Alexander Tripp

CIS 587 – Dr. Maxim

November 26, 2014

Rollo

Design Document

Table of Contents

Overview....2

Appearance....2

Gameplay....3

Story Abstract....3

Game Mechanics....3

Gameplay....3

Level Progression....3

Mechanics....4

Physics and Movement....4

Items....4

Static Obstacles....5

Active Obstacles....5

Actions....6

Artificial Intelligence....6

User Interface Description....6

Level Design....7

Technical....9

Game Engine & Language....9

Target Platform....9

Assets....9

Artwork....9

Music and Sound....10

Overview

Rollo is a puzzle game in which the player, represented by a sphere, must navigate through various 3D levels, collecting items that are required to get to the next stage. It will resemble a platformer game with restricted movements.

Appearance

The game environment will have a whimsical look and feel, characterized by bright colors and simple, blocky shapes with vibrant textures. Many objects in Rollo will suggest children's toys such as building blocks, and the player object itself will be a bright, metallic sphere.

Each level will take place in a unique setting, such as grassy forests, icy mountains, tropical islands, and . The lighting, colors, and textures will set the mood for each environment.

Gameplay

The player will navigate through puzzle levels by controlling the motion of a rolling ball. In order to reach the next level, they must collect a set of five medallions. In some cases, collecting an item will require taking an irreversible action, such as falling to a lower level, so the player must carefully plan the order in which they will collect these medallions.

Aside from the medallions, there will be coins throughout the level for the player to collect. These are not required to advance, but will boost the player's score.

The player also has some control over various obstacles in the game. For example, rolling onto a switch may open a gate or collapse a bridge, allowing the player to reach a previously inaccessible location.

Story Abstract

As a puzzle game, there is no "storyline" in the sense of a plot with characters. However, the progressive nature of the game will have some elements of a story; for instance, the player will advance through levels in a linear manner. Within levels, items must often be collected in a particular order (there are one-way paths).

