

Milestone 3: Alpha Release

Fall For Me!

Team Cicisoft

Andrea Solanas de Vicente

Ankur Deria

Michael Dey

Bendegúz Timár

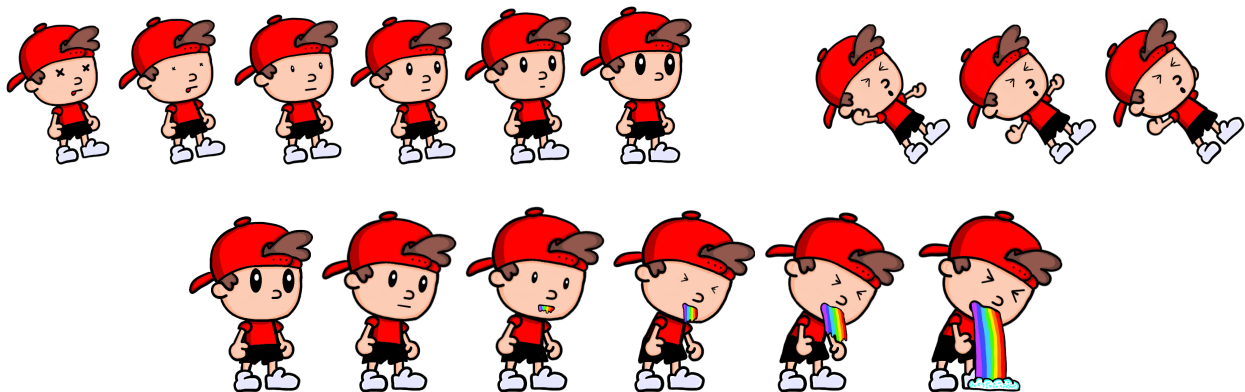
Task progress

Our goal for the alpha release was to produce a finished working version of our game that implements the main gameplay and some extra features. In order to reach this goal, we first wanted to add some UI elements like menus, and character animations, also we added some form of interaction for the characters so they could choose their players and pick the mode of the game. Finally we added the multiplayer functionality so the game could be played as 2 versus 2.

Animations

New character animations

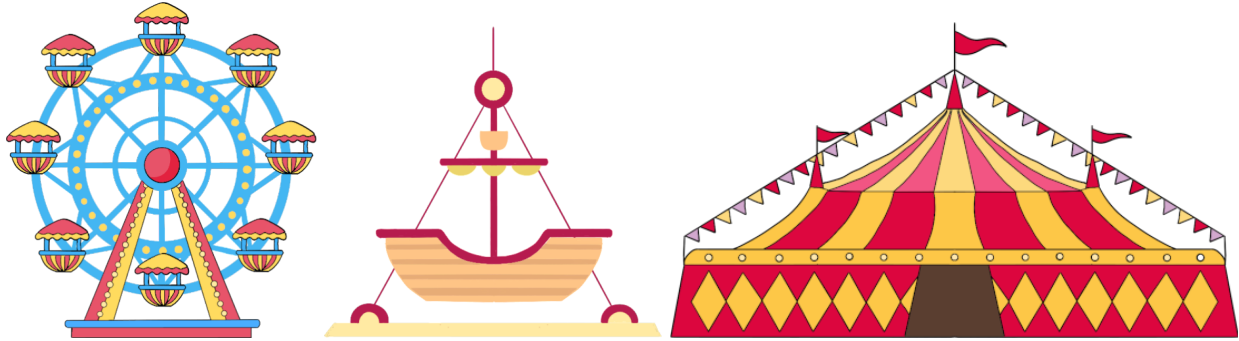
We added a few new character animations: falling, getting hit and vomiting. Below are some of the frames for the main characters.



We also created 2 brand new characters and designed the same animations for them. The animations for all the characters are very similar, with the exception that one set of characters wears a cap and the others don't. This way the players know which team they are on and their characters are easy to tell apart. The new players are shown further in the document.

Background changed

For the background we wanted to add some elements that would represent the environment of an amusement park, since for the previous demo we went with a very simple background. For this milestone we wanted to make sure that the players knew or could see that they were in an amusement park. To accomplish this we created amusement park rides with their own animations to set them in the background. Some examples: a ferris wheel that turns in circles, a circus tent with moving flags or a ship that sways back and forth.



The background now looks more vibrant and it makes the player feel as if they were on a roller coaster in an amusement park. In the following image, it can be seen how the background looks now. In the future we would like to make the animations more fluid and add more frames to make them better.

