Software Requirements Specification

for

- DECK 15 -

Version 2.0 approved

Prepared by Andrew Sales, Katherine Bozin, Kaylene Petrin, Max Iniguez

Eastern Washington University

6/4/2020

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Andrew Sales	1/22/2020	Initial creation	1.0
Andrew Sales	1/26/2020	Added system features and introduction	1.0
Max Iniguez	1/27/2020	Proofreading and edits	1.0
Kaylene Petrin	2/2/2020	Minor Edits, Stim/Responses, Glossary	1.0
Andrew Sales	2/3/2020	Updated Table of Contents	1.0
Andrew Sales	6/4/2020	Updated SRS to version 2.0	2.0
Kaylene Petrin	06/04/2020	Added items to Appendix C	2.0

1. Introduction

1.1 Purpose

Deck 15 is a proof-of-concept game built in Unreal Engine 4 for flat screen PCs and plays as a first-person puzzle game. You play as the captain of a starship guided by an AI to repair ships systems when an unknown object impacts your vessel. A glowing blue power core is used to navigate the ship, power subsystems, and advance through levels.

1.2 Document Conventions

Standard coding conventions will be inherited.

1.3 Intended Audience and Reading Suggestions

This document is intended to be read by the Deck 15 development team, instructors overseeing senior project teams, other development teams looking to further understand the game, and future employers to understand the intent and development process behind the game.

1.4 Project Scope

Deck 15 is intended to be a playable 3D game for Windows operating systems to be played on a flat screen. Specific objectives include designing and creating four playable levels, a wind component to effect the flight of the ball, locations where gravity will vary, a throwable ball mechanic, as well as the ability to teleport the ball and summon the ball back to you. Wind, variable gravity, and a throwable ball physics will be prototyped and included in playable levels.

1.5 References

2. Overall Description

2.1 Product Perspective

As our client, Cyan Worlds requested the development of a flatscreen game for Windows operating systems with at least three playable levels, a ball that can be picked up and thrown, and the ability to deposit the ball into a goal to advance to the next level. It is intended to be a stand-alone game and not a followup or predecessor to any current game or software.

2.2 Product Features

Significant and major features of the game include the following:
- Four well-designed levels for the player to traverse
- Prototyped changed in gravity in a playable level
- Prototyped wind in a playable level
- Ball physics and throwing prototyped in a playable level
- Teleportation and summoning of the ball prototyped and usable in all levels

2.3 User Classes and Characteristics

Character:
The player will inhabit the character while playing Deck 15. Through the character, the player can interact with the ball and use it to solve puzzles throughout the game and progress through levels. The character can teleport to the ball or summon to the ball to the holding position in front of the character

Ball:
The ball is the primary puzzle solving device in Deck 15. It is needed to solve all puzzles and can be picked up, thrown, or dropped.

Switch:
Throwing the ball at a switch will change the color of the switch and activate various systems onboard the ship including elevators, gravity platforms, and wind.

Goal:
At or near the end of each level is a cylindrical goal where the ball must be deposited in order to move onto the next level. The character cannot teleport to or summon the ball while it is in the goal.

Wind:
In certain areas of the ship, heavy wind will be present and affect the flight characteristics and behavior of the ball. This is both a necessity to solve puzzles and hindrance to make puzzles more difficult to solve.

Gravity:
In certain areas of the ship, the ball will increase or decrease the gravity of the entire level.

2.4 Operating Environment

Deck 15 is intended to be played on Windows PCs using a flatscreen. Compatibility will include Windows 64-bit operating systems, including Windows 10, Windows 8, and Windows 7.

2.5 Design and Implementation Constraints

It is recommended that you run Deck 15 on a Windows 64-bit PC with at least 8gb of RAM with DirectX 10 or later installed.

2.6 User Documentation

Deck 15 Documentation Sheet

2.7 Assumptions and Dependencies

Assumed that the user has installed Windows 10, 8, or 7 and is running DirectX 10.

3. System Features

3.1 Four Playable Levels

3.1.1 Description and Priority

The game will include at four playable and well-designed gray-boxed levels. This is a medium priority feature. Other features must be completed first before each level can be implemented to specification.

3.1.2 Stimulus/Response Sequences

An individual who decides to play the game will have the opportunity to progress through four levels, each with different mechanics, styling, and layout. Each level will be progressively more challenging.

3.1.3 Functional Requirements

REQ-1: Throwable ball mechanic as well as functional teleportation and summoning of ball.

3.2 Prototyped Wind Mechanic

3.2.1 Description and Priority

Wind will act as a force against the ball moving it in a direction the player likely will not anticipate or accelerating the ball in a desired direction with correct usage. This is a high priority feature as it will dramatically affect the quality and difficulty of gameplay.

3.2.2 Stimulus/Response Sequences

Stimulus: Ball comes enter a wind trigger box

Response: Ball trajectory is influenced by wind and sent of course from its original trajectory

3.2.3 Functional Requirements

REQ-1: Functional ball actor is required so that the wind can interact with it.

