jamk

Core Features of 3D Platformers

Anni Konttinen

Bachelor's thesis
December 2024
Bachelor's Degree Programme in Business Information Technology

jamk | Jyväskylän ammattikorkeakoulu
University of Applied Sciences

jamk

Description

Konttinen, Anni

Core Features of 3D Platformers

Jyväskylä: Jamk University of Applied Sciences, December 2024, 32 pages.

Degree Programme in Business Information Technology, Game Production. Bachelor's thesis.

Permission for open access publication: Yes

Language of publication: English

Abstract

It is important for a game developer to know the genre or genres they intend to work with. By having at least a basic understanding on what games in the chosen genre usually look like and what parts they consist of and why, one can utilize that knowledge in one's own game projects and save valuable time and energy in each project. Even if a developer chooses to go in a different direction with their game, they will at least know which "rules" they are breaking. This knowledge can help when planning for ways to put a new twist to an old genre, for example. The goal of this research was to shed light on 3D platformer games. The data for it was collected from the internet, from better known sources focused on video games.

The objective was to find out the core features of 3D platformers: that is, the features that are common and vital to games in the genre of 3D platformers. In the practical portion, the features were implemented into a small-scale video game or, in case implementation was not viable due to time or skill constraints, there was some theorizing on how they might be implemented in future.

Therefore, the results were a description of each core feature, examples on how they might appear in games, as well as a small demo for a platformer game. It was concluded that 3D platformer games have at least seven features that flag the genre.

Keywords/tags (subjects)

3D, 3D graphics, 3D platformer, Game design, Game industry, Level design, Platformer games, Video game industry, Video games.

Contents

Figures

1 Introduction

The purpose of this thesis is to find out the core features of 3D platformers and form them into a comprehensive list. In the end, a small-scale 3D platformer will be produced, in which the goal is to implement at least some of those features, depending on how feasible it is to do so within a solo project and with limited time.

The research part of this thesis examines the core features of 3D platformers. The thesis will provide a way for readers to see these core features as well as give them an outlook of what it is to work on such a project. The aim is to collect different people's views into one location in order to create a singular easily accessible paper on the topic, saving time in future research. The author of the thesis aims to become more competent as a game developer. These, along with an interest towards 3D platformers, were the reasons for the choice of topic.

1.1 Methods

The theory part of the thesis uses qualitative research as a method, the overall implementation method being research-based development assignment. Analyzing the data is done by examining which features the sources agree on. In cases where some features are only mentioned in singular sources, those features are ignored. This way, the list for core features for 3D platformers will be as comprehensive and reliable as possible.

The project that relates to this thesis will be created with Unreal Engine 5, using ready assets from the Unreal Engine asset store. The assets will be used because it is not feasible within a reasonable timeframe to produce all assets necessary for the game. The implementation of the project starts with coming up with and planning a concept for a 3D platformer as well as looking for suitable assets from the Unreal Engine asset store. It goes on with implementing the different core features that are deemed necessary for a good experience with the game. These features will be made into a list for easier visualization. Finally, there will be some last adjustments to the game, with the purpose of increasing the enjoyment of players.

1.1.1 Research-based Development Assignment

This method starts with a practical problem and questions relating to it (Jyväskylän ammattikorkeakoulu, n.a.). It combines theory and practice; use of prototypes is a vital part of the method (Wikipedia, n.a.). In case of this thesis, the practical problem is lack of knowledge about the features 3D platformers tend to contain, and the prototype is the game produced based on the research conducted.

1.1.2 Information Retrieval

The data for the theory section was collected from articles published by trustworthy sources and YouTube videos made by content creators. Keyword examples for finding information: 3D platformer, 3D platformer features, platformer games, best 3D platformers, as well as various 3D platformer games' names. The search tools used were YouTube and Google Search, and occasionally Google Scholar. Other sources such as books were also used, to an extent.

1.2 Research Questions

1. What is a 3D platformer?
2. Are there features that are common to titles in the genre of 3D platformers?
    a. If so, which features?

2 What is a (3D) Platformer?

A few key features used to flag 3D platformers. In the past, they used to be about "the ability to move around and jump between platforms in an attempt to reach a goal point". Nowadays, platformers have evolved and tend to offer players many possibilities when it comes to story and mechanics (Steward, 2013).

Figure 1

A Hat in Time (2017) is a 3D platformer

2.1 In Comparison With 2D Platformers

Camera: 3D platformers do not usually use the traditional view seen in 2D platformers, where the player tends to move horizontally or vertically along the X and Y axis. However, there are exceptions, such as Little Big Planet and the bonus levels in Crash Bandicoot. Even in these, 3D platformers usually add in the Z dimension – the possibility to move along three axes instead of two (Codex Gamicus, n.d.).

3 Core Features of 3D Platformers

3.1 Game Feel

“From the beginning of preproduction until the final game ships, design should include game feel” (Swink, 2007, para. 47).

Figure 2

Kao the Kangaroo (2022). (Nintendo, n.d.)

The idea of this “game feel” on a very basic level is that no matter how good a player is at the game, they should enjoy moving the character around (Boulton, 2018, 11:36). On a wider scope, game feel means literally how the game feels to play. The player receives this feeling via the sounds they hear and the shapes they see. When the game feels right, the experience of playing becomes more satisfying (Hacktic, 2023, 4:25).

