

Firebreath Forest

Interim Demo

Johannes Madest

Mingyang Li

Bassant Elnaggar

Task Progress

Functional Minimum

Excitement Minigame: The Basic Gameplay of the Minigame focused on driving away with gold using a hand trolley is close to being complete, including basic controls, scoring rules depending on both the distance to the chasing dragon and rate of success in completing the quicktime events as well as increasing difficulty in the course of a single playthrough. While the development of the accompanying scenery the player will traverse throughout their escape is well underway, a problem that arose is to find the proper scaling of the intended path to both have a smooth ride while maintaining good visibility but also considering the expected high amounts of speed the player is supposed to have in order to invoke the feeling of excitement. Our next steps will include filling out the scenery and UI with fitting assets as well as implementing the actual dragon chasing you visually.

Anticipation Minigame: For the minigame focused on stealing gold from a dragon's nest, the functionalities for movement, collecting gold from various tiers of storages and getting spotted by the dragon are completed. Still work in progress is implementing the states of awareness of the dragon, influenced by the actions of the player such as making sound by moving quickly or interacting with gold storages. Here, the big design challenge we are working on is finding fitting indications of how close the dragon is to waking up that can be easily understood by the player. Similarly to the Excitement Minigame,

we are still working on designing the level, in which the player should be presented with a good balance of targets of varying risks, opportunities to hide and observe the dragons behavior, all while ensuring that the player always has a clear understanding of how and when they could be at risk of detection.

Low Target

Building roller coaster level: Unfortunately, the development of the rollercoaster level has encountered delays due to the extensive time and effort invested in implementing various approaches for the rollercoaster's movement along the tracks and simulating certain physical effects (as mentioned in the Implementation section). As of now, the core functionality has been completed and integrated into our excitement minigame. However, the construction of the open forest environment in the main scene is behind schedule and will be prioritized to ensure timely completion.

Regarding additional physical simulations such as acceleration and deceleration during lifts and loops, we understand their importance in enhancing the user experience. While these features are not currently functional, we aim to include them in our desired target, as they have a significant impact on the overall user experience.

Implementation

Excitement Minigame:

In this minigame, the player will be prompted with a stream of quicktime events, each consisting of a varying number of randomly generated buttons to press in order. The amount of buttons as well as the pool of buttons the randomizer will draw from will vary depending on the current state of difficulty. On a successful round of button presses, the player will be awarded with a boost of speed corresponding to how fast this round was completed. If the player enters the prompted buttons incorrectly, that round will be considered failed, resulting in a decrease in speed. If the player does not complete the button sequence in the required time, no further penalty will be applied, yet the minecart will continuously lose momentum over time. Once a round is completed in either way, a new button sequence will be generated. This cycle will continue until either the goal is reached or the

player is caught by the dragon. The player will start the game with a set amount of gold. Over the course of the minigame, a certain amount of gold will be lost periodically depending on how close the dragon is.

Anticipation Minigame:

The controls of the player include sprinting, sneaking, crouching as well as interacting with objects, each of them being assigned a noise level that affects the state of the sleeping dragon. The game will feature multiple tiers of gold deposits where a higher tier indicates that more gold can be gathered here, yet interacting with them will create more noise. A player can interact with these deposits to begin collecting gold from them, which will take a certain amount of time depending on the specific deposit, yet can be interrupted at any time. The specifics on the dragon's Ai are still work in progress and will be explained in detail in the alpha release chapter.

Roller Coaster and Railroad Tracks

To enable the rollercoaster to travel along the railway track, three approaches were proposed and tested:

