

Game Design Document

Game Name: Cursor.zip

Mach 26

Contents

Initial Information.....	3
MoSCoW	4
Design Pillars	6
Core Gameplay Loop.....	7
Controls	8
Mechanics	9
Level Design	22
User Interface	24

Initial Information

Game Name: Cursor.zip

Genre: Speed Platformer

Players: Single player

Gameplay summary:

The player will select a level and then run through it, collecting score and trying to complete it within a time threshold to get a high rank once completing it. The levels will feature hazards and obstacles to avoid as well as objects which benefit the player. Upon completing a level, the players rank will be calculated and displayed to them and be given the option to play again or return to the level selection screen.

MoSCoW

Must Have

At least one level

The game must have one fully functional, tested level with polished assets and obstacles.

Core Game loop

The game must load the players into the level, and track score throughout to give a rank when finished.

Movement mechanics

The character must have polished working basic mechanics which have plenty of potential for momentum and fun movement.

Functional UI

The game must boot into a main menu and there must be ways to pause the game mid gameplay.

Should Have

At least two playable levels

The game should have two fully functional, tested levels with polished assets and obstacles.

Ball form

The character should be able to change into the ball form to provide more depth to the movement system.

Score system

The game should have a score system which tracks actions which the player does so that they can rack up points to improve their final score at the end.

Set dressed environments

The game should have completely decorated environments which fit the theming of the game.

Music and SFX

The game should have sounds for every action and event in the game. There should be background music to the menus and levels.

VFX and Juice

The game should be fully juiced and have visual effects for actions and events.

Could Have

Three or more levels

The game could have three fully functional, tested levels with polished assets and obstacles.

Level Builder

The game could have a level builder function which players would be able to use to make their own levels.

Other character(s)

The game could have other characters than the main character such as a rival or a character to save.

Won't Have

Multiplayer

The game won't have multiplayer or online gameplay.

Slow paced levels

The level design won't be made in a way where the player needs to backtrack or stick around to complete puzzles.

Open world

The game won't have open worlds to explore and move around in.

Design Pillars

Movement

Movement is the most important pillar of this game. If movement is not fun, the gameplay loop will be significantly worse

Form Change - The character has a basic running mode but can also transform their torso into a ball which has very low friction and gains lots of speed performing actions going down slopes. The speed will be carried over if you change back into the walking form but if they are on the ground, the friction will slow the player down fast.

Balance Between Forms - Both forms should be necessary to complete a level. There should be equal amounts of practicality with both forms.

Silly Cartoon Style

The character and environment should be able to squash, stretch and do other wacky movements without it looking out of place. They will have bouncy animations and minorly stretchy limbs which look natural on the character.

Low Skill-Floor and Optional High Skill Ceiling

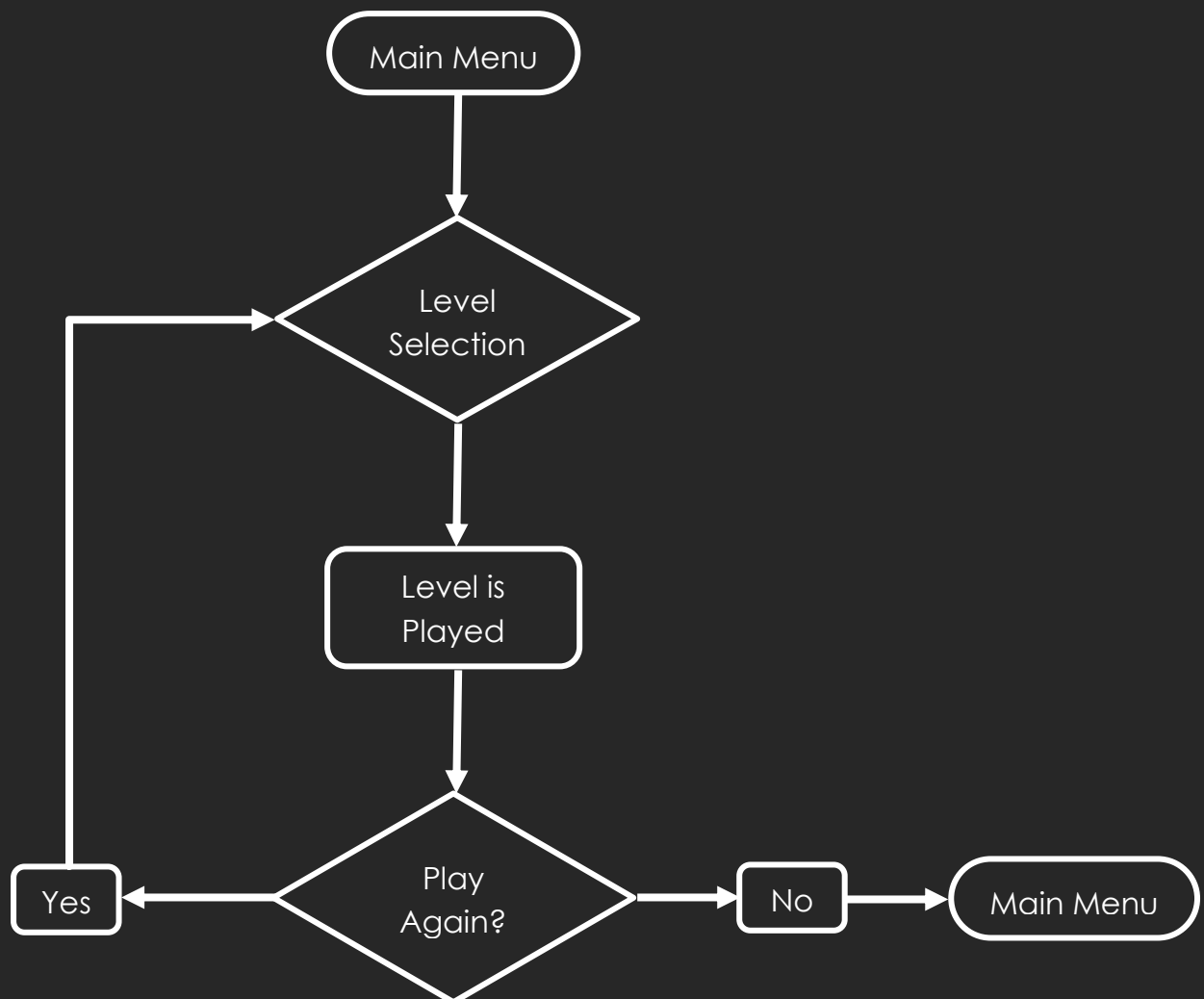
The game should be easily accessible to players who are new to the genre but pose some challenge so that it is still engaging. The game should give the player the option to complete more difficult than usual challenges, but these will not be required to progress through the game.

End of Level Rating System - At the end of a level, the player will be given a rank based on their performance this will be increased by completing the level quickly, defeating lots of enemies, picking up collectibles, and doing midair tricks.