Candy animations and effects

The candy collides with players, bases and with other collectibles. We added a particle system to make the candy more visible for the players. Whenever candy is hit by a player, a fireworks particle is activated with other post processing effects. When it is hit by other objects, an explosion particle is activated and the candy disappears. We added more types of movement to the candy like it would follow along the track while doing a sine wave motion in the y axis. Similarly there is a cosine wave motion, a loop motion, etc.



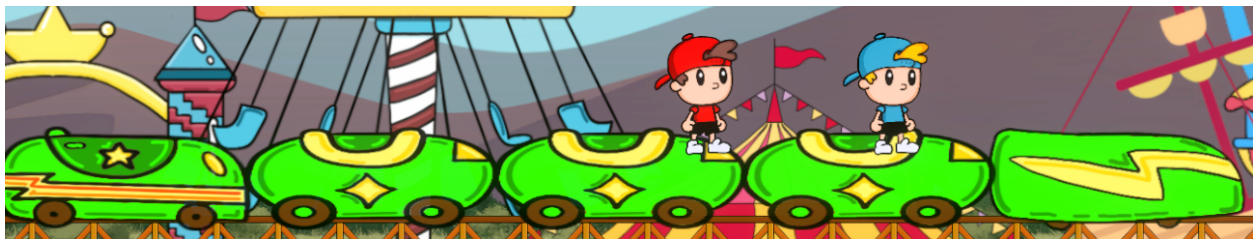
Balloons popping

For the bases which will be explained in more depth later, we added a set of balloons that symbolize how much time the bases will be available for the characters. There will be a number of balloons that will slowly pop and when the last balloon pops the base will fall out of the screen. For this we created the animations for the two teams' bases and their balloons popping.



Roller coaster sprite

For the previous milestone we didn't have the final icon for the roller coaster yet. But for this milestone we were able to create three different icons for the roller coaster to use as the front wagon, the ones in the middle and the last wagon. We also made two different styles (orange and green) so it could be randomized at the beginning of the game and the roller coaster was a different color for each time.



Visual effects

Mainly particle systems, and post processing effects like depth of field, lens distortion, chromatic aberration, etc. We also use timeScale to slow down for a few seconds to give a cinematic effect. 3rd party assets from Unity Asset Store were used for the particle systems, whereas some were made by us.

Music and sound effects

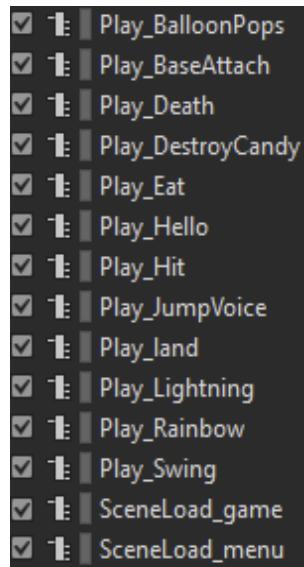
Music

In this milestone we finished the main gameplay background music. It starts with an instrumental part, which we will use in a loop for the controls tutorial at the beginning of the game. At the first chorus the song changes to a kind of EDM or electro dance pop style, and the later verses stay electronic with a very similar melody as the first instrumental part. This part will be looped for the remainder of each round. The whole song is about 3 minutes long, which should be enough considering the rounds with a time limit will last 5 minutes according to plans.

We also started working on the main menu music. This song will be much shorter and with a consistent electro pop style. This song is in its very early phases, but we suspect about half a minute will be long enough.

For composing the music we used FL Studio, with which one of the team members already had experience. Even though composing the main gameplay music took long, we plan that finishing the main menu's theme will take a much shorter amount of time, given the experience we got while working on the gameplay music.

Sound Design



The sound is designed to be fun and happy like the artstyle. The Rollercoasters sound depends on the velocity of the cars. There is a pulling sound, when the minimum speed is achieved and cheering at the first drop after each pulling up.

Before the new visual effects were applied to the candy the eating sound was someone biting an apple. We changed that to bubbles bursting and bubbling, to reflect the visual effect.

On respawn the visual effect will be a lightning strike. The sound for this is already in the game.

The rainbow sound is already finished, but it doesn't start properly when run from the code. It is inspired by this [video](#).

One sound that is still missing is footsteps.

Bases

The original design of the bases proved to be too rigid for such a vibrant environment. They had a static world position, which also made accessing and leaving the bases way too hard, since the players barely had any time to spend on the base's surface if they wanted to reach the roller coaster.