Swink (2007) separates game feel into six factions: input, response, context, polish, metaphor and rules (Figure 3). Input is the commands the player gives to the game, for example, move forward. Response is the game responding to the commands: in this case, moving forward. This response should be as close to real-time as possible in order to ensure the best results; this way, the player does not need to wait for the game to respond to input. Context, on the other hand, is making sure that the movement compliments the levels and vice versa. In other words, there should be elements in the levels that invite the player to use certain abilities: the abilities should work well with the environment. Polish is the little things: a particle effect that indicates when one thing collides with another, or a slight camera shake when shooting a weapon. These effects can have a surprisingly big effect on how the game feels. Metaphor: the expectations and connections to real

life elements that the player makes based on what looks familiar. For example, if the player character looks like a regular human, the expectation would be that it moves like a human. This helps give the player an intuitional idea on how the character should work. Lastly, the faction of rules is about long-term objectives that should naturally rise from moving the character. For example, the objective can be to get to the top of a hill or to collect a certain number of collectables within a level.

Figure 3

Game feel factions (Adapted from Swink, 2007)

Game feel factions (Swink 2007)	Description of faction	Example
Input	Commands player gives to the game	Move forward
Response	Game responding to player's commands	Moving forward in real-time
Context	Movement and levels complement each other	Certain areas invite using certain abilities
Polish	The little things	Particle effects
Metaphor	How something should work based on how it looks	A human
Rules	Long-term objectives arising from movement	Player must reach the top of a hill

One can test game feel by stripping away everything but the basic features in a game, leaving only the most crucial parts such as movement of the character (Game Maker's Toolkit, 2015, 1:03). Using this method makes it easier to notice when something does not feel right, as the tester will not be distracted by other, possibly alleviating things such as environment or music.

For example, if the player needs to be able to make precise jumps often, this type of testing might reveal that having a static camera serves best for this purpose, because the player does not get distracted by its movement. Another thing could be that adding friction to the character makes those jumps feel better; the movement becomes more believable (Game Maker's Toolkit, 2015, 4:35).

Swink (2007) notes that very often it happens that game developers do not pay enough attention to game feel, thinking they can handle it at the end of the production. They do not spend enough time getting right the one thing the player is going to do for the duration of the game experience: controlling the character.

It is difficult to master game feel, as it can take several months or even years to get it right (Game Maker's Toolkit, 2015, 4:53).

3.2 Engaging Movement

A character that feels good is responsive to controls. It jumps when the player presses "jump", without pausing to think or prepare to jump (Hacktic, 2023, 4:41). When planning for character movement, some crucial factors to consider are how it feels to move the character and how the movement plays along with different obstacles.

In contrast to 2D platformers, where being able to make precise jumps and move in an exact manner in order to progress is key, 3D platformers tend to give more lenience to the players. 3D platformers usually help the players with jumping by putting the character's shadow directly underneath the character, compensating for the constantly moving camera and varying perspectives. In addition, 3D platformers often have a few moves that can be used in a "chain" with the jump move, making it easier to change directions mid-air or jump a larger gap. Examples of these additional moves are double jump and gliding (Hacktic, 2023, 3:10).

Overall, when designing character movement, it is important to consider how different kind of movements work together and how they look when performed one after another. Knowing the abilities of the player character will help with the level design progress, as those abilities can be used to design obstacles and problems that can be solved with the usage of those abilities (Sinclair, 2020). For example, in one of the levels in Crash Bandicoot (2017), the player must get under an electrical fence and over a ravine right after the fence. This requires combining two abilities by performing them one after the other: first, the player must use Crash's "slide" ability, and after that they must be fast enough to use the "jump" ability, or they will slide right into the ravine. If these two abilities were something different, it is likely that the challenge would also look different.

3.3 Hub World

“We need to work with not just the movement, but the space that you move within” (Boulton, 2018, 7:42).

A hub is a point of interest, about which action and events take place (Merriam-Webster, n.d.). A hub world is usually a level that players come back to between levels. Access to new levels is found within the hub world, which makes the world a sort of “home” for the players. These worlds tend to hint at the theme of the levels they offer access to (Everman, 2021). This is often done by changing the level design at a certain location within the hub world (Figure 4). This will also help players navigate back to the level later on, should there be need.

Figure 4

Crash Bandicoot 2: Cortex Strikes Back (2017). The designs of the portals hint towards two levels with different themes. (Video Games Uncovered, n.d.)

Hub worlds encourage players to explore their character’s movement. If the player is having fun just by playing around in the hub world, there is a good chance that the character’s movement is appealing enough and works well for the game (Hovermale, 2018). Therefore, it seems that a hub world is a convenient way to see if the game itself has potential.

Sinclair (2020) advices that developers add shortcuts in their hub worlds. These can be useful for more advanced players, or simply at a later point in time, when the player has gained enough abilities to access them.