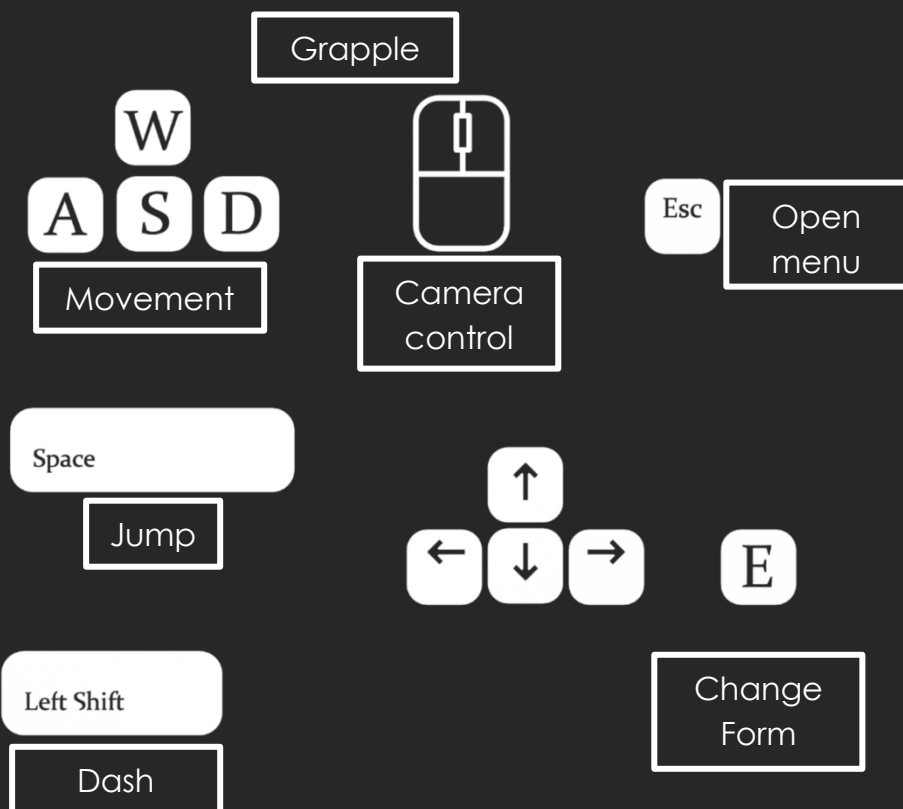
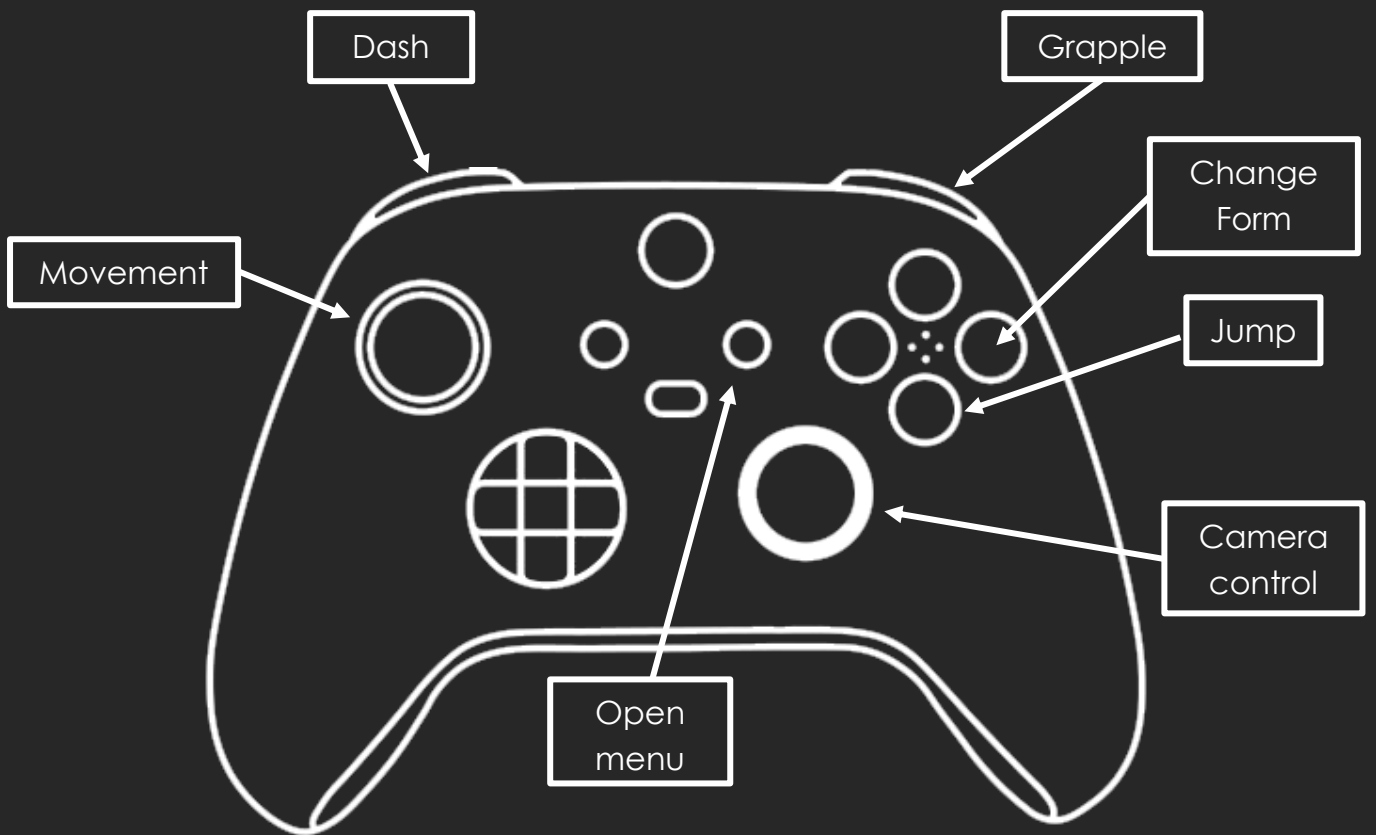
Shortcuts - There should be small deviations in the route of the level which will cut the time needed to complete the level which results in more potential points. These routes may be more difficult than the main pathway. There will be hidden collectibles on these routes which give points.

Checkpoints - The levels will have checkpoints which will respawn the player at these points after a half second delay. Time will not be rewound when respawning. The player will be able to turn checkpoints off in the settings before starting a level. If a level is completed deathless, more points will be rewarded.

Core Gameplay Loop



Controls

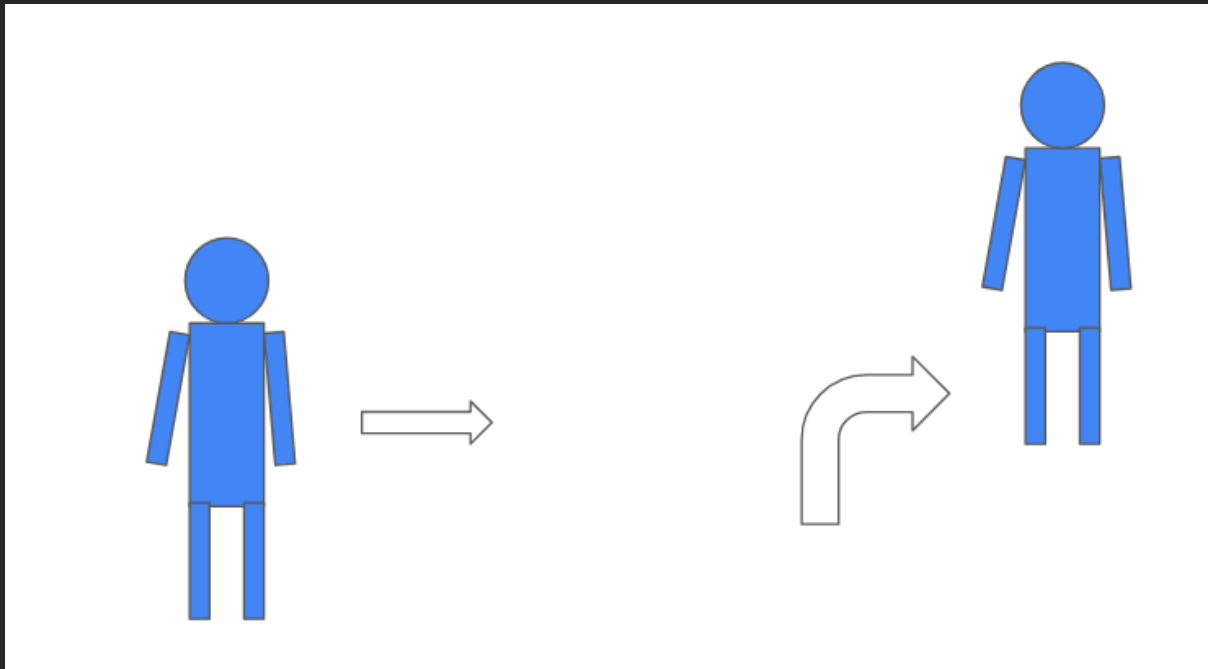


Mechanics

Player mechanics

Running Form

When in the base (running) form, the player will be able to run around and jump as well as grapple. They will have limited friction and if running along the ground while moving with lots of speed, they will gradually slow down back to their top running speed. When at top running speed or above, the player will have a slightly different running animation and will be able to wall run.

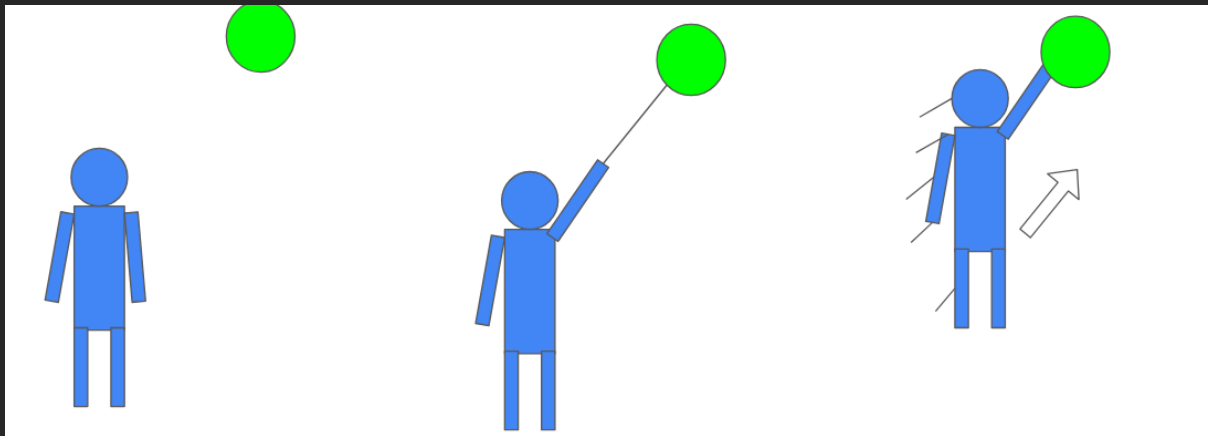


Grapple - Pull and Swing:

The player will have a grapple hook ability which can be fired at a grapple point. If the player is within range and can see a grapple point. There will be two types of grappling in total, and the one used will be dependent on the player's current form. It will aim automatically.