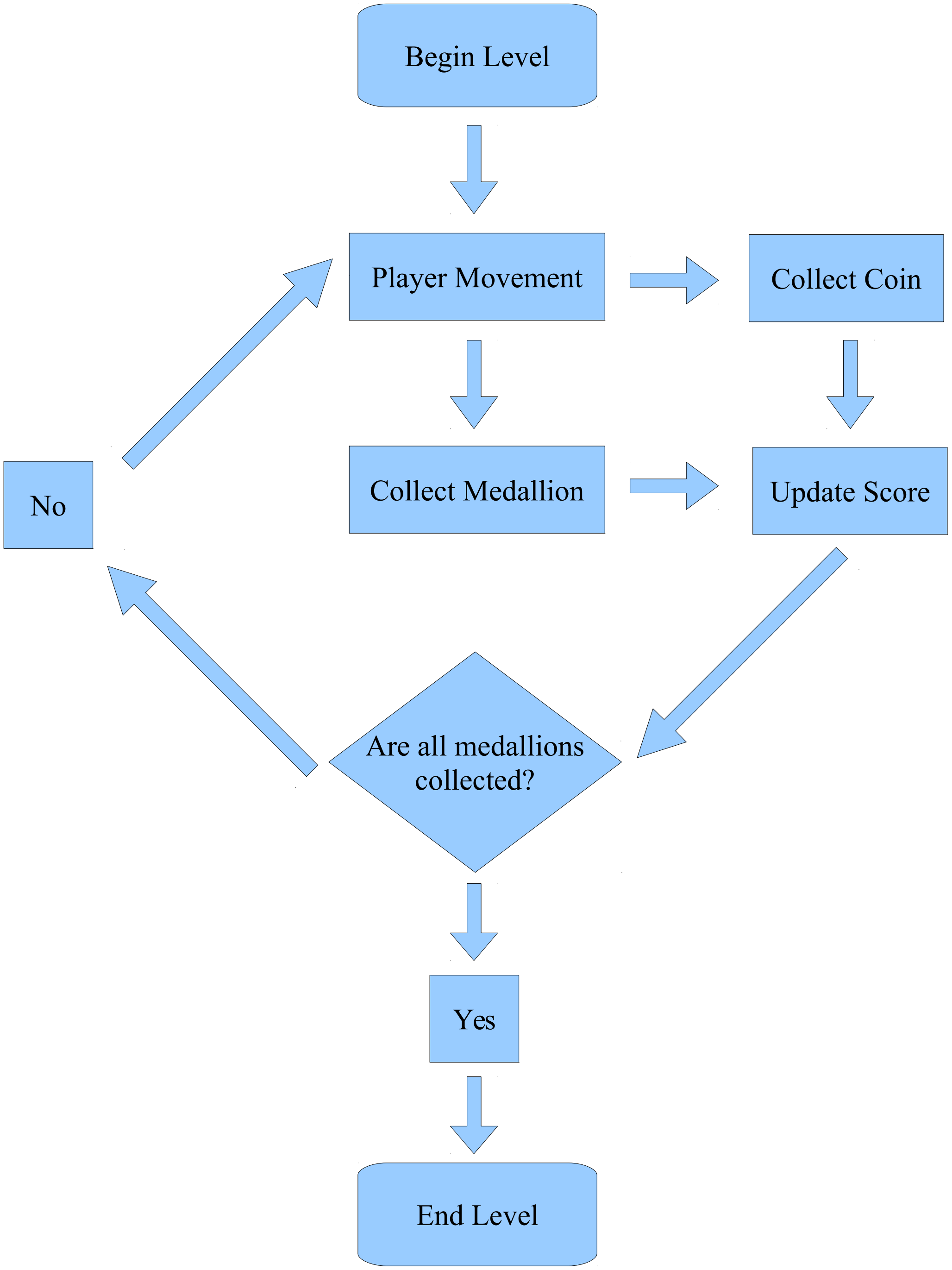
Game Mechanics

Gameplay

The player has two main challenges to overcome in each level: the strategic challenge of planning a route to collect all five medallions, and the control challenge of navigating that route. Because the ball is controlled by applying forces based on key presses, it can get out of control and roll off the path; sometimes off the scene entirely. In such a case, the level must be restarted.

The levels will become more difficult as the game progresses. Also, each level will take place in its own environment and have unique features not found in other levels (for example, a water level could have floating platforms and a switch to raise the water level.

Level Progression



Mechanics

Physics and Movement

Rollo will use realistic physics, through the Unity3D physics engine. The player's actions are implemented through forces applied to the ball, causing it to roll. It can collide with obstacles and fall off of cliffs, so the player must keep it under control to avoid having to restart a level.

Items

There are two collectible items in Rollo: Medallions and Coins.

Medallions are required to access the next level. Each level contains five of them, and all five must be collected. In theory they can be collected in any order, but the topology of the level may enforce a certain order. For example, one medallion could be located at the bottom of a hole that has no way out. This one must be collected last, because after reaching it there is no way to reach the others.

Coins are not required, but collecting them will boost the player's score. They may be used to provide a hint to the player – often, collecting a medallion will involve passing through a chain of coins – or they may be located in hard-to-reach places, providing skilled players with an extra challenge.

Items are collected simply by passing through them. They do not provide any physics collisions and are immediately collected upon contact with the player – similar to coins or powerups in a classic platformer.

Static Obstacles

Most obstacles in Rollo will be static. They do not move or interact with the player in any way beyond providing physics collisions. While their appearance will depend on the environment of each level, they are functionally identical throughout the game.

Examples of static obstacles include:

- Blocks
- Ramps
- Tracks along which the player will roll

Active Obstacles

Although they are less common that static obstacles, active obstacles provide the most interesting challenges. Some will appear in nearly all levels (like switches), while others may only appear once. They will provide most of the strategic variation between levels, and give each level its unique character.

Examples of active obstacles include:

- **Single-use bridges:** These bridges collapse immediately after the player crosses them
- **Switches:** Activate other objects, but have no behavior of their own.
- **Strikers:** Solid moving objects that can knock the player off the path, or lift them to a new area.
- **Gates:** These must be opened with switches, which may be located elsewhere in the level. After it is opened, the gate may remain open indefinitely, close after the player passes through, or close after a timer expires.
- **Moving platforms:** Behave just like static obstacles, but they move.
- **Water:** The player moves more slowly through water. Also, some active obstacles will float on the surface of the water. The water level may change in a level: in some cases, it must be activated by a switch, while in others it simply follows a schedule. In the former case, the water change may be permanent, it may return to normal after a timer expires, or it may require another switch to reset.
- **Floating platforms:** These platforms never drop below water level (but may sit on top of other objects when the water is below them). They may or may not have their own motion on top of that.
- **Gravity:** Normally, gravity only pulls down. However, switches may change the direction of gravity and rotate the camera to match.
- **Teleporters:** Active obstacles that rapidly move the player to another location. They may be switches that actually "teleport" the player instantly, or they may be objects such as pipes, tracks, or cannons that temporarily suspend user control while moving the player to the new area.

Actions

All actions that the player can take apart from moving and collecting items must be done by colliding with various active obstacles. Most of the time this means rolling onto a switch, but some active obstacles can be triggered by direct collision with themselves (for example, a single-use bridge).

Artificial Intelligence

The AI in Rollo will consist of the behavior of active objects. In most cases, this falls into one of the categories below:

- **Object Movement:** Some active obstacles simply move along a predefined path. This will usually be circular, sinusoidal, or sawtooth motion.
- **Timers:** Many obstacles will have timers attached to them. For example, a gate opened by a switch may only remain open for a short time, requiring the player to hurry through before it closes. It will always be obvious to the player that a timer is active: a ticking sound effect will accompany an active timer.