3.4 Landmarks and Points of Interest

In a clearly designed map, the player can easily understand what the map is like and what the points of interest are. In these well-designed areas, landmarks are used to help the player locate themselves and get to where they want to go (Hacktic, 2023, 3:26). The first level of Yooka-Laylee (2017) utilizes landmarks effectively: especially the tall tower at one corner of the level (figure 5) is an excellent navigation tool, because it can be seen from nearly every location on the map.

Figure 5

Yooka-Laylee (2017). In addition to the tower structure in the back, many smaller elements (such as the temple near the center) have unique designs. (Pandaboy78, n.d.)

"A good platform environment also needs clarity so that people can clearly identify where they can traverse," says Steven Hurst in an interview by Game Developer Staff (2015, 5. Worlds Must… section, para. 3). Landmarks make this clarity more achievable.

In addition to guiding the player through levels, landmarks make them more memorable (Galuzin, 2008). They can act as the thing the player will later associate the level with. Francis (2021): Landmarks (also known as “weenies”) can also be smaller things, such as a rock of unusual color or a gate in the middle of an environment that is otherwise in its natural state. If a game has no landmarks, the player will have to excessively rely on any maps they might have available and stumble through unrecognizable locations until they find what they are looking for.

3.5 Variety

With movement, variety comes from different abilities and ways to combine them in ways preferable to the player or the situation (Hacktic, 2023, 0:54). However, variety is needed in other aspects of platformer games. In levels, for example, this can be reached with unique enemies in each level and levels with different themes, as well as different types of obstacles from logical puzzles to deep pits (Lantz, 2023).

Generally, the first level of a 3D platformer is a “safe” space with less enemies than other levels. The obstacles are not too difficult to pass, and the variety in terrain and different materials is not too overwhelming. The purpose of this is to give the player an opportunity to try out the character’s movement and abilities and get an idea of what is expected from them (Boulton, 2018, 7:47).

Figure 6

Agent Hugo: Lemoon Twist (2007). The first level. (Orlong, 2017)

It seems to be common for 3D platformers to have “boss battles” at certain intervals. These can happen for example at the end of a level, or when progressing into a new, larger area of the game. Boss battles test the player’s mastery of game controls and adaptivity to different attacks from the enemy. While the attacks keep coming, the player needs to be able to keep an eye on opportunities for counterattacks. In these fights, the pacing is faster than with normal enemies, so good reflexes are often key (Boulton, 2018, 6:58).

For example, in Crash Bandicoot (2017), there is a boss battle every few levels, 6 battles in total. One of the boss battles is against a character named Ripper Roo. The player avoids exploding TNT objects for three rounds, and after each round the enemy gets stunned by the explosions if it is close proximity of exploding TNT. This is when the player has a chance to do a counterattack. If they manage to do it, the enemy moves into the next phase, becoming more dangerous: it places

the explosives into more platforms than before and also moves faster. These increases in difficulty keep happening until the player defeats the enemy in the last – the third – phase **(**figure 7).

Figure 7

Crash Bandicoot (2017) – the fight against Ripper Roo. (CrashBash2000, n.d.)

3.6 Collectables

Collectables can be used to lead the player along a certain path to emphasize which direction the game wants them to go. By keeping the chain of collectables clearly visible on the main path and hiding them further down secret ways is a technique often used by platformers. That is just one way that collectables are useful. In addition, they add to the players' enjoyment, not because of the rewards they offer but simply because collecting them feels good. Once the player knows what it feels like to collect them, it enforces them to keep collecting in the future (Hacktic, 2023, 4:01; 5:49).

The reasons that picking up collectables feels good may vary, but in its core, it comes from satisfying sound effects and the anticipation of getting to collect more and having them add up to the total number of collectables usually shown on the screen during gameplay (Figure 8) (Hacktic, 2023, 7:04).

Figure 8

In Lego Indiana Jones: The Original Adventures (2007) (Dark52, 2020)

Note. As in other Lego platformers, the main collectable in this game is studs. Once the player has enough, the stud counter in the specific level becomes full and therefore completed for that level.

It should not be necessary to collect every little thing in the game: it should be possible to complete the story even when some collectibles are skipped or simply missed. This leaves opportunities for "completionists", who like to collect everything they can and are willing to go an extra mile for that (Boulton, 2018, 10:23).

A way to make collectables more rewarding to pick up is to add in an effect that rises in pitch once the player gathers many collectables in a row (Game Maker's Toolkit, 2015, 3:40). For example, Crash Bandicoot (2017) does this. If the player is fast enough to collect one collectable after another, the little sound effect that comes with each collectable rises and rises in pitch, until the time between one collectable and the next becomes too long or the player has collected so many of them that the pitch no longer rises higher. Having the game randomize the "pick-up" sound effect out of a few options also adds in variation, if rising pitch is not what the developer wishes to implement (Game Maker's Toolkit, 2015, 3:40).

3.7 Story

It is usual that the main plot of a platformer takes place within the hub world. The levels act as "extra" content where players go to collect a certain number of collectables before they can unlock a new area of the hub world and advance the story. The levels may have some story to them, but in general, the main story advances inside the hub world (Tv Tropes, n.d.). This is the case with Yooka-Laylee (2017): the player goes through different worlds, collects a number of items in each world and once they have done enough of that, in addition to purchasing a new ability or two from a specific side character, they can advance the story in the hub world. The new abilities help gain access to the next level, and on the way to the new level there is a cutscene that advances the plot.