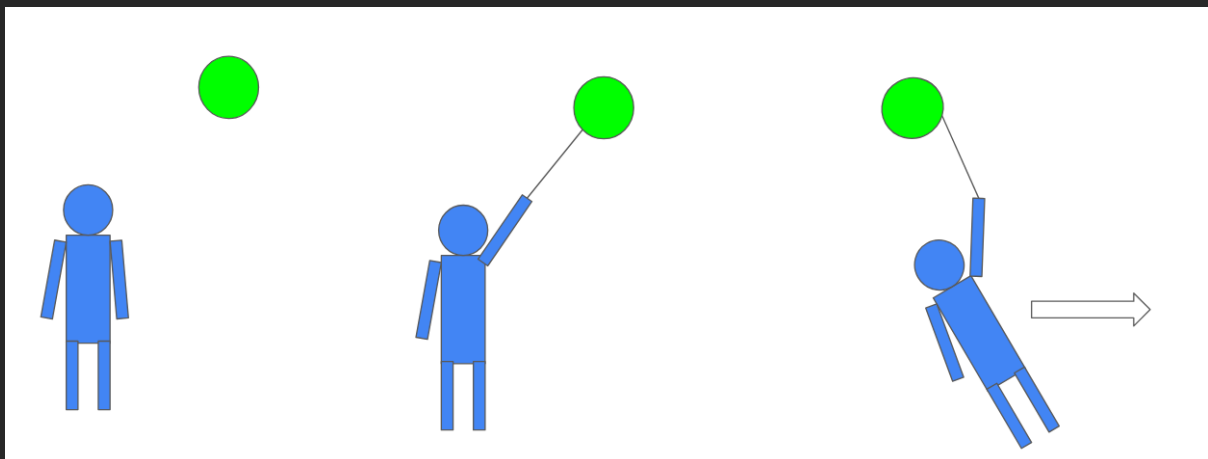
Pull

In the running mode, the grapple will pull the player towards the designated point in a straight line. The grapple point has no collision so the player will travel through it gaining small amounts of speed. The grapple hook will disconnect from the point once the player passes through it.



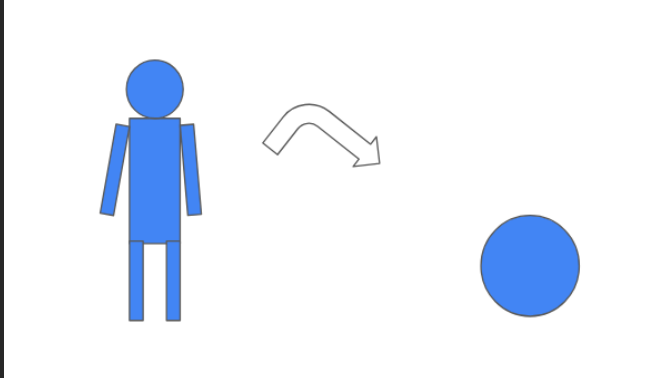
Swing

The swing grapple will be used instead of the pull when in ball form, this will be a loose physics-based grapple in which the player can swing around. This grapple will be active for as long as the player has the button held. They can gain lots of momentum by timing their swing correctly.



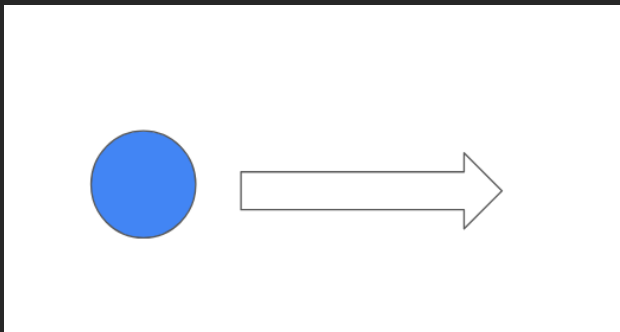
Switching Forms

By pressing a button, the player will be able to change between the running and rolling forms. This transformation will be instant, and the player will not slow down by changing forms.



Rolling Form

When rolling, the player will have low friction which will make it hard to gain speed but easier to conserve speed. In the rolling form, the player can move and jump. The player can use this form to conserve momentum. The player can use the ball form to roll down hills to gain lots of speed and up ramps to make large jumps. While in the rolling form the player cannot wall run, nor dash.

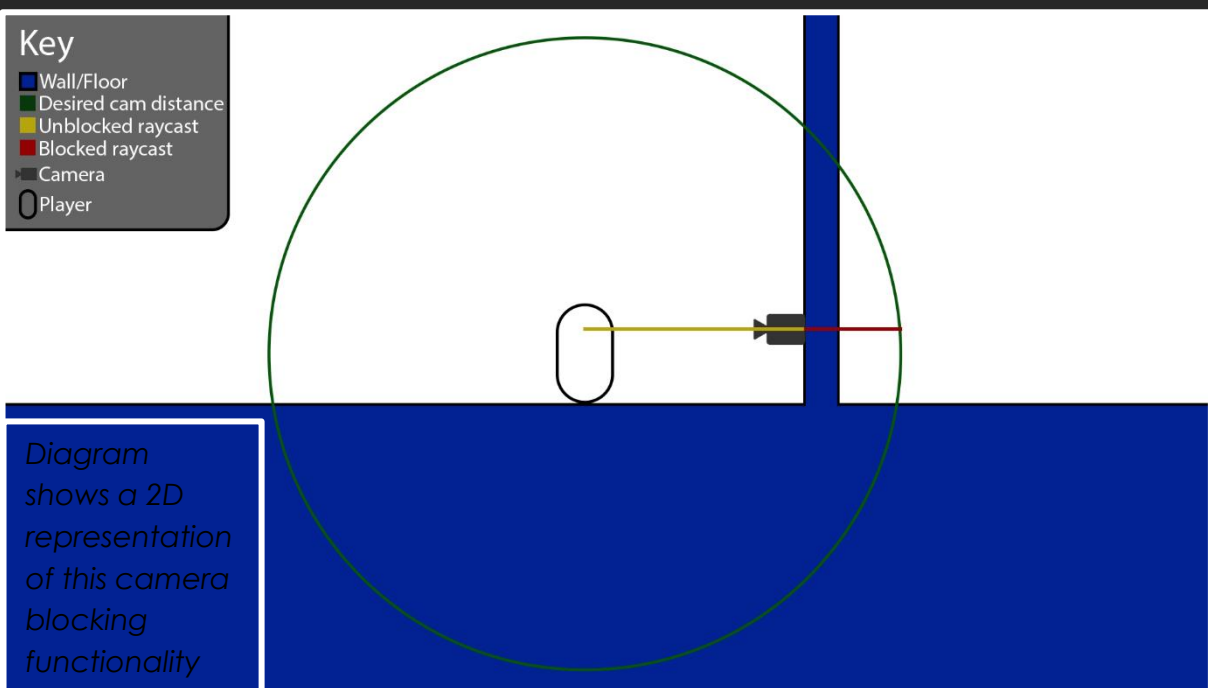


Camera Functionality

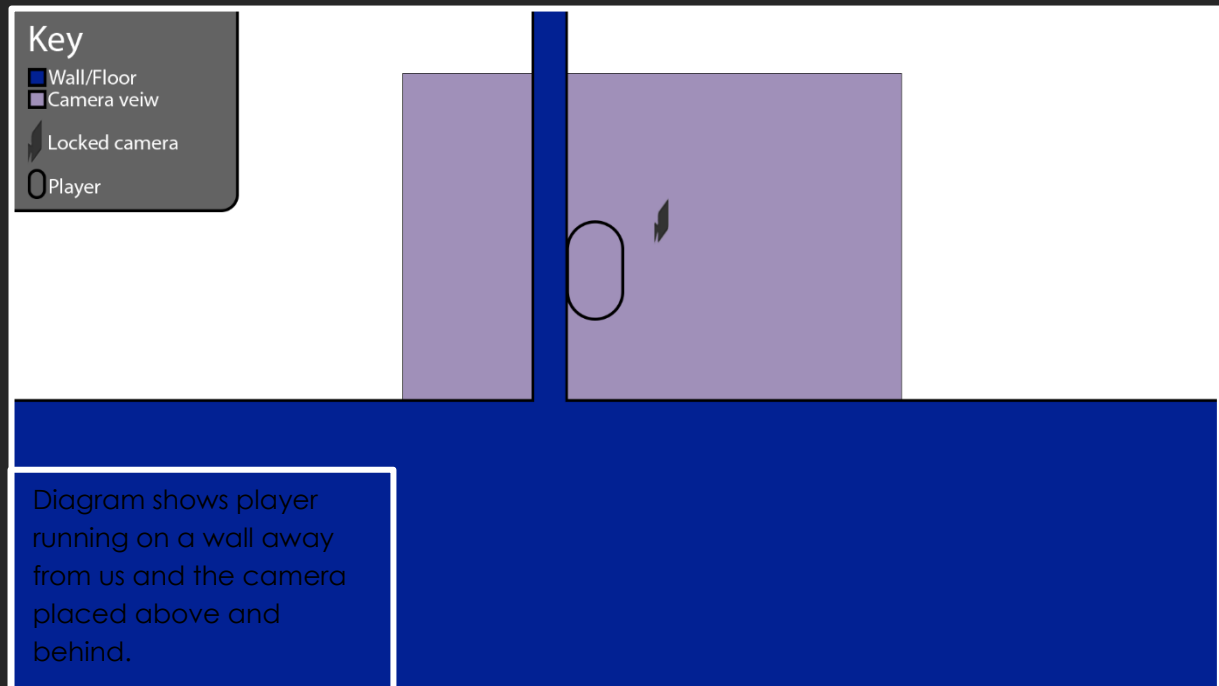
The camera will move with full accuracy to input and have no delay to allow players to react instantly and be informed of their environment. In normal form the camera will sit directly above and behind the player. When looking up the camera may clip through walls so we will push it forwards by ray casting to its desired position from the player and if a surface is hit then the camera will sit on the first hit point.