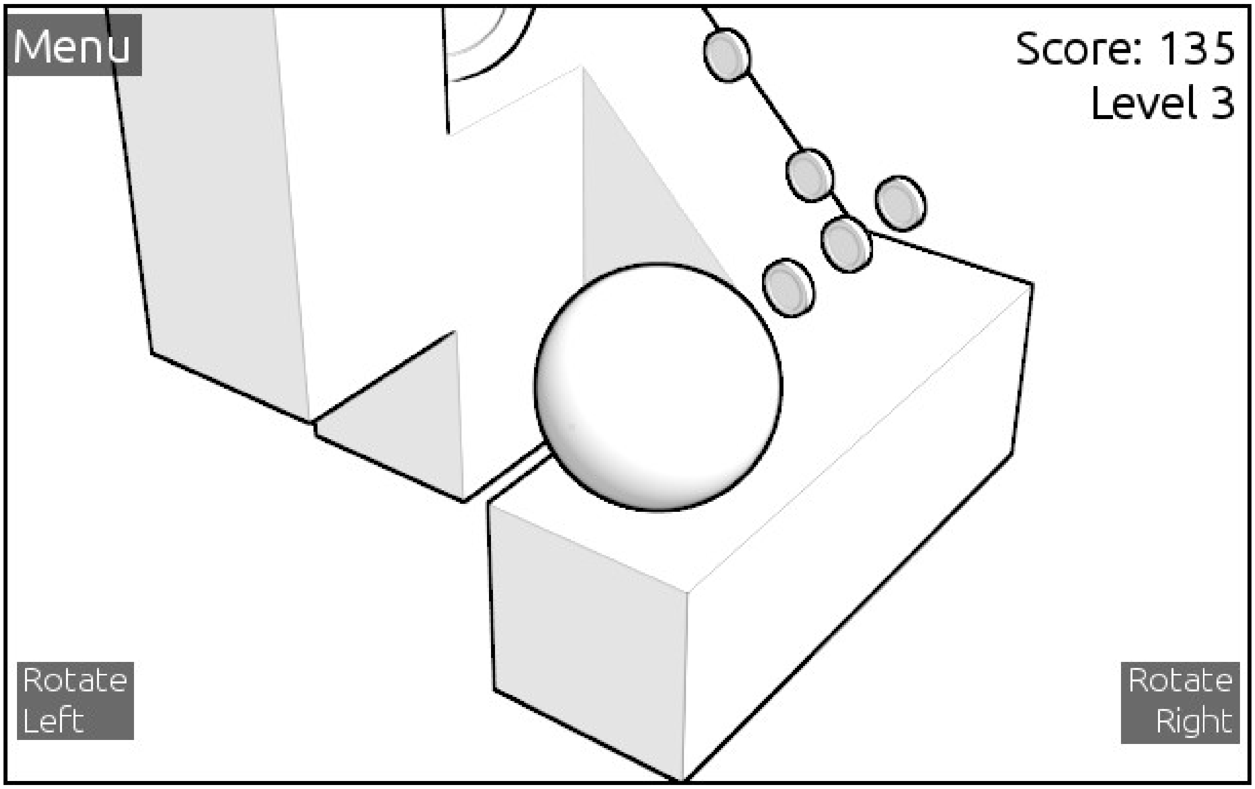
- **Single-Trigger Actions:** An active obstacle's behavior may have a single trigger, such as the player crossing a collapsing bridge. As soon as the object is activated, it will complete the action.
- **Scripted Events:** For a few unique obstacles, unique scripting may be required. For example, floating platforms must continually monitor the water level and adjust their height accordingly.

User Interface Description

Within levels, the UI will be minimalistic, as there are very few controls. The player will be able to control motion and rotate the camera, using WASD and arrow keys. Apart from that, any interaction with the world will happen through physical triggers (such as rolling onto a button that opens a gate).

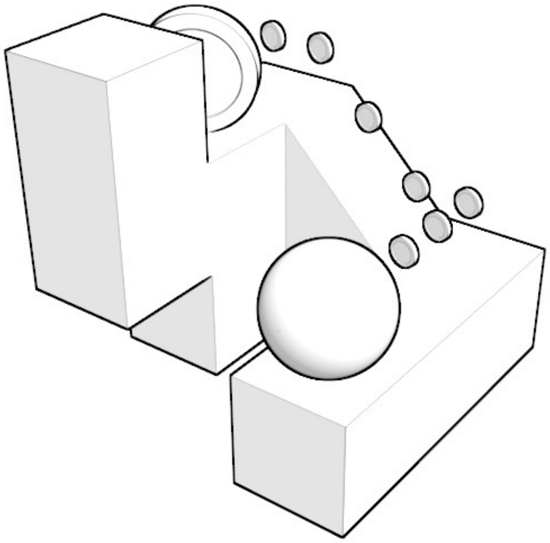
The UI will also display information such as the current score and level number, and provide options to access the level menu, restart the level, quit to main menu, and exit the game.