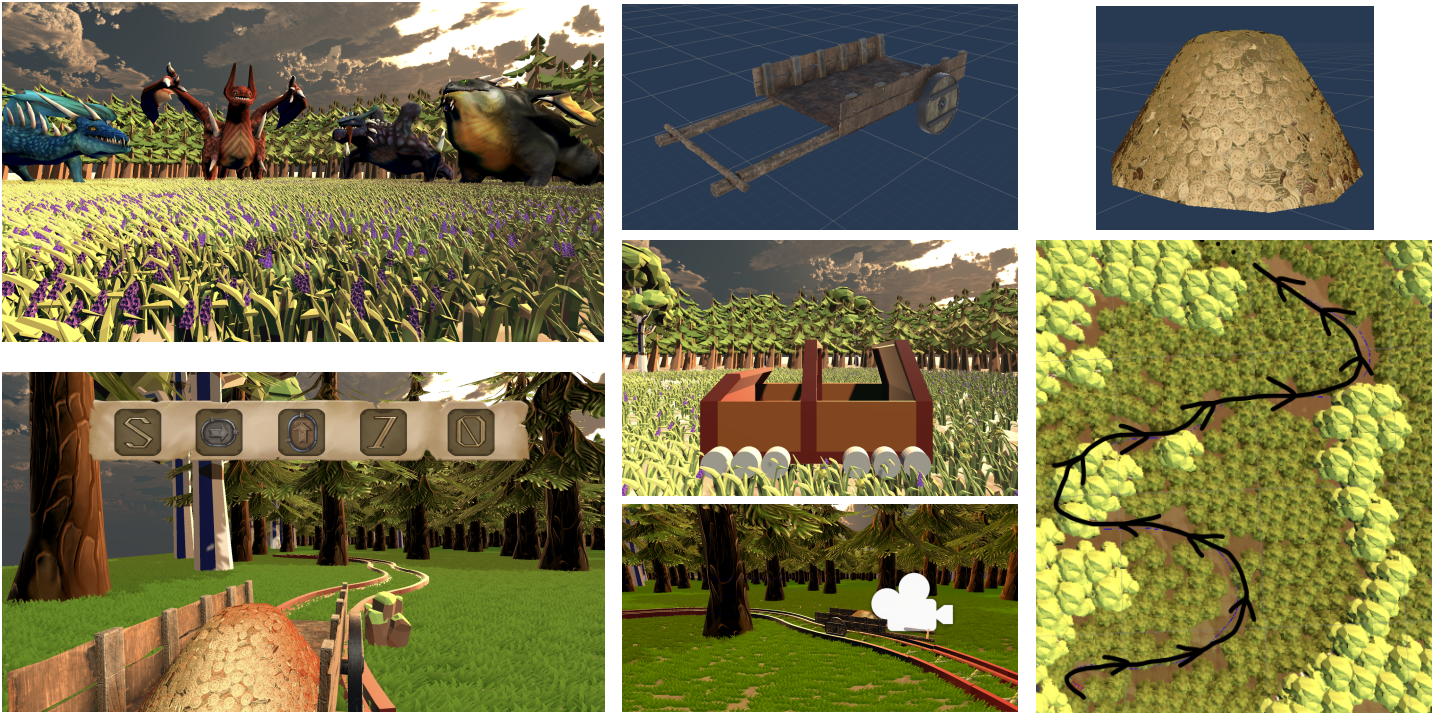
1. **Navigation Mesh:** The initial idea was to create a navigation mesh for each section of the railway track, allowing the rollercoaster to find its path autonomously. However, this approach presented some challenges. The track pieces were too small to bake a mesh, even with a small agent radius setting. This limitation resulted in reduced building efficiency. Additionally, implementing a navigation mesh made it difficult to incorporate vertical loops, which are crucial for providing an exhilarating experience, especially in a first-person view.
2. Another approach is to have the rollercoaster travel along the track piece by piece by connecting the vertices of two adjacent tracks. While this method works well with simple prefabs like capsules, difficulties arise when applying it to our track mesh. Since we are not proficient in using Blender, we imported a track model from the internet and made only minor adaptations. However, due to the complexity of the track mesh, we decided to abandon this approach to avoid wasting time.

3. Spline:After careful consideration, we have decided to utilize the Spline Packager provided by Unity. This choice offers the advantage of convenience: we simply need to draw the desired route using the spline tool, and the rollercoaster will automatically follow it. This approach saves us time by eliminating the need to connect track pieces individually. However, it is important to acknowledge that it is a somewhat artificial method, as the rollercoaster follows the spline rather than our actual track. To create a convincing illusion of the rollercoaster running on rails, we will need to ensure a high degree of overlap between the spline and our track structure.

After evaluating the limited functionality provided by the Spline package, we realized that our game requires more control over the rollercoaster's speed, position, and movement posture during gameplay. As a result, we made the decision to customize and modify the Spline code to better suit our needs. Additionally, we incorporated a rigid body component to the rollercoaster to enhance the physical simulation. This final solution combines the Spline functionality with the added realism of a rigid body.

However, it's important to note that this part of the implementation, which involves the physical simulation and integration of the Spline and rigid body components, is still a work in progress. We plan to introduce and discuss it in more detail during the upcoming milestone.

Designs and UI



For the level design and UI, we started by collecting the assets that match with the theme of the whole game. Also, we started with building up two scenes, one for the anticipation game and the other for the excitement game. In the excitement mini-game scene (Figure on the bottom left), we used a terrain to make it easier to adjust the forest details. The scene is initially built to have a track within the forest, and the cart of gold goes along that path. The cart speed is controlled by the player by entering the correct inputs that are shown on the screen fast in order to escape by the gold, and the cart itself follows the path by following the spline identified in the scene, similar to the drawing shown in the figure on the right. The excitement mini-game still needs more level designing, which will be achieved in the following milestones. We also started working on the level design for the anticipation mini-game, which is initially an opened field within the forest and as we mentioned in the game proposal that the player will be moving around searching and collecting for the gold while avoiding gaining the attention of the dragon. Again, we used terrain to build the environment and paint the initial assets in the forest, along with collecting different forms of the dragons that will be in the game play. The scene still needs to have forest

assets along with the randomly placed gold for the player to search for. For the UIs, we are following the theme of the old adventures, as you can see in the figures below, as it matches with the idea of our game.



Future Plans

Our goal of invoking specific emotions with our minigame can only be properly tested once all parts have been fully implemented. Once that is complete, we will focus on testing our minigames with that goal in mind, intending to add complexity or making adjustments to gameplay to further improve the feeling we wish to invoke. In particular, we will focus on improving our minigames in such a way that a player remains engaged and immersed in the intended emotions even after multiple playthroughs.

In its current state, the excitement minigame might not offer enough complexity to keep engaging the player for the intended duration of the minigame. To tackle this problem, we intend to add complexity in the form of events that happen along the path in which the player has to make quick decisions that will effect the remainder of the minigame in terms of difficulty, required button prompts in the quicktime events or chance for extra rewards for completing special tasks.

This issue has also been addressed in feedback we have received. One point that stood out is the idea of designing the minigames to be very challenging with the intention of replayability. Especially when considering the aim of achieving high scores, we will definitely consider designing our minigames in such a way that a player will have difficulties in their first playthrough, only being able to master the game mechanics and achieve high scores over the course of multiple attempts.