When moving faster the FOV will be increased to emphasize speed and the camera can lag on extreme speed boosts to give a feel of the camera being left behind.

One issue to consider is if a player runs behind small thin obstacles, then the camera moving back and forth suddenly may be disorienting, to minimize this we will use a layer mask to identify walls and floors and only allow the ray cast to interact with these elements of the environment while ignoring small obstacles. (Shaders making these small obstacles clear near the camera would be good but we have no shader experts making this a push goal)



When wall running the previous system may result in uncomfortable camera positioning and so we will have a set position for the camera when wall running. When wall running the camera will be locked on the opposite side to the wall pointing straight or slightly away from the wall to inform the player of what is ahead and of what is out from the wall if they jump off as anything left of the wall is unimportant to the player.



Considering our third person camera system we should look for situations where we can unlock the camera from the player and put it on tracks, in certain areas this would help to emphasize the scale of environmental elements and player speed. This will mostly depend on level design and so may not be included if level design feels it is unnecessary. This will be a stretch goal after all movement systems are finalised as if the camera is unlocked or moved separate from the player some movement systems may break.

Score System

At the end of each level the player will be assigned a rank based on the players score, the completion time and if any checkpoints were used. The player will gain score points by doing tricks. The ranks will be through A to D. If the player obtains an A rank without dying in that level, S rank is to be awarded. A, B and C are meant for more conventional performances, and a D rank is the bare minimum completion.

Each level will have its own threshold for the score and time required for each rank. Typically, S rank will require a very fast completion with a high score and no deaths, while ranks A-C will allow the player to make some errors, while D rank will have no requirements and will be awarded for any completion of the level, no matter how long it took or how low the score. Once the level has been completed for the first time the player will be informed of the requirements for each rank for that level.

Score Points:

The player will gain score points by performing a variety of actions

All actions that grant score points:

Using a grapple

Using a dash

Wall running (gain more points the longer you are on the wall)

Performing any of these actions immediately before or after changing between forms

Performing any of these actions in quick succession (bonus points if they are different actions)

Moving fast (gain more points longer you are at this speed)

Spending time mid-air without touching the ground

Each levels score threshold will take how many of these actions the player will be able to complete within the levels time restriction into account.

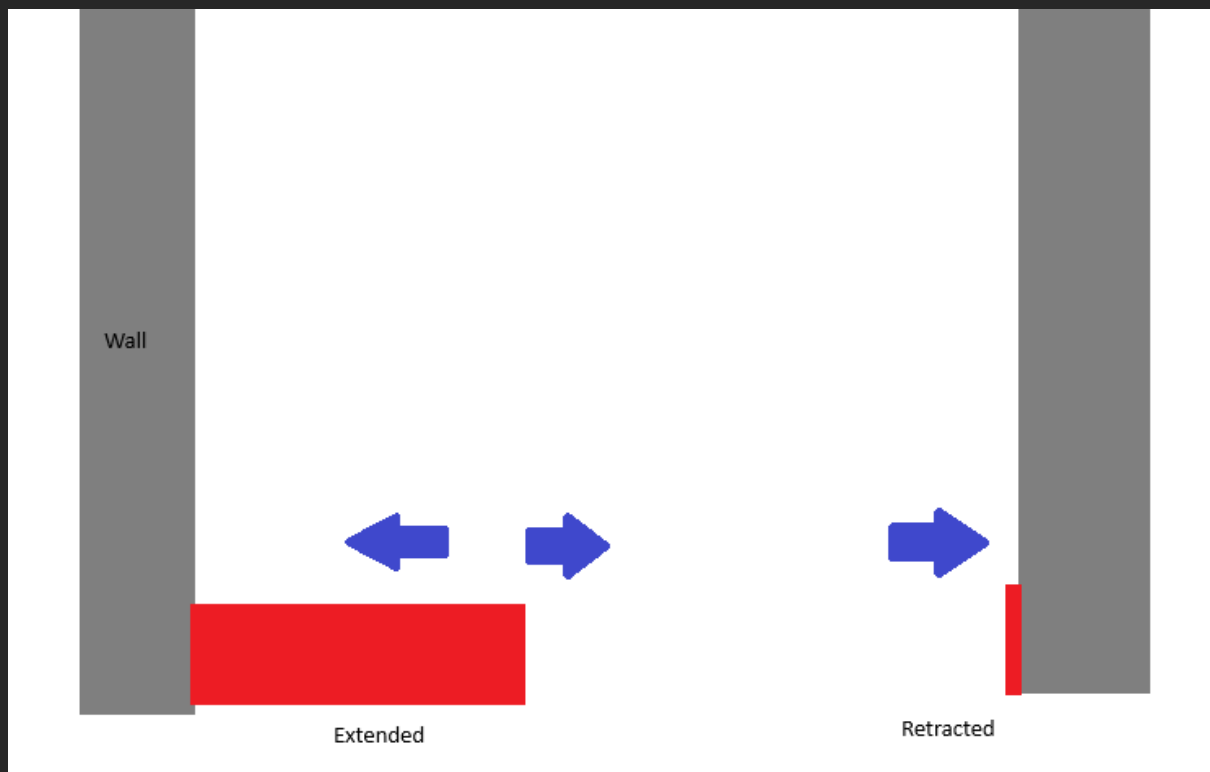
Level Mechanics

Checkpoint System:

There will be numerous checkpoints spread out throughout each level, the player walking into one will replace the player's respawn point at that location, to be used upon death, replacing the previous checkpoint. Using a checkpoint will lock the player out of S rank.

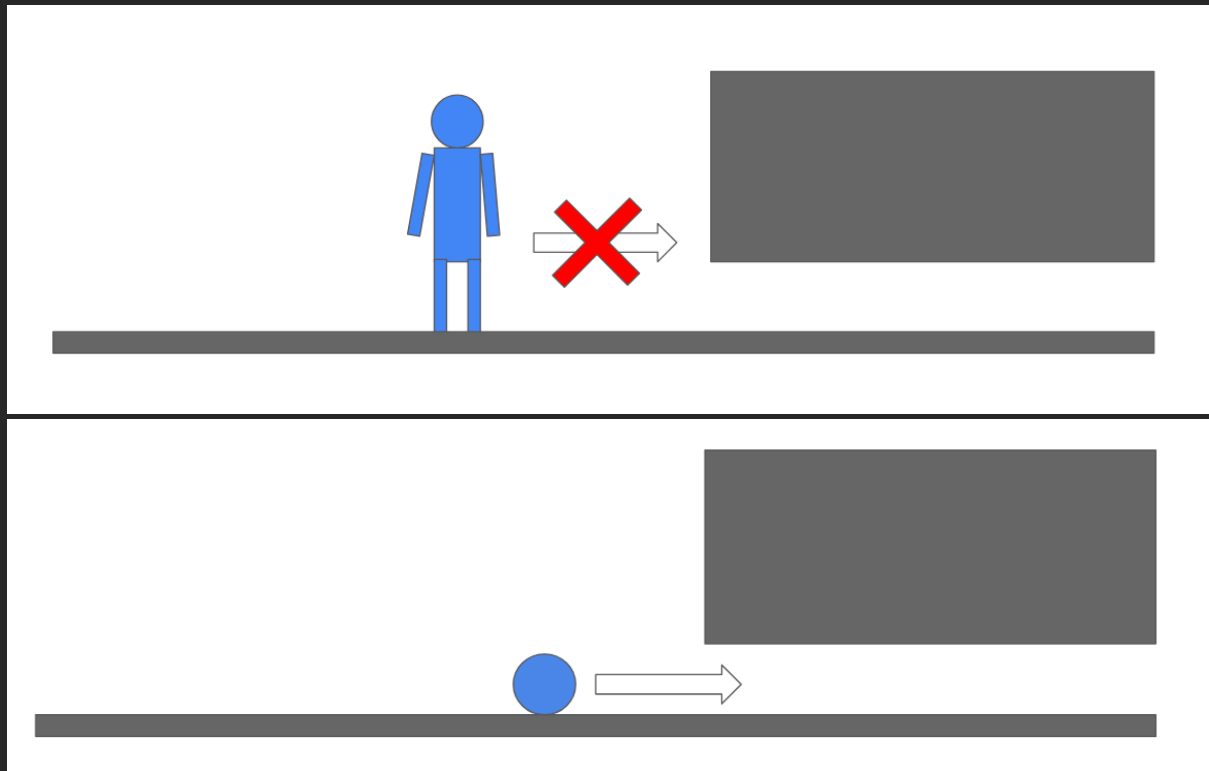
Moving Platforms:

Certain platforms will be able to move, going in and out of a wall, forcing the player to time their jumps based on their situation. This makes levels more dynamic and helps with replay ability by introducing slight randomness, whilst keeping the gameplay intense since the player is forced to make split-second decisions. There should ALWAYS be an alternative route or another platform synced up to be extended when the other isn't to ensure the player always has a path to continue and does not need to stop.