We came up with a new design, where the bases are hung on balloons. Using balloons allowed the idea that the bases can follow the roller coaster for a short amount of time. The base floats in with a number of balloons attached to it. The balloons pop after some time one after the other, and intuitively fall down after all the balloons popped. This makes reading the amount of time the players can spend on the bases really easy, and also adds to the overall vibe and atmosphere of the game. Players can now respawn at their respective bases after they fall off the rollercoaster as well.



Menus

For designing the UI of the menus we decided to work with the new UI system of Unity called UI Toolkit. The UI Toolkit ditches the old GameObject-based UI of uGUI (Unity's old UI system), where elements of the UI are put on a Canvas Object. Instead, it organizes UI layouts into so-called .uxml files, which are simple .xml files and follow the principles of web UI design, specifically the FlexBox structure. Attributes of layout elements can be collected in .uss files, which work similarly to .css files. The user needs to define handles to the different layout elements. The names of the handle only have semantic meaning to the user, meaning a button handle can be put on a label or a panel too.

Separating the layout and the logic of the UI is considered good practice in every front-end project, as it enables building and testing the two areas in parallel. Not to mention that the programmer doesn't have to redo the whole UI after each change made by the artist. However, composing a web-based UI has its limitations, as flex attributes affect one another. This means that simple UIs can be achieved with relative ease, but more complex structures can be and were a head-scratcher for us. The bigger problem however was the user intractability of this new system. Even though the UI Toolkit has been out for more than 2 years now, tutorials on the subject, especially controlling with multiple gamepads, are not of the highest quality, and the documentation of the Toolkit can be very confusing. Since the layout and the logic is separated, working with the layout elements is a bit harder on the logic side. Also, the new UI Toolkit has a built-in input handling system, but it is still not the most usable for gamepad inputs.

Since we were losing a lot of time on the subject, we decided to make the new menus for this milestone with the old uGUI system. The gameplay UI still uses the new UI Toolkit. In all of the menus, the background is blurred out but still playing the gameplay's background. This adds a welcome vibrancy to the menus, and makes starting the game more seamless.

Main menu

The main menu has a very simple layout, consisting of 3 buttons - Start, Options and Quit. Intuitively, the options button leads to the options menu, while the quit button exits the application. At the moment we do not have any settings in the options menu and we plan to add

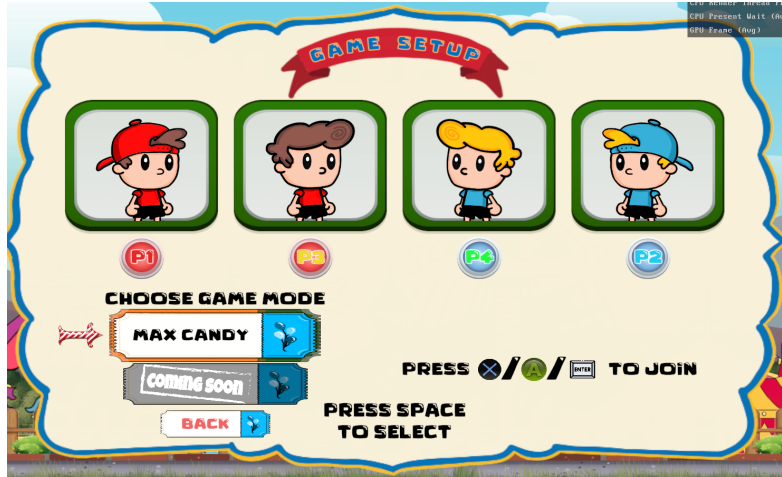
this during the next milestone so the players can modify the settings in the game. The start button leads to choosing the characters and the game setup screen.