Figure 9

In Yooka-Laylee (2017), the player goes against Capital B.

One might think that the choice of the player character's looks comes from the story: however, in platformers, the player character is chosen based on what attributes they offer to the game (Stuart, 2017). This is why many platformers use animals as their characters. For example, in Yooka-Laylee (2017), the main characters are a bat and a chameleon. These creatures naturally offer the

player the ability to use the bat to fly or use echolocation; the chameleon allows the player the cover of camouflage and lets them grab collectables with its tongue.

Despite the way the player character is chosen, many platformers aim to tell a story instead of focusing on challenging the player's skills. These games allow just about anyone to experience them, while skill-based games are aimed for those who want to challenge themselves and become good at something (Bycer, 2020, 3).

4 Research Implementation

4.1 Planning

The aim was to create a small-scale 3D platformer. The game's style would be more low-poly and cartoon-like rather than realistic. The choice of style helped make the game feel more lighthearted, as was the purpose.

The game engine chosen for the project was Unreal Engine, and more precisely its version 5.0EA (Early Access 2). The engine was chosen for learning purposes and also for interest in visual scripting. The Early Access version was chosen because at the start of the project it was the latest available version and the only way to access Unreal Engine 5.

The project was planned with the help of a notepad app for Android (Samsung Notes), creating a bullet point list of the minimum features that were wanted into the game. A list of things that might be added later on was also created in case after the core features were completed, there still was interest in continuing to work on the game. The game was given a working title: "Cat Game."

Functions that were felt to be the bare minimum necessary for creating a working game presented in Figure 10. The core features of 3D platformers for this project would be considered later. This was done because at the start of the project, no research was done on what 3D platformers tend to contain, instead going with what felt right. Adding other features would also be considered later, once the basic idea for the game was implemented.

Figure 10

List of features planned for Cat Game's first demo version

Function	Subcategory	Requirement						
Interact with found cat		Number of cats in UI rises with 1 (Cats found 0/4 -> 1/4)	Cat does not disappear	Cat says a random line as greeting	Cats have collision			
Enter next level		All cats from current level must be greeted	Display message once all cats are greeted	Display message if trying to leave early	Level number rises with 1 when entering new level	Game finds next level based on name of current level		
Randomize hiding cats' spawn		Out of a few locations	Two cats cannot be in the same location	Randomize locations at start of level	Different types of cats in each level	Different amount of cats in each level		
UI	***Main menu***	Resume button	New game button	New game button prompts: "Are you sure you want to start over?"	Within confirmation menu, buttons for "new game" and "cancel"	Level select button	Options button	Quit button
	Pause menu	Appears with Esc, disappears from Esc or Resume button	Game pauses	Content (buttons): Resume, Options, Quit to Main Menu, Quit				
	Level select	Icons or buttons for each level	Arrow for "return to main menu"	15 levels in one view	(Arrow to see more levels)			
	Options	Music on/off button	Sound effects on/off button	Volume slider				
	Game UI	Cat counter	Level number	"Can't leave yet" - text	"Greeted every cat in the area!" - text			

4.2 Setting Up

Unreal Engine offers a few ready templates to choose from, to make starting a project easier. For Cat Game, the choice was the template called "3rd person template" because the game was intended to be played in third person. The third person point of view was chosen because it allows the player to view the character they are controlling, which has been found a common feature of platformer games.

4.3 Player Character

The player character, as well as the other characters in the game, were from an asset pack called "Low-poly cats" (Radik Bilalov, https://www.fab.com/listings/6d62c947-14a3-401a-84d7-6d4667a1dd03). This asset pack was used because it was once one of the "Free for the Month" packs, a selection of which Unreal Engine store offers each month. It seemed suitable to have a cat

as the player character because of the developer's fondness of cats. The default character was replaced from the third person template with one of the cats from the asset pack.

Figure 11

Player character

4.4 Animation

The "Low-poly cats" asset pack also included animations for the cats, so there was no need to create those. This helped save a great amount of time in the game creation process. Some of these animations were applied to the player character, for example making the "run" animation play when the player presses down the shift-key (run). These bindings for certain actions were made by creating a new key binding in the game settings and within the 3rd Person Character blueprint, adding nodes that make the animation happen when a certain button is pressed.

Figure 12

Animation graph

4.5 Levels

For the levels, a landscape was used for the ground. The asset pack “Greenwood Fantasy Village” (NOTLonely, https://www.fab.com/listings/77a8dfc4-b0b3-4882-96a6-b454b2dc5747) was used for the assets within the levels. The Fantasy Village pack was chosen because it fit together with the Low-poly Cats pack.