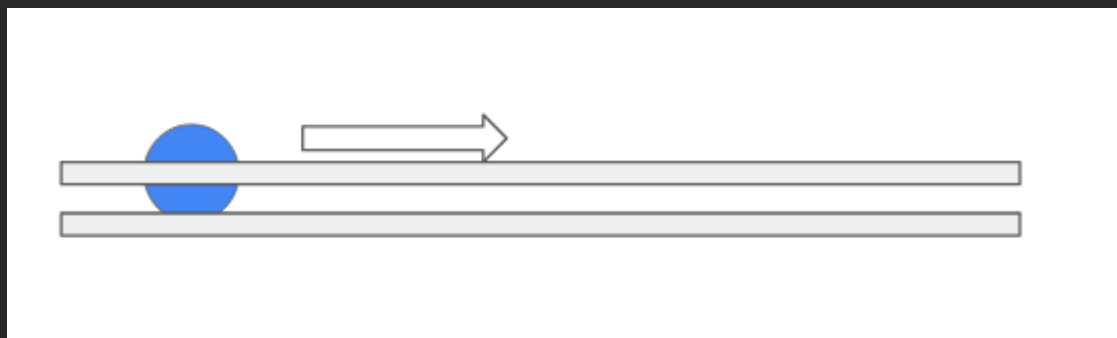
Roll Door:

A small circular door that the player needs to be in the roll form to fit through.



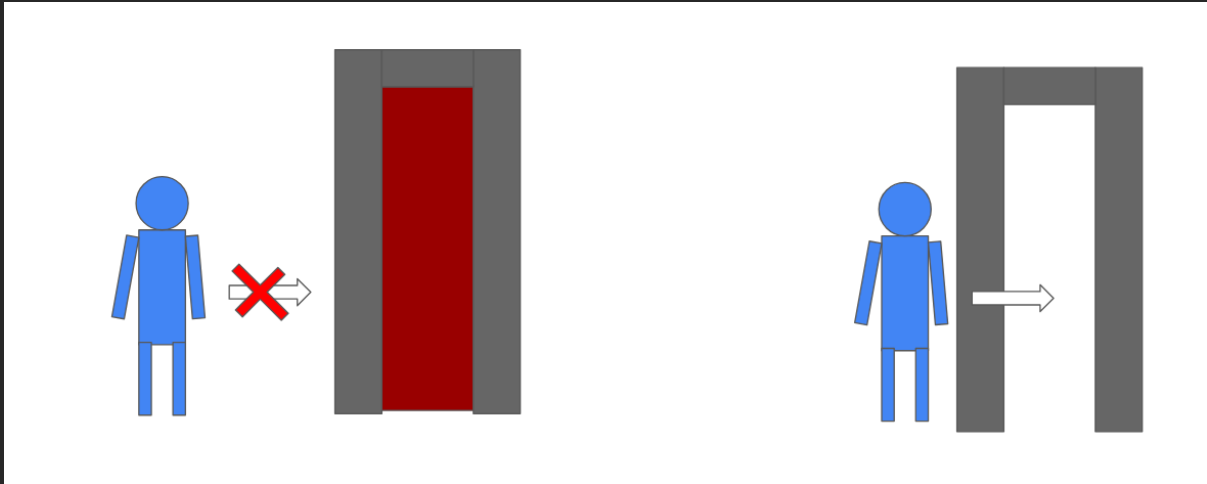
Roll Rails:

A set of rails (like pinball rails) if the player enters this while in the roll form they will roll along its path. They cannot switch to walk form while using the roll rails.



Toggle Door:

A door that will alternate between being turned on or off every few seconds. When the door is turned on there will be a wall that will disappear when it is turned off. The player can only go through it when it is off.



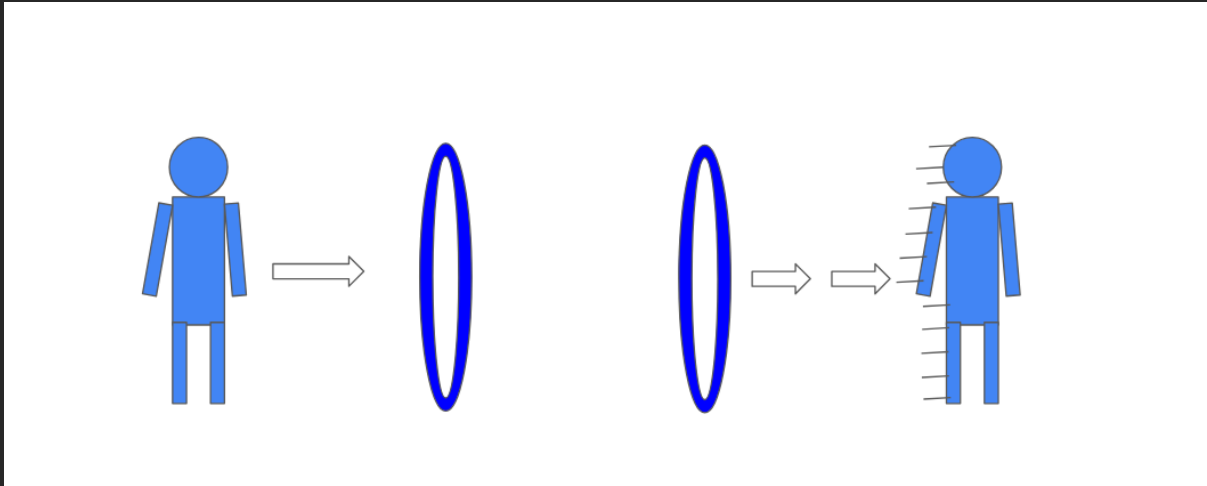
Toggle Platform:

A platform that will disappear and reappear every few seconds. The player can walk on it while it is there and will fall through when it disappears.



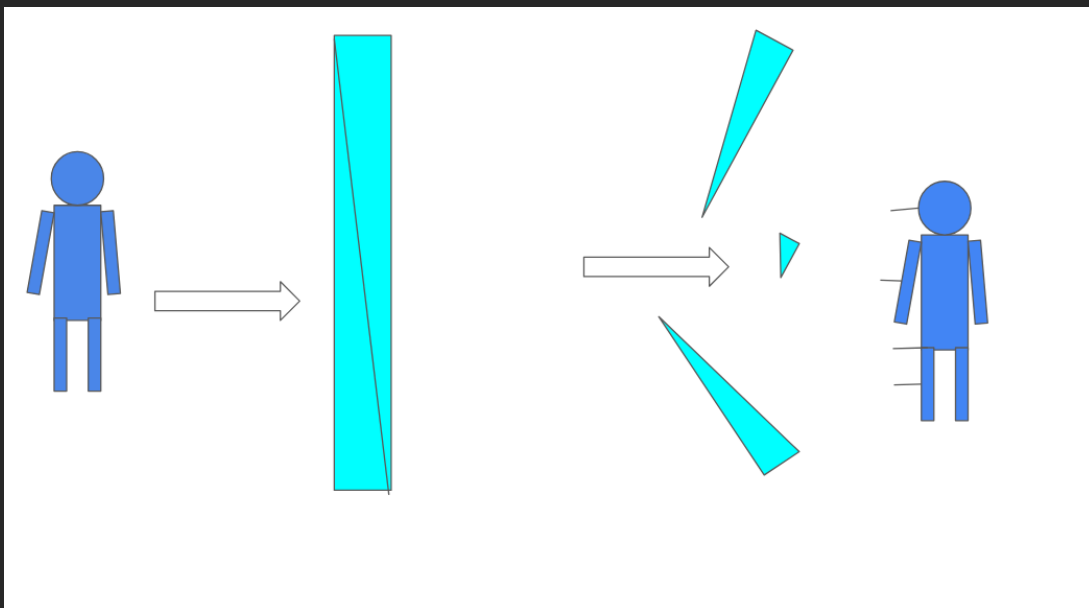
Speed boost:

Object with no collision, if the player crosses through it while in the roll form it will give them a speed boost in whatever direction they are travelling. Using this will give score points.