Game setup menu

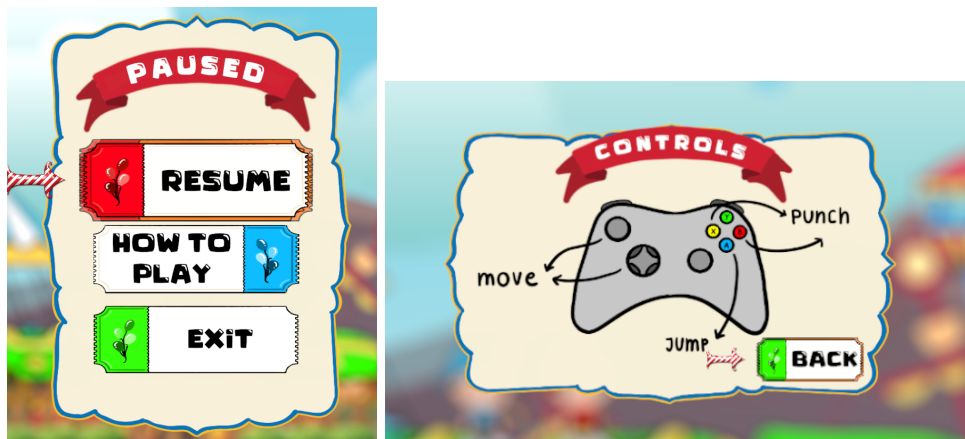
The players can get to this menu by clicking the Start button in the Main menu. On this screen the players can choose their character and the gamemode of the game. The characters only have cosmetic differences, and each handles in the exact same way. We currently plan two different game modes: a time-limited mode which runs until 5 minutes, and the team who collected and deposited the most candies in this interval, wins; and a collector mode, where the teams aim to collect a number of candies, and who first deposits this amount, wins. The time mode has not been implemented yet but we plan to have it for the next milestone and have it during the play testing for the users to try it out.

For this menu we planned to include a cursor for each player, similar as to that in the Super Smash Bros. character selection screen. On connecting a controller a cursor for the new player would be spawned, and disconnecting the controller would remove the appropriate cursor. The player could hover their cursor over their selected character, and click to select it. This solution would efficiently show the number of active players, and would make navigating this menu more fun. However, this is where we decided to roll back to the old uGUI, since achieving this behavior with the UI Toolkit proved to be unnecessarily overcomplicated. While controlling a cursor in the uGUI is flawlessly working, spawning in the cursor is still unstable, and therefore we removed it from this release. The current solution uses only a navigation flow between the elements of the UI, where the players can select the different choices similar to the Main menu.



Pause menu

To access this menu players can click Esc (keyboard) or select (controller) during the game and pause the gameplay. This gives the players control to take a break from the game. In the pause menu there are three buttons Resume, How to play and Exit. If a player clicks on resume, the game will resume from where it left off when the players paused. If the player clicks on How to play, the controls for the game will show to inform the players how to play. If the Exit button is selected the gameplay will end and the main menu will show like at the start of the game.



Tutorial menu

For playtesting we added a simple tutorial for the players to get the hang of the game and its controls. The tutorial menu will show when the player is playing the game for the first time. If the gameplay is restarted it will not be shown anymore.

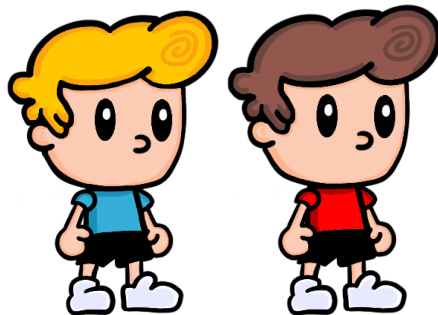
The menu is very simple and it only consists of a few pages with the basics of the game, how to play, how to win and the controls among others.



Multiplayer Functionality

New players

We created 2 new players for the two teams, with the same design except for the caps. The new characters have their own candy counter in the gameplay UI but they have the same controls as the other characters. We did the animations fairly similarly so that the players could switch back and forth from the different characters easily.



How it works

Players can join in the game setup menu. They need to press a button to join, 1 player per device. They can choose any of the characters they want (if its not already chosen). The players get carried over to the gameplay scene after that. Each device is bound to the player and that player character only accepts input from that device and ignores all others.

Winning conditions

As previously mentioned we plan to add two different winning conditions to the game: either reaching a specific amount of points first, or getting the most points in a given time. For this release we only implemented the former: the players need to collect and deposit 50 candies in total, at which point they win the game.