The main menu, displayed when Rollo starts, will contain options to start a new game, view instructions, view high scores, and exit the program.



Level Design

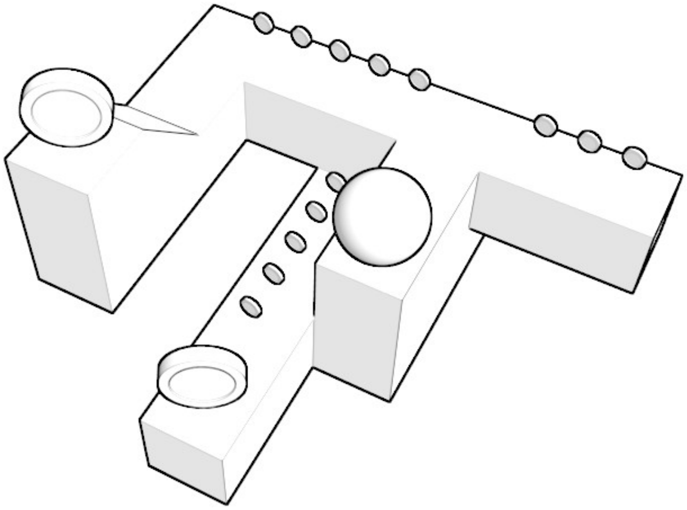
Below is an example of a very simple level with only one medallion to collect:



In this level, a chain of coins leads the player up a ramp to the medallion. While this level is extremely simplified, it illustrates some of the key features common to all levels:

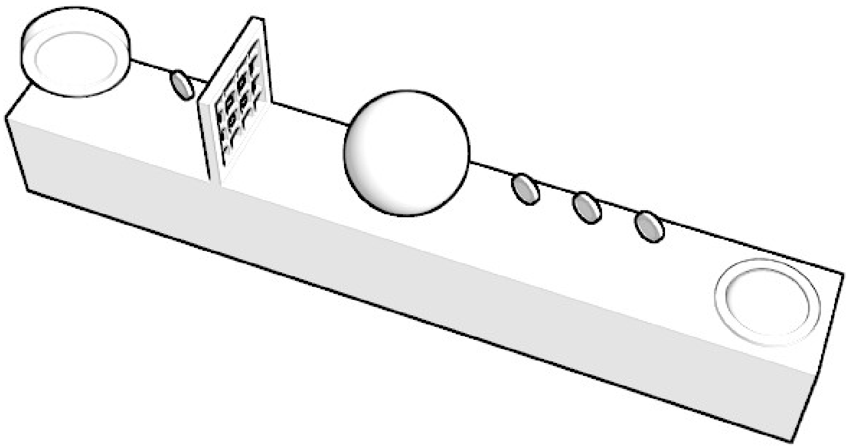
- The player object
- Coins and medallions to collect
- Blocks and ramps, forming the level terrain
- Edges to fall off of, requiring a restart

A slightly more complex level is shown next:



This time, there are two medallions. The player must collect the one on the upper level first before dropping down to collect the other. If the lower medallion is collected first, there is no way to access the upper one and the level must be restarted. Also, note that this level contains an optional area containing only coins.

The final example features an active obstacle:



In this example, the player's path to the medallion is blocked by a gate. The player must activate the switch on the ground at the right end of the level, which will open the gate and give access to the medallion.

Technical

Game Engine & Language

Rollo will be built using the Unity3D game engine, and will use Unity's physics engine for physics simulation. Coding will be done in the C# programming language.

Target Platform

The game will be released to run natively on Linux, Mac, and Windows computers.

Assets

Artwork

Models for Rollo (both visible objects and their physics colliders) will be developed using the Blender 3D modeling system. Textures will be created in the GIMP.

The game will require art assets for the following:

- Player object

- Medallions
- Coins
- Static obstacles (blocks, ramps, etc.)
- Active obstacles (switches, gates, bridges, etc.)
- Skybox
- Background scenery (trees, water, rocks, terrain, etc.)
- Lighting

Music and Sound

Rollo will feature sound effects for various events. While much of this is purely aesthetic, it will provide useful information to the player, such as warning them that a timer is running out.

- Player motion
- Collecting items
- Activating items
- Ticking to warn of active timers
- Collisions with physics objects

Additionally, the game will feature background music while the player is in a level, or at the main menu.