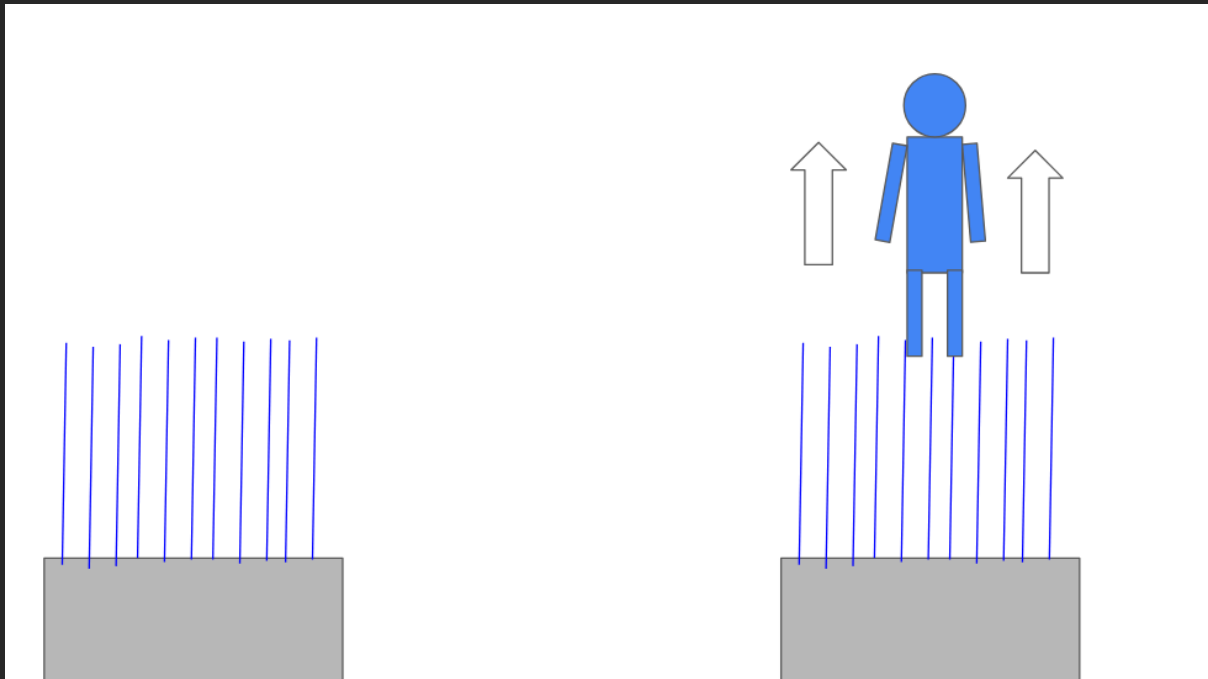
Breakable wall:

A wall that will be destroyed if the player rolls/runs into it at a fast enough speed. Breaking this will grant them score points.



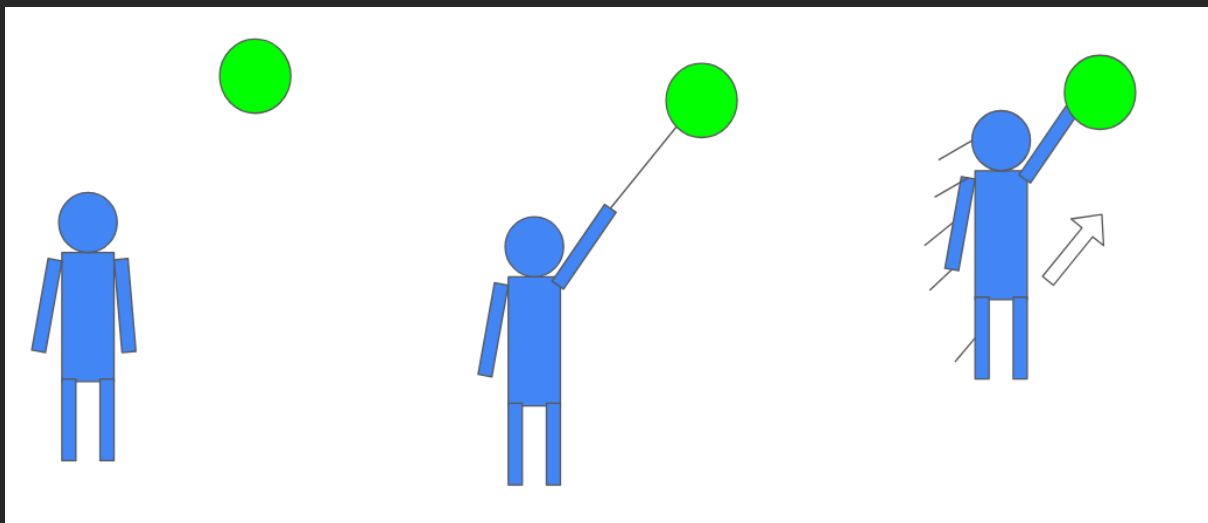
Jump pad:

A fan that is blowing a stream of air upwards, this air will push the player upwards if they stand in it.



Grapple point

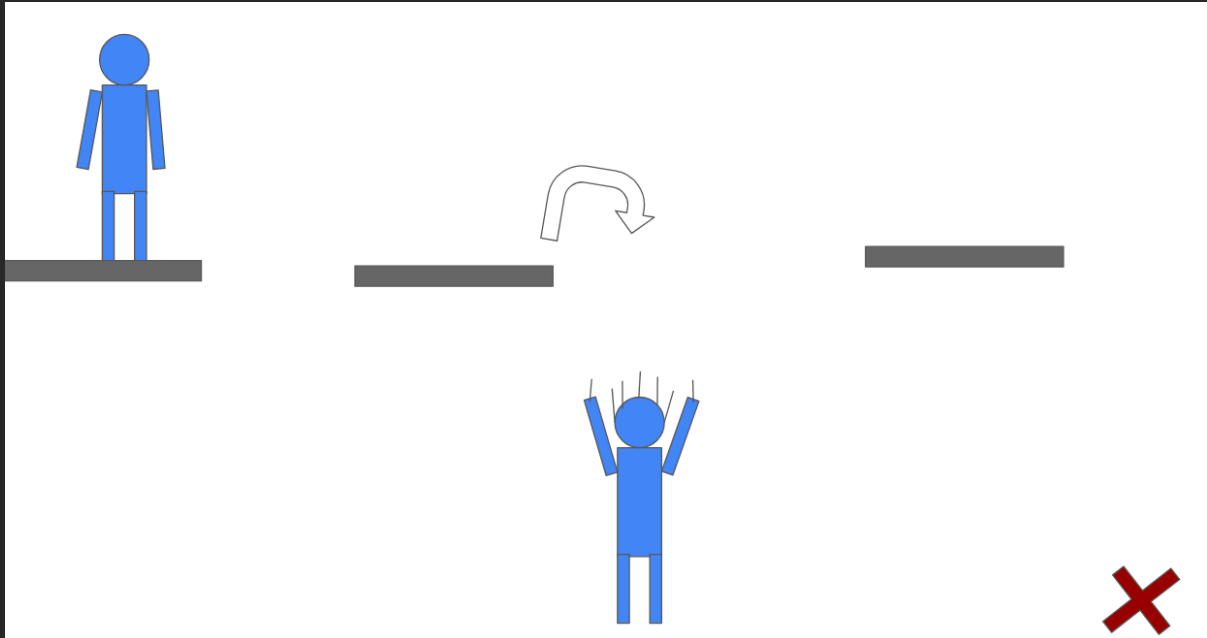
A floating object which the player can use their grapple on to be pulled towards it.



Hazards:

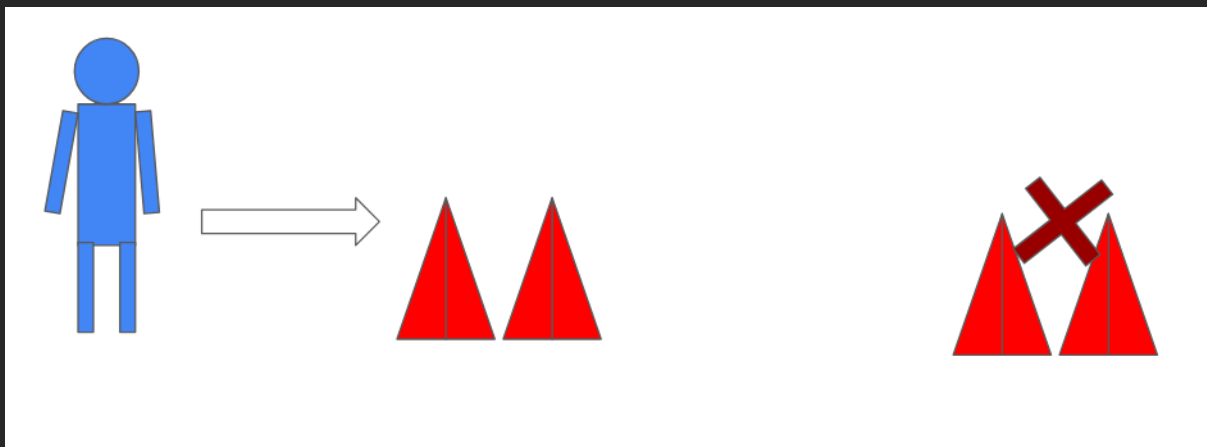
Bottomless void:

An endless void beneath the level that will kill the player if they fall into it.