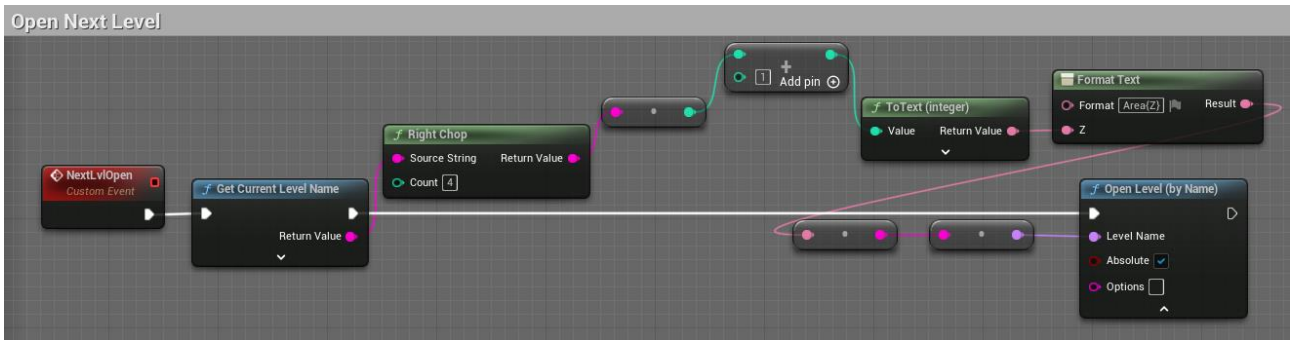
The landscape was sculped to make it more interesting than a flat plane and create variation that, in addition to the textures themselves, would help distinguish one type of ground from another. For example, where a path runs, the ground around it was left a little higher than where the path is intended. A grass material was applied for the basic texture of ground, after which other textures were painted, such as paths and sand. This painting was done by using brushes available in the game engine. For large groups of foliage, the desired objects (such as trees) were selected and the brush settings adjusted to suit the needs of the level. Some examples of adjusted settings are density, rotation, and size. The foliage brush was then used to paint where the objects should appear. Using the foliage brush saved a considerable amount of time, as instead of placing every object individually, it was possible to place them on the ground with a few brush strokes. Some of the foliage did not go where wanted; these were later removed using the erase brush to polish up the level.

4.5.1 Level Change

Changing levels was handled with a trigger area that unlocks once all cats are collected within a level. A text appears on the screen: “Greeted all cats in this area! Time to move on.” Once the player enters the trigger area, they are transported into the next level with the help of a script (image below) that reads the current level number and adds one to it. This was done because using this method solves the problem of having to create a separate node for each level, which would bloat the code and cause the developer to have to do changes to it every time a new level is added.

Figure 13

Blueprint setup for opening the next level



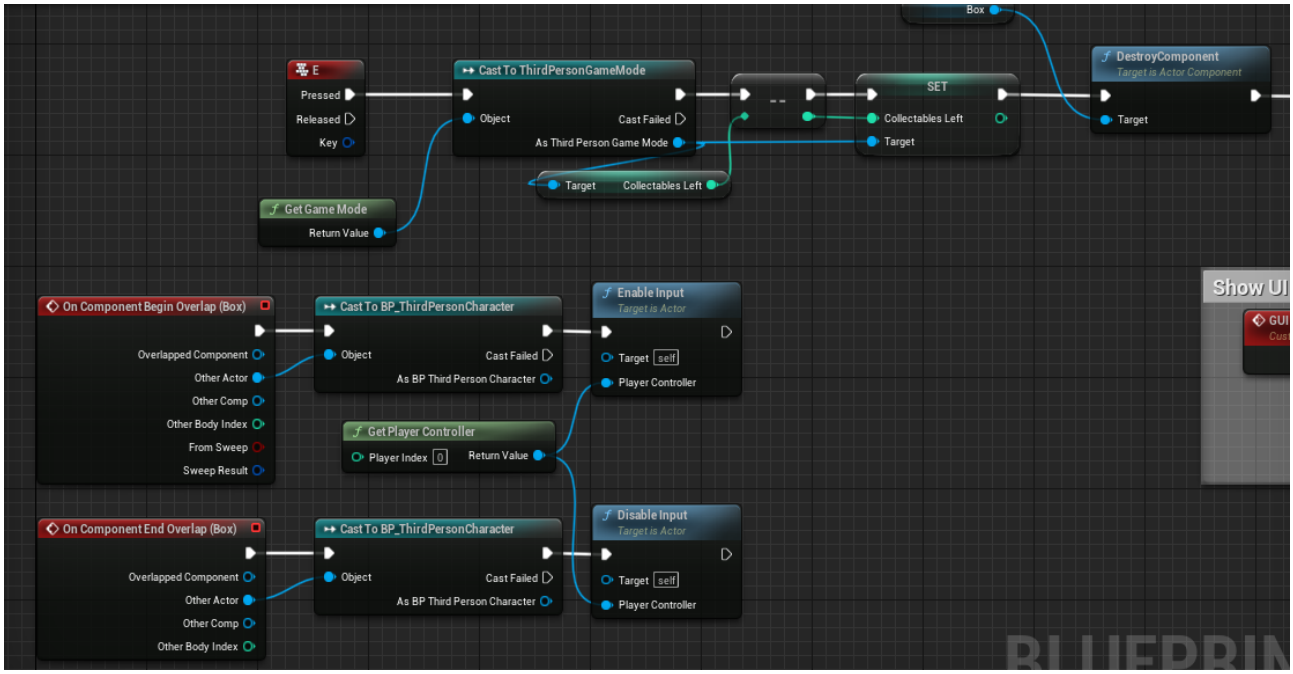
Note. Each level was named using the format ”AreaNumber”, so ”Area1”, for instance. In this blueprint, the word “Area” was chopped away, leaving only the “Number”. By adding 1 to this number and retrieving the name of the current level, then combining them, the blueprint finds the name of the next level and opens it.