Implementing a time-based gameplay would require some more logic, such as designing and implementing a clock in the main gameplay UI, and tackling scenarios where both teams have

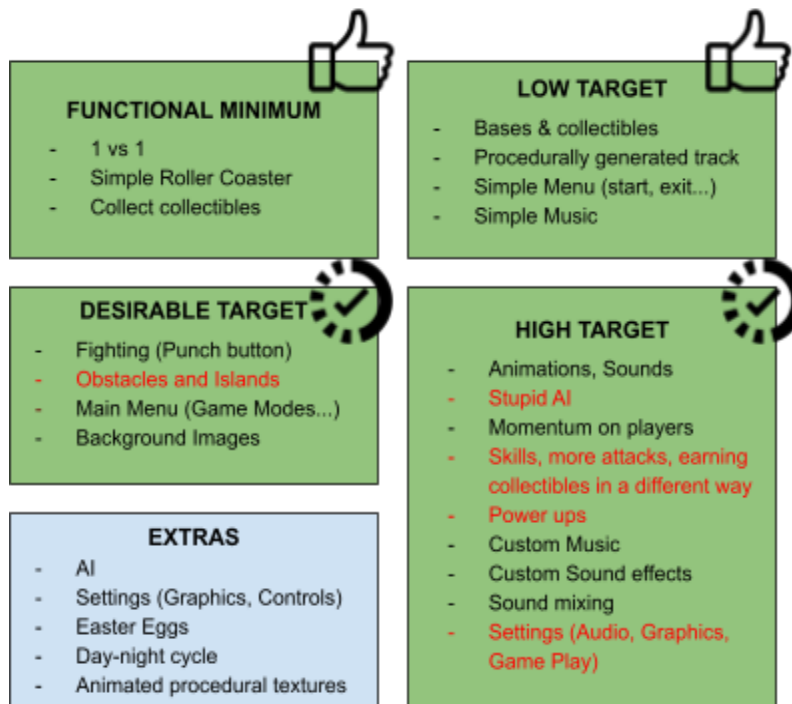
the same amount of points when the timer runs out. Because of this, we decided to include this condition in a later release.



After having it implemented, the players can choose between the different game modes in the Game Setup menu. In the future, we also may include other game modes, or a custom mode where the player can enter their own time- or point-limit.

Layers

In comparison with the previous milestone we have been able to achieve the low target in its entirety plus doing most of the desirable target. The only thing we were not able to achieve was the obstacles. We plan to add them for playtesting so the users can test them and give us their feedback. We were also able to achieve some of the goals from the high target and plan to achieve some more which is explained in future work.



Problems

Meta files in gitignore

To avoid the recurring merge conflicts in hard-to-decipher scene files we introduced more prefabs. One of our team members had included Unity meta files in gitignore. Meta files store the hash ID's of all objects in Unity and as a result the new prefabs weren't synced well over git.

Problems between using both old and new UI

Old UI has a vast amount of documentation compared to the new UI but the new UI offers much better development with html and css like framework. It has native support for a lot of features that make scaling for different aspect ratios a walk in the park. But due to the lack of much documentation we faced a lot of problems with it and eventually decided to fall back to the old UI system.

There were some problems with mixing both of the UI types since the text fonts and some of the images were messed up when using both at the same time. We had to fix this problem multiple times which became very tedious.

Choosing characters and going into gameplay

We had some trouble carrying over the spawned player to another scene without copying all the classes in it. This was necessary when choosing the characters in the setup menu and creating the gameplay scene altogether.

Integrating Wwise into Unity

There was an unexpected Error, when opening Unity after integrating Wwise. This Error also happened, when retrying the integration with a fresh Unity project. It seems to be connected to recent changes in unity because there were no discussions about it on the internet. We were able to solve the problem on our own though.

Future work

As we mentioned we plan to add a few ideas that we were not able to include in this release, for example the game mode of using time as a winning condition and maybe obstacles for the players. The animations will also be worked on to make them more fluid.

We want to add some more effects for the characters so they really stand out. We also have as part of our high target making an AI so players can play the game as 1 or as 3 and play with or against the AI. Another thing we wanted to add is for the player to be able to modify the settings of the game.

If we have enough time we would love to include an easter egg or maybe a day and night cycle for the gameplay.

Work log

Week	Goal	Tasks	Assigned to	Hours expected	Actual hours
Week 8: 29.05 - 04.06	Desirable Target mechanics	Fighting	Ankur, Bende	5-6	6
		Obstacles	Ankur, Andrea	10-15	Not done
		Add sound effects	Mischa	8	8
Week 9: 05.06 - 11.06	Main menu	Artwork and different game modes	Andrea	5	10
	Sound	Add Music	Mischa, Bende	2	4
Week 10: 12.06 - 18.06	Background artwork	Finish other artwork missing	Andrea	10	15
	Sound	Mix Sound	Mischa	3	3
	Milestone 3 documentation	Presentation	All	5	4
		Document	All	4	4