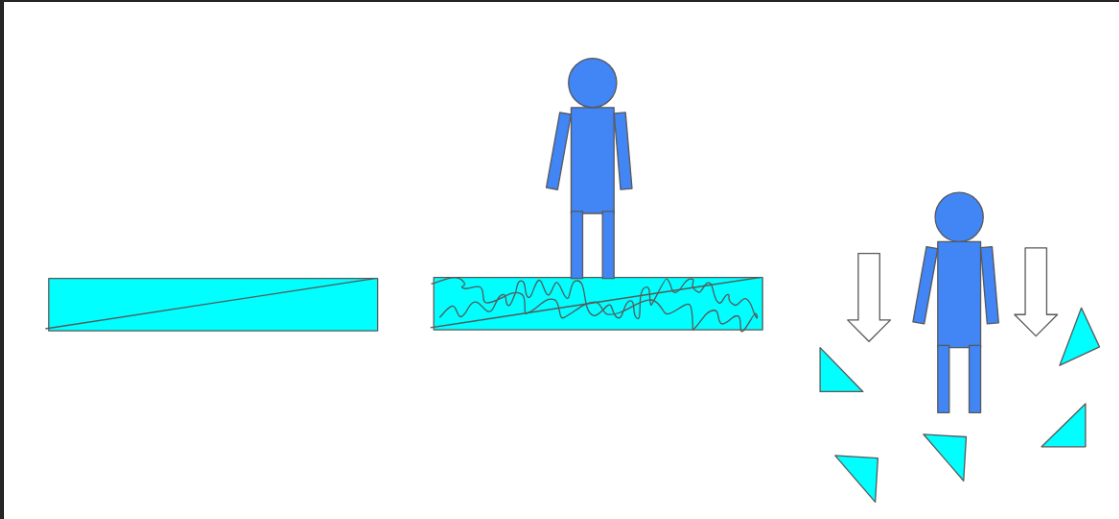
Spikes:

An object that will kill the player if they touch it.



Breakable floor:

A platform that breaks a few seconds after the player lands on it. Will reappear after a few seconds (likely around 3 or more) or when the player respawns. It will need some sort of indication once the player lands on it that it is going to break soon (something like shaking or cracks appearing).



Level Design

This is where you will document the design choices made with building the levels. This will include diagrams such as bubble maps and references to important factors such as affordance, pacing, pathways, tutorialisation and linear/ open world design.

Length

Levels should be 2-6 minutes long and contain multiple checkpoints which are evenly distributed to allow the player to not have to redo difficult sections once completed the first time.

Tutorialisation

The first level needs to have clear tutorialisation on how the mechanics work in a way which doesn't overload information to the player. Most of the tutorialisation should be diegetic as to not make the player read while they are moving fast. This can be done by slowly introducing the mechanics in a way that the player won't be overwhelmed, the first level should not make use of every mechanic, and the mechanics it does use should be done one by one.

Tutorial Billboards - We want to convey how to do something in as little words as possible. The diagram should have a picture of the player doing the action in a readable, clear way and have an icon of the button/ key/ mouse movement needed to do this. Every diagram should have the same scale and be fit on a modelled billboard which fits the theming of the game (like a browser tab maybe).

Pacing

The levels should be reasonably fast paced as the player will be on a timer. The levels need to keep a feeling of momentum, the player should always be moving and almost always advancing.

Linear/open world

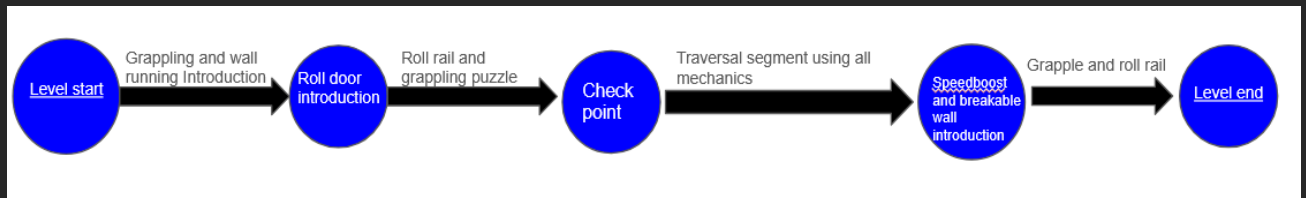
The levels should be fairly linear but with some hidden shortcuts or alternate routes.

Affordances

The way new obstacles are introduced to the player will reveal a lot of the context of how they are supposed to get around them (E.G introducing the

speed boost right next to the breakable wall will communicate that the player needs speed to break it) this can effectively used as a method of affordance. It is important that the context in which players encounter mechanics allows them to intuitively understand them.

Bubble Maps



Levels

Each level should have a goal in mind and focus to its content.

Level 1 – Introduce the player's kit and movement. Running form, ball form, wall running, grappling. Little to no world mechanics.

Level 2 – This level introduces the spikes as an obstacle, It features verticality and a higher level of difficulty than level 1.

Level 3 – The third and final level introduces fans and makes use of all previous mechanics, this level features a high level of verticality and since it is the final level it has the highest level of difficulty and makes use of every mechanic the player has encountered so far.

User Interface

General

Every menu user interface is controllable with both the keyboard and mouse or a controller. Move the mouse or use the left stick to navigate the menu. Left click or press the A button to select/ progress through the menu and press the escape key or the B button to cancel/ close a menu. When a button is hovered, it will be highlighted and make a sound effect.

Functionality

An explanation of what each feature will do in the UI diagrams.

Title Screen

The title screen is the first screen of the game that opens when booted.



Functionality

Start Game – Opens the level selection screen

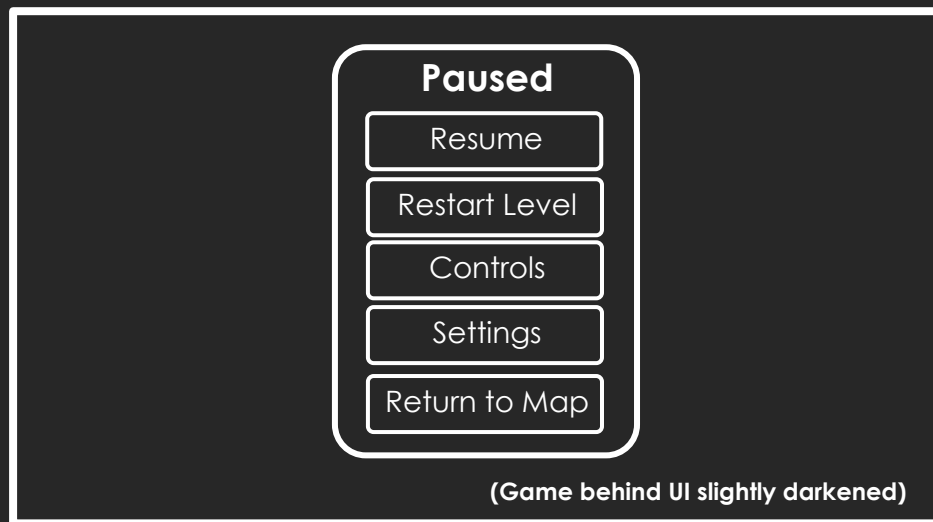
Controls – Opens the controls page of the UI

Settings – Opens the settings menu

Quit – Closes the application to desktop

Pause Menu

When the pause button is pressed, the pause menu opens. This menu will pause all gameplay as well as the in-game timer. The game and timer will continue when un-paused.



Functionality

Resume – Removes pause UI and game continues

Restart Level – Restarts the level with all level progress being reset

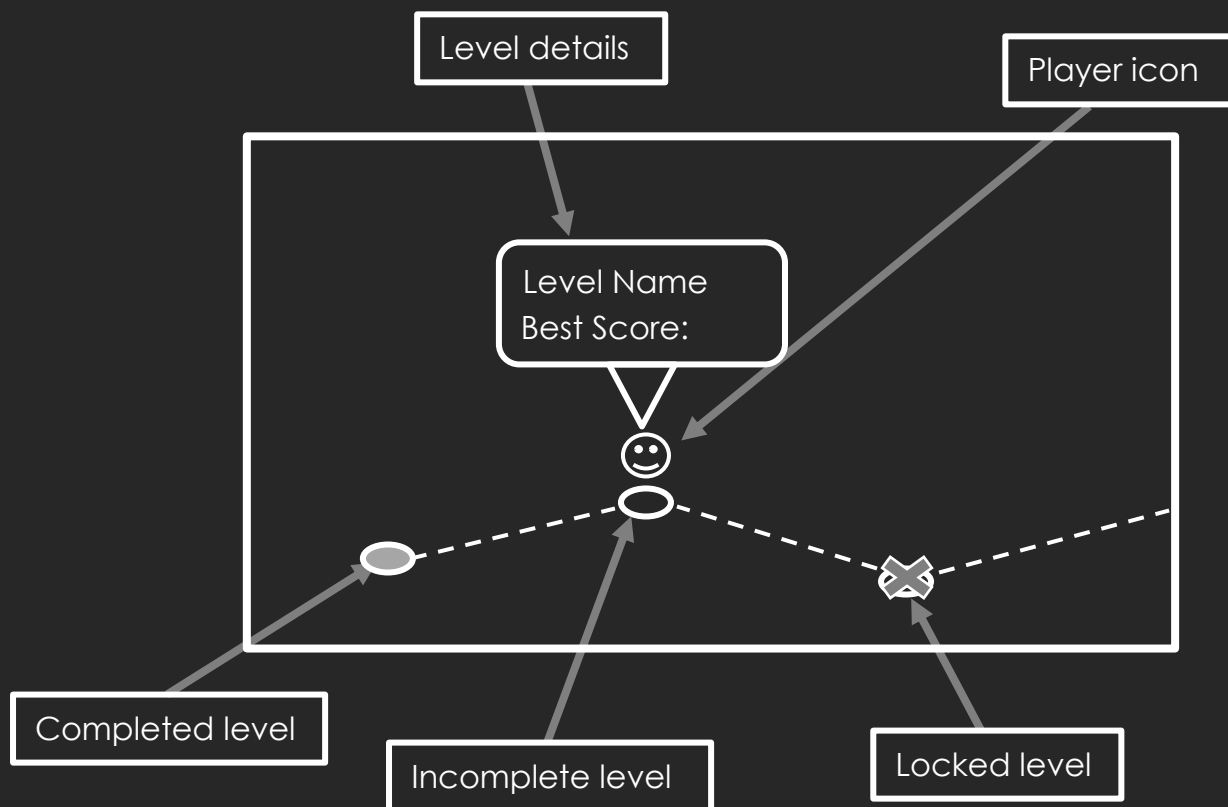
Controls – Opens the controls page of the UI