4.6 Collectables

The collectables in the game are cats. When the player character goes close enough to them, they can interact with the other cats with ‘E’. After that, the cat says a randomized line. The player can only interact once with each cat in each level. This was done by adding into the collectables blueprint a command that destroys the invisible box around the interactable after the interaction. The box is the collectable part; removing it prevents further interaction with the cat it is tied to.

Figure 14

Part of the greeting blueprint

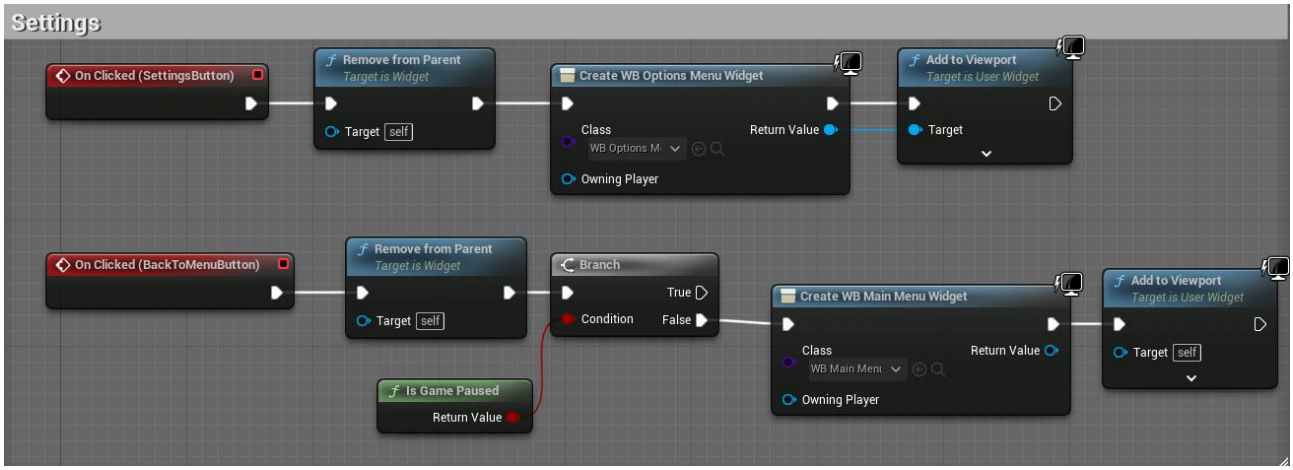


4.7 Menus

The menus were made using widget blueprints. They consist of a canvas panel and on it are different buttons that can be pressed. These buttons lead to other menus, which removes the previous menu from sight with a simple blueprint.

Figure 15

Settings menu blueprint setup.



4.8 Sound

It was decided that for this small demo, no music or sound effects would be implemented because of lack of personal interest, as well as to save time.

4.9 Testing

Testing was conducted periodically to ensure that all features work as intended in the final product and that there are no differences between Unreal Engine game view and the separate game build. Running the build worked as intended.

4.10 Core Features applied to “Cat Game”

The chapter handles how core features of 3D platformers, which were discovered during the research phase, apply to or were applied into this game.

4.10.1 Game Feel

The controls are responsive to the player’s input and responses happen in real-time. For example, there are no delays in responding to movement commands. Context faction in Cat Game: The obstacles that the player is meant to climb are only as high as the character can jump. There is also use for the run ability, as the distances between points of interest may feel dull to walk, if the player wants to complete levels quickly.

Polish: this factor is the most lacking in Cat Game, because it is yet to have any particle effects or other examples of polish. This is because of lack of time. What comes to the metaphor faction, however, Cat Game does make use of it. The main character of the game is a cat, and this fact creates certain expectations about how it should walk and how high it should jump, as well as its running speed. Because it has no wings, for example, the player will not expect it to be able to fly.

The rules of Cat Game are to greet all cats inside an area and thus open access to the next area. Therefore, the biggest long-term objective that naturally rises is to reach cat X by clearing obstacle Y, such as scaling a large boulder.

4.10.2 Engaging Movement

The movement in Cat Game is unfortunately on a very basic level, due to using only a few of the default animations that come with the “Low poly cats” asset pack.

4.10.3 Hub World

Cat Game does not have a hub world that the player returns to after each level, because the cat character wanders the world, wanting to see new places and new cats. Therefore, there is no need to return to a previously visited location.

4.10.4 Landmarks

Every level in Cat Game has a landmark that is the most remarkable thing in the level. In addition, smaller landmarks such as special rock formations are used. This helps the players navigate the environment more easily and find their way through an already familiar area. Adding very tall landmarks such as cliffs might be something that could be implemented in further levels, as they could be seen from farther away than regular buildings or natural formations.