3.3 Prototyped Variable Gravity Mechanic

3.3.1 Description and Priority

Variable gravity will allow a ball to act lighter or heavier depending on its location in the level. The gravity of the entire level is altered when the ball enters a gravity trigger box. Gravity is then restored to its original level when the ball leaves the trigger box. This is a high priority feature as it will dramatically affect the quality and difficulty of gameplay.

3.3.2 Stimulus/Response Sequences

Stimulus: Ball comes begins overlapping the gravity trigger box

Response: Gravity for the entire level is increased or decreased while the ball is inside of the trigger box

3.3.3 Functional Requirements

REQ-1: Functional ball actor is required so that the gravity can interact with it.

3.4 Prototyped Ball Throw and Physics

3.4.1 Description and Priority

The ball can be thrown by the user in the direction of their choosing. Upon clicking the left mouse button while holding the ball, it will be thrown and interact with the set environment physics and behave accordingly. Additionally, the ball can be dropped instead of thrown if the user wishes. Picking up and dropping the ball can be done when clicking the right mouse button. This is high priority as it is a core mechanic of gameplay.

3.4.2 Stimulus/Response Sequences

Stimulus: User presses the right mouse button to pick up the ball, then presses the right mouse button again to drop the ball or the left mouse button to throw the ball.

Response: The ball is propelled in the desired direction.

3.4.3 Functional Requirements

REQ-1: A character must be present in order to house the pickup and throw component.

3.5 Prototyped Summoning of the Ball

3.5.1 Description and Priority

The ball can be summoned to the player into the holding position. This is a high priority as it is key to solving puzzles and navitating levels.

3.5.2 Stimulus/Response Sequences

Stimulus: The user presses the G key.

Response: As long as the ball is not presently located inside of a trigger box that does not allow the ball to be summoned, the ball will instantly appear in front of the player in the holding position.

3.5.3 Functional Requirements

REQ-1: Character must be present to house the pickup and throw component for the ball.

REQ-2: Ball must be present in order to summon it to the character.

3.6 Prototyped Teleportation to the Ball

3.6.1 Description and Priority

The player can teleport directly to the ball's location. This is a high priority as it is key to solving puzzles and navitating levels.

3.6.2 Stimulus/Response Sequences

Stimulus: The user presses the T key to teleport directly to the ball's location.

Response: As long as the ball has already been summoned once (as to not teleport to locations the player should not go), the player will instantly be relocated to the location of the ball.

3.6.3 Functional Requirements

REQ-1: Character must be present to house the pickup and throw component for the ball.

REQ-2: Ball must be present in order for the character to be able to teleport to it.

4. External Interface Requirements

4.1 User Interfaces

The user or player will use a keyboard and mouse combination to interface with the game. Through mouse movements, clicks, and key strokes a player can select a direction (vertical and horizontal angle) to propel the ball in order to reach the goal. Additionally, the same interfaces will be used to navigate menus to set up and play the game.

Player controls:

-A: strafe left

-S: move backward

-W: move forward
-D: strafe left
-shift: double player's moving speed (run)
-G: summon ball to the holding position in front of the player
-T: teleport directly to the ball's location
-space bar: jump
-left mouse: throw ball when holding
-right mouse: pick up ball or drop ball when holding
-esc: in-game pause menu

4.2 Hardware Interfaces

Mouse:
A mouse is required in order to operate the in-game menus, alter the direction the player is facing, and pick up and throw the ball.

Keyboard:
A keyboard is required for in-game character movement as well as teleportation to and summoning of the ball.

4.3 Software Interfaces

Appropriate and require drivers must be present in order to run the required mouse and keyboard as well as Deck 15.

4.4 Communications Interfaces

N/A

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The following are the minimum requirements recommended by the Unreal Engine team:
- Desktop PC
- Windows 7 64-bit later
- Quad-core Intel or AMD processor, 2.5 GHz or faster
- NVIDIA GeForce 470 GTX or AMD Radeon 6870 HD series card or higher
- 8 GB RAM

5.2 Safety Requirements

N/A

5.3 Security Requirements

N/A

5.4 Software Quality Attributes

N/A

6. Other Requirements

N/A

Appendix A: Glossary

ball - An object that the user can interact with and move. It's purpose is for the user to throw and use to solve puzzles unroute to the goal in each level.

goal - A container in each level where the ball must be thrown in order to power ship systems and move on to the next level.

gravity variability - An area in which the gravity is altered for an entire level. The player and/or ball may act heavier or lighter.

user - The individual currently playing the game with the ability to pick up, throw, and drop the ball.

wind - An obstacle the user may face at any given level. Replicates an airstream and has the ability to redirect the ball if it comes in contact.

Appendix B: Analysis Models

N/A

Appendix C: Issues List

1. When the ball is in the holding position and is pressed against some surfaces in various levels, the player can teleport through walls when pressing the T key and gain access to undesirable locations.
2. The character can get stuck in some objects when teleporting to the ball.
3. The ball will become "deformed" by having an egg-shaped or rigid appearance when picked up or summoned. Though this is rare, it can happen.
4. Dialogue can be cut off when another dialogue trigger box is overlapped.

5. Pausing the game only pauses the player, not the background actors or sounds.
6. Player still has the ability to summon and teleport the ball during loading times. This action cannot be seen, but can be heard.