing the game, the user will feel satisfied with our default settings for buttons.

### 3.3 Performance Requirements

The *Maze of Dreamz* can continuously run on the *DreamZBox 2.0* for at least 6 hours without any battery charge.

#### *3.3.1 Use Cases*

- Use Case 1: The user can play *Maze of Dreamz* after a recent charge unplugged for 6 hours.
- Use Case 2: The user start playing a game on *DreamZBox 2.0*. While playing the game, the user will feel satisfied with our default settings for buttons.

### 3.4 System Interface

An interface will be provided to user to select games, change system settings and shut down the machine.

#### *3.4.1 Use Cases*

- Use Case 1: The user boots up the *DreamZBox 2.0*. The main interface will be shown to the user when the machine is fully boot up, then the user is allowed to select games, change system settings or shut down the machine in the main interface.

### 3.5 System Modes

Results of a tested environment using scripts will be provided to the client, detailing the restricted nature of the two modes. This will log the mode of operation before and after many different likely operations within code. Success will be shown if the modes only transition between modes is upon ”launching into a game”. The system should revert back to *DreamZDash* upon failures. The tests will also screen for erroneous behavior such as *DreamZDash* failing. The features will deem complete upon success of these tests.

### 3.6 Information Management

The *DreamZBox 2.0* will read the data from controllers and generate string type data files for user to store their system settings and game saves if necessary.

#### *3.6.1 Use Cases*

- Use Case 1: The user is playing *Maze of Dreamz*. They can choose to ”save and quit”, and a save file will be generated by *Maze of Dreamz* to record the state of the game

### 3.7 Packaging

The stakeholders will have access to the compressed digital file. Any computer may download the software on their own. The software may also be emulated, but such verification will not be supported. Instead the publicised existence of the file will represent the accomplishment of packaging the software.

## 4 Closing Statement

The *DreamZBox 2.0* alongside the *DreamZDash* and *Maze of Dreamz* will provide an entertaining experience to a diverse audience. The developers here at *CS29 Dreamz Catcherz* are really excited to get started. They share the vision of innovation the *DreamZBox 2.0* can carry ahead of other iterations. The innovating factor within *Maze of Dreamz* is its creation of adventure puzzles rather than simple geometric maps. *CS29 Dreamz Catcherz* feels this game will pack a punch for the *DreamZBox 2.0*. All in all, the *CS29 Dreamz Catcherz* are gracious for the opportunity to conquer this project.

## 5 Appendix

### Assumptions and Dependencies

The group assumes the following for the project:

- The *DreamZBox 2.0* is a portable piece of hardware which can compute images on to a screen
- The *DreamZBox 2.0* will be packaged with at least 2 controllers with a push-in joy-stick, 5 buttons, and an accelerometer.
- The team will have access to the hardware before release and the end of project.

# 6 GANTT CHART

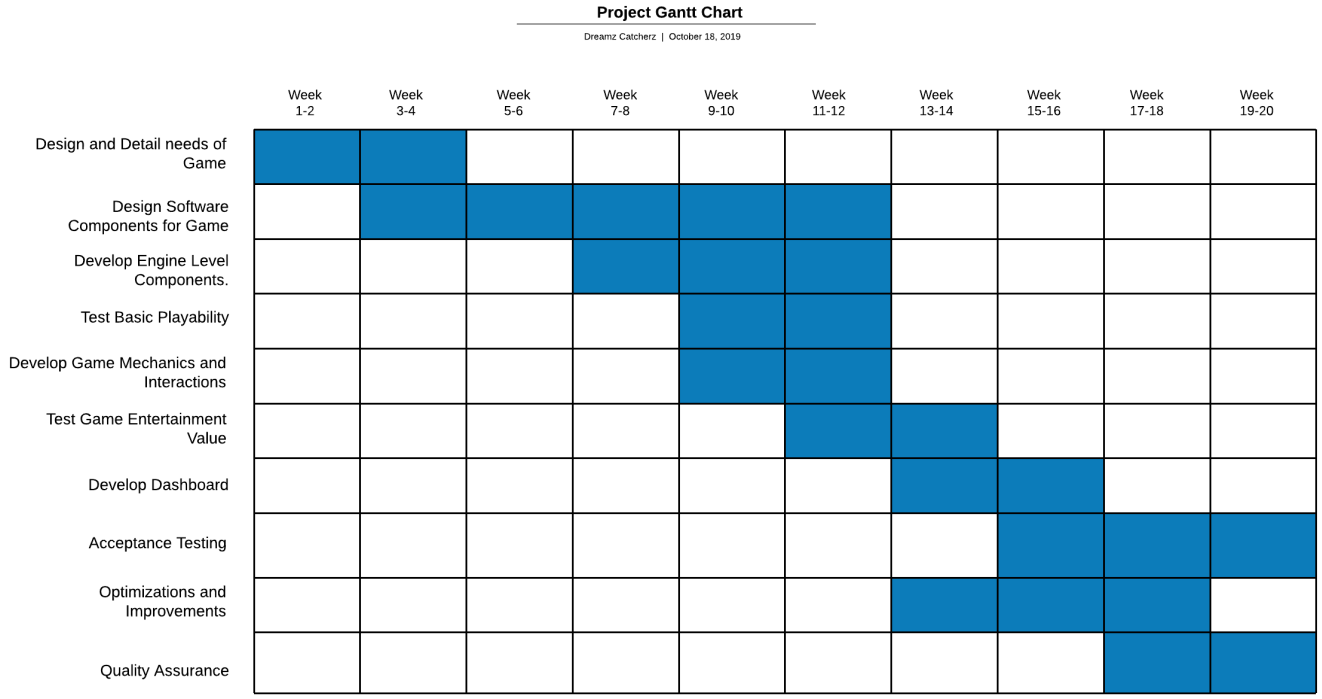


Fig. 1. Gantt chart detailing the goals and dates of the perspective project.