Figure 16

Central landmark of level 2

4.10.5 Variety

Variety in Cat Game is achieved by designing environments that look and feel different and having them vary from level to level. The changes aim to be reasonable, however: if the previous level takes place in a summertime forest, the next setting must not be a snowy mountainside without proper preparations in the former level. For instance, the first level is on a farm and the next is in the forest right next to the farm.

Variety is also considered with the cats in the game: there are a few different colors from white to orange and even hairless. The different kinds of cats make meeting one cat after another more interesting, as they are more than mere copies of each other.

4.10.6 Collectables

The only collectables in the game are other cats. This decision was made in order to comply with the straightforward concept behind it.

4.10.7 Story

The story of Cat Game is simple: a cat leaves home to meet other cats. It wants to greet as many cats as possible and see the world. This explains the player's need to leave an area whenever every cat within that area has been greeted, and why they cannot leave before completing that objective. The absence of a deeper plot allows for a very casual experience that is easy to return to even after a longer break, because the usual requirement of remembering recent plot points is removed. It also makes it easier for young children to play the game, since the idea can be grasped even without the ability to read.

5 Results

5.1 Discussion

There was surprisingly much struggle with getting the level number show at the bottom of the screen. At some point, there was also a problem with the UI; it would only appear once the player interacted with the first cat. Solving this issue was a matter of moving a node from one place to

another, however – something that became clear to me when I got a little more familiar with how blueprints work.

The feeling of managing to solve a problem on my own after looking for an answer from the internet for a long time was certainly an important highlight during the project. I am most proud of figuring out how to change levels without creating a separate node for each level, because there was no clear tutorial on this and so I needed to puzzle it out by myself. I also managed to make the level counter work better than in the tutorials I reviewed.

I would say that just the right amount of planning was done before starting to build the game: for me personally, spending too much time on the planning portion tends to drain the excitement from a new project.

I believe that I managed to find answers to my research questions in the theory section, because I now understand what a 3D platformer is. The core features of 3D platformers were found and collected in this paper, answering questions 2 and 2a.

5.2 Why These Core Features?

I had my doubts about landmarks being one of the core features and thought of merging it into variety, but I decided not to. This is because removing landmarks from a 3D platformer would make the environment very generic. It would feel that way because landmarks are a big part of what makes a level memorable and interesting to explore. Landmarks as a core feature made me realize that my levels did not have any larger landmarks. Perhaps they do not need them because they are so small, but in my opinion, it would be good to add them in order to make them more memorable. For example, in level 2, the landmark could be a large rock at the top of a hill.

Game feel, on the other hand, is a core feature for 3D platformer games because if it is not right, the whole gaming experience suffers. How much, depends on what exactly is off about the feel.

6 Conclusion

The research should allow game developers to gain an understanding of what 3D platformers generally contain, in a way that can help save valuable time. This is because they may not need to go through all the same sources that were used in the research: instead, they may look at the research and get a comprehensive understanding of the topic from one source that has compared and summarized multiple sources. In addition, anyone else wishing to delve into the topic should be able to with this research paper, as the terminology used is aimed not to be overly complicated.

Sources were used and no claims of superior knowledge were made by the author. While YouTube videos are hardly scientific, the fact that many of the sources talk about similar information adds into their reliability, as it is evident that they share their views on the topics at hand. If there were only one or two rather unknown sources mentioning some concept, it would not be as reliable. Used sources were selected as objectively as possible.

If one were to replicate my research, the results should be much the same in terms of the core features. The implementation should be different for everyone, since each individual has their own ideas and interests. Successfully replicating the research would add into the reliability of this work, as it would showcase that someone else could reach the same results.

6.1 Future Developments to Cat Game

In future, the game could be developed further by adding more levels. Additions such as enemies could be made. The enemy could be for example a dog, or in areas where wild animals may appear, a wolf or a bear. The players would need to take care to stay away from their line of sight for example by hiding, and they could get past the enemy by sneaking when they are not looking or simply by taking a detour. There should be cases when sneaking is mandatory, so that the sneaking feature has a real purpose. This addition would make the game more varied, therefore adding variety as well as improving the context factor of game feel. Something that would further improve game feel, more specifically its polish faction, are particle effects. They would add “life” to the game.

In order to make the game more interesting of an experience, the player character's movement should be improved. It could have some unique maneuvers or at least different emotes it can perform with different cats, such as touching noses or circling around the other cat. This would serve to the core feature of engaging movement as well as variety. The game could also have special movements such as climbing vertical wooden surfaces or clearing obstacles by pouncing longer distances than the normal "jump" ability allows.

Another point that would add to the game's variety would be to add more collectables, ones that can be collected if the player feels like they want to. These could be, for instance, finding small animals within levels and greeting them as well, adding into another counter. Another collectable could be finding one of a few unique plants that could only be found inside a certain level. These could be recognized by interacting with the plants and would look different from regular plants.

If there were to be a hub world in Cat Game, it would in all likelihood be the starting level, as that is where the cat's home is. There might, for example, be a location there that allows the player to return to previous levels by "recalling" the memory of a specific level, thus reliving (replaying) it.

A reasonable ending to the game would perhaps be to return to the starting farm from level 1 after the last level. This would further serve to the first level working as the hub world. Alternatively, the player character could think something along the lines of "It may be time to return home" once they complete the last level, after which the end credits may appear on the screen.