Settings – Opens the settings menu

Return to map – Closes the level and opens the level selection screen

Level Selection Screen

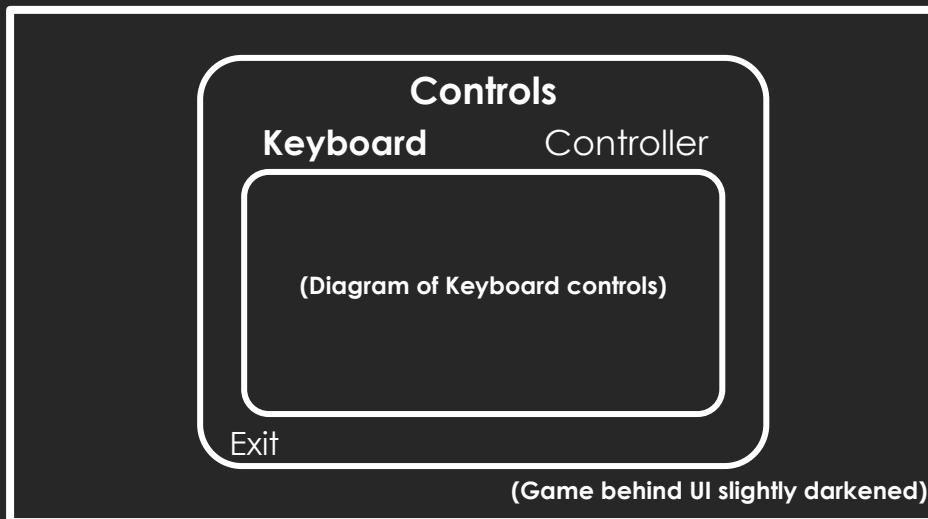
This screen contains all the levels available. Once the player completes a level, the next level will be unlocked. Hovering over a level will show the level name and the current high score for that level. If a level is selected, the player will go into the level. If the back button is pressed, the game will return to the title screen.



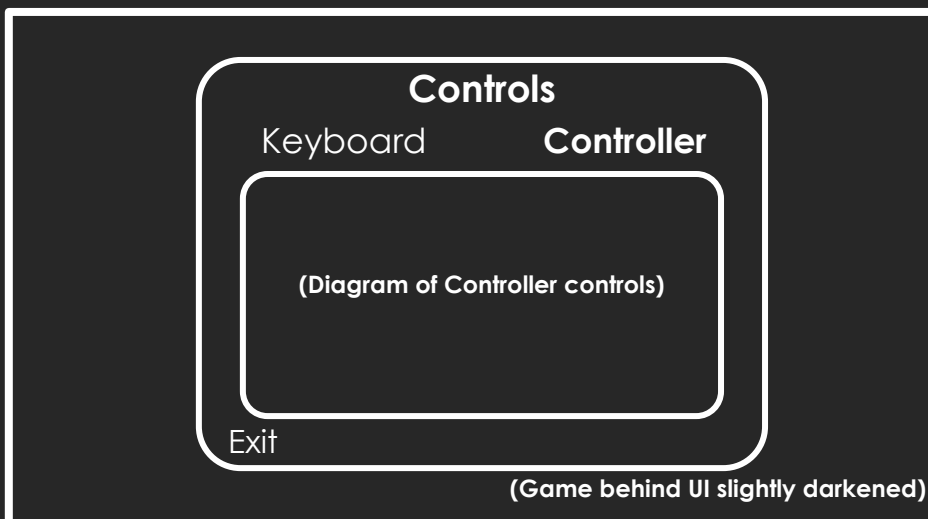
Controls Screen

The controls screen opens from pressing the controls button on either the main menu or pause screen. It will display either keyboard or controller controls depending on which one is currently selected, you can toggle which is selected by pressing on the names of the controllers.

Keyboard controls toggled



Controller controls toggled



Functionality

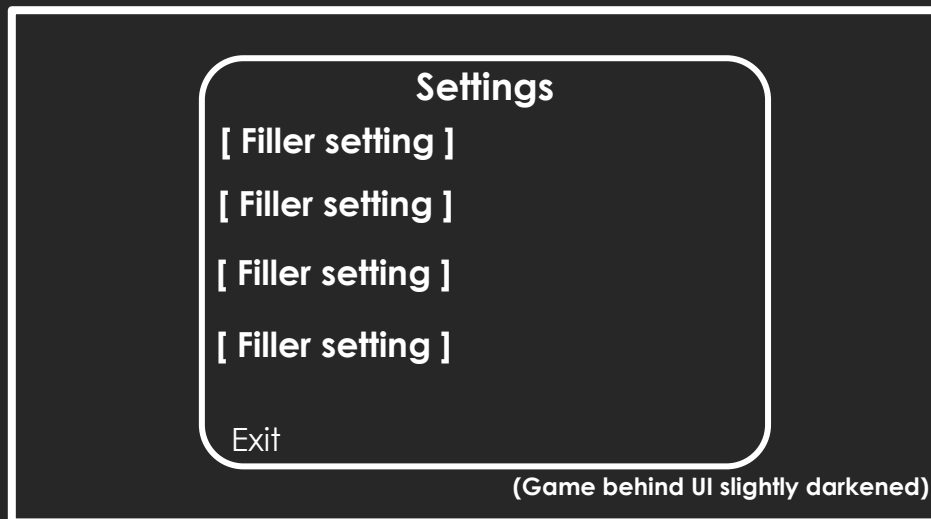
Keyboard – Changes the controls diagram to display keyboard controls

Controller – Changes the controls diagram to display controller controls

Exit – Returns to title screen or pause menu depending on which screen opened it.

Settings Screen

The settings screen opens from pressing the settings button on either the main menu or pause screen. The user can change settings to their preferences.



Functionality

Exit – Returns to title screen or pause menu depending on which screen opened it.

HUD

The HUD displays UI on the screen during gameplay. It displays the in game timer as well as the current score of the player.



Functionality

Current Score – A display of the player's current score as a number that updates in real time.

Game Timer – The timer for the game that goes up as it is being played.

End of Level UI

When the level ends, the final score is calculated in front of the player, and a rank is then awarded to the player.



Functionality

Game Score – The score accumulated during the level

Completion Time – The time it has taken to complete the level

Final Score – The final score calculated by the game score plus the score equivalent of the completion time.

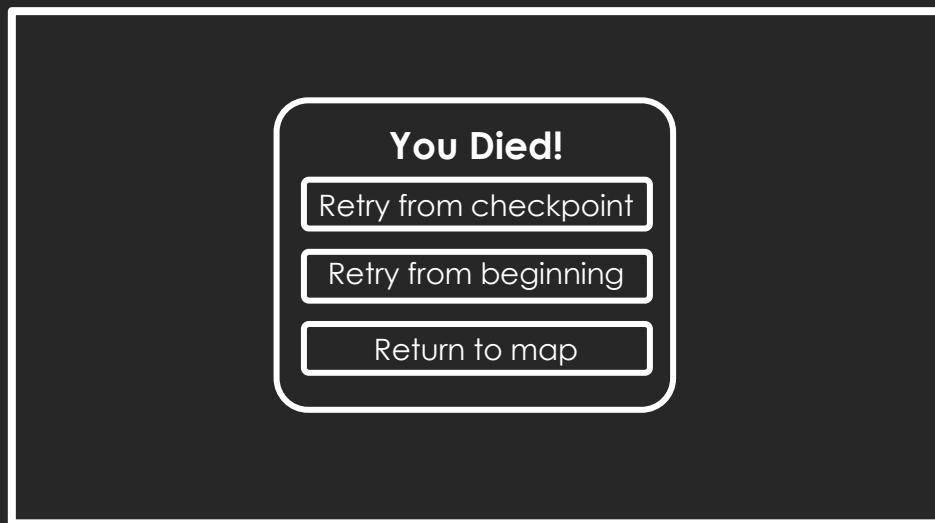
Deathless – A yes or a no which whether the player did the level in one go or not.

Final rank – The final rank awarded to the player calculated from their final score. If the player completed it deathless and reached A rank with the final points, they will achieve an S rank.

Back to map – Returns the player to the level selection screen.

Death Screen

When the player dies, this screen will pop up.



Functionality

Retry from checkpoint – Restarts the level from the last reached checkpoint.

Retry from beginning – Restarts the level from the beginning and resets the timer and score.

Return to map – Returns to the level selection screen.