References

Boulton, N. (2018, October 20). *What Makes a Good 3D Platformer?* [Video]. YouTube. https://www.youtube.com/watch?v=SGq5Zaygl-g

Bycer, J. (2019). *Game design Deep Dive: Platformers.* CRC Press.

CrashBash2000 (n.d.). Ripper Roo (Boss Fights). Bandipedia. Retrieved October 16, 2024, from https://crashbandicoot.fandom.com/wiki/Ripper_Roo_(Boss_Fights)?file=Ripper_Roo_Remastered_%28Crash_Bandicoot%29.png

Dark52 (2020). Chapter 1: Artefact Piece 9/10. Darkbricks. https://www.darkbricks.net/indiana-jones/the-original-adventures/walkthrough/raiders

Everman, T. (2021, August 24). *How Hub Worlds Shape Video Game Design.* WIRED. https://www.wired.com/story/how-hub-worlds-shape-video-game-design/

Francis, B. (2021, July 22). *A taxonomy of Weenies: the landmarks that define Ghost of Tsushima.* Game Developer. https://www.gamedeveloper.com/design/a-taxonomy-of-weenies-the-landmarks-that-define-i-ghost-of-tsushima-i-

Galuzin, A. (2018, August 3). *Map Creation Guidelines.* World of Level Design. https://80.lv/articles/introduction-to-level-design-for-games/

Game Developer Staff (2015, May 13). *6 musts for a perfect platformer, from the _Yooka-Laylee_ team.* Game Developer. https://www.gamedeveloper.com/design/6-musts-for-a-perfect-platformer-from-the-i-yooka-laylee-i-team

Game Maker's Toolkit. (2015, February 17). *Secrets of game feel and juice* [Video]. YouTube. https://www.youtube.com/watch?v=216_5nu4aVQ

Hacktic. (2023, February 25). *This is why your favorite 3D Platformers are so fun* [Video]. YouTube. https://www.youtube.com/watch?v=wR0YWx1bPHQ

Hovermale, C. (2018, April 24). *Home is where the hub world is.* Destructoid. https://www.destructoid.com/home-is-where-the-hub-world-is/

Jyväskylän ammattikorkeakoulu. (n.a.). *Research-based development assignment.* https://help.jamk.fi/opinnaytetyo/en/thesis-implementation-methods-and-results/research-based-development-assignment/

Lantz, P. (2023, Nov 27). *3D Platformer Level Design Process.* LinkedIn. https://www.linkedin.com/pulse/3d-platformer-level-design-process-peter-lantz-ztjsc

Merriam-Webster. (n.d.). Hub. In *Merriam-Webster.com Dictionary*. Retrieved November 22, 2024, from https://www.merriam-webster.com/dictionary/hub#:~:text=%3A%20a%20center%20of%20activity%20%3A%20focal,routes%20most%20of%20its%20traffic

Nintendo (n.d.). A Hat in Time [Screenshot]. Nintendo. Retrieved October 16, 2024, from https://ec.nintendo.com/NZ/en/titles/70010000003812

Nintendo (n.d.). Kao the Kangaroo [Screenshot]. Nintendo. Retrieved October 16, 2024, from https://www.nintendo.com/us/store/products/kao-the-kangaroo-switch/

Orlong, K. (2017, September 8). Agent Hugo: Lemoon Twist (Windows) screenshot: Crab [Screenshot]. MobyGames. https://www.mobygames.com/game/81311/agent-hugo-lemoon-twist/screenshots/windows/902779/

Pandaboy78 (n.d.). Tribalstack Tropics [Screenshot]. Yookapedia. Retrieved October 16, 2024, from https://yooka-laylee.fandom.com/wiki/Tribalstack_Tropics

Sinclair, N. (2020, May 8). *Your guide to 3d Platformer game design (Clear explanations).* Career Gamers. https://careergamers.com/your-guide-to-3d-platformer-game-design/

Stewart, S. (2013, November 19). *Top ten: Platforming Game Mechanics - Nintendojo.* Nintendojo. https://www.nintendojo.com/features/columns/top-ten/top-ten-platforming-game-mechanics

Stuart, K (2017, April 8). *The strange, timeless appeal of early 3D platformers.* Eurogamer: https://www.eurogamer.net/the-strange-endless-appeal-of-3d-platformers

Swink, S. (2007, November 23). *Game Feel: The Secret Ingredient.* Game Developer. https://www.gamedeveloper.com/design/game-feel-the-secret-ingredient

TV Tropes (n.d.). *Hub Level.* TV Tropes. https://tvtropes.org/pmwiki/pmwiki.php/Main/HubLevel

Video Games Uncovered (n.d.). *Crash Bandicoot 2 Coloured Gem Location Guide* [Screenshot]. Videogamesuncovered. https://videogamesuncovered.com/wiki-guides/crash-bandicoot-n-sane-trilogy/crash-bandicoot-2-coloured-gem-guide/

Wikipedia (n.d.). *Research-based design.* Retrieved December 4, 2024, from https://en.wikipedia.org/wiki/Research-based_design