Emotional Weather

An integrated real-time rendered weather simulator based on unreal engine and Media Research

Individual Work

Development Time: 2 Months
Duration: 5-10 Mins

Emotional Weather

Contents

Research And Idea Initiation

Media Text Research

Art and Technology

Design

Project Design

Technical Design

Project Production

Unreal Engine Implement

Art and Production

Iteration

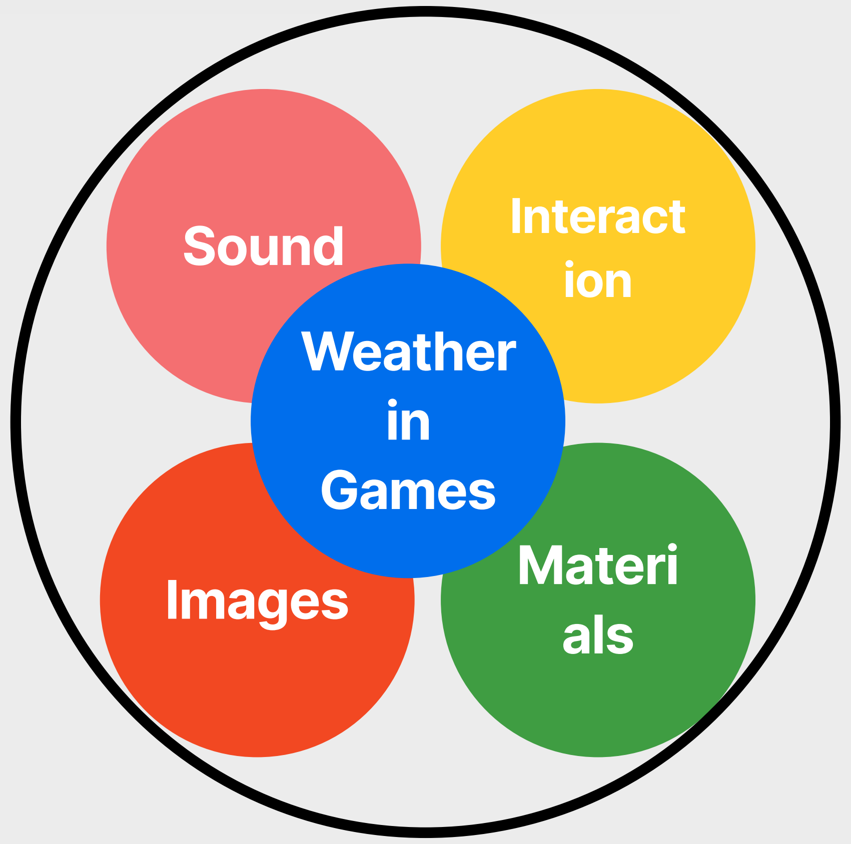
Reflection

Further Development

Project Preview

Research & Idea Initiation

Media and Semiotics Research: How Weather in games influence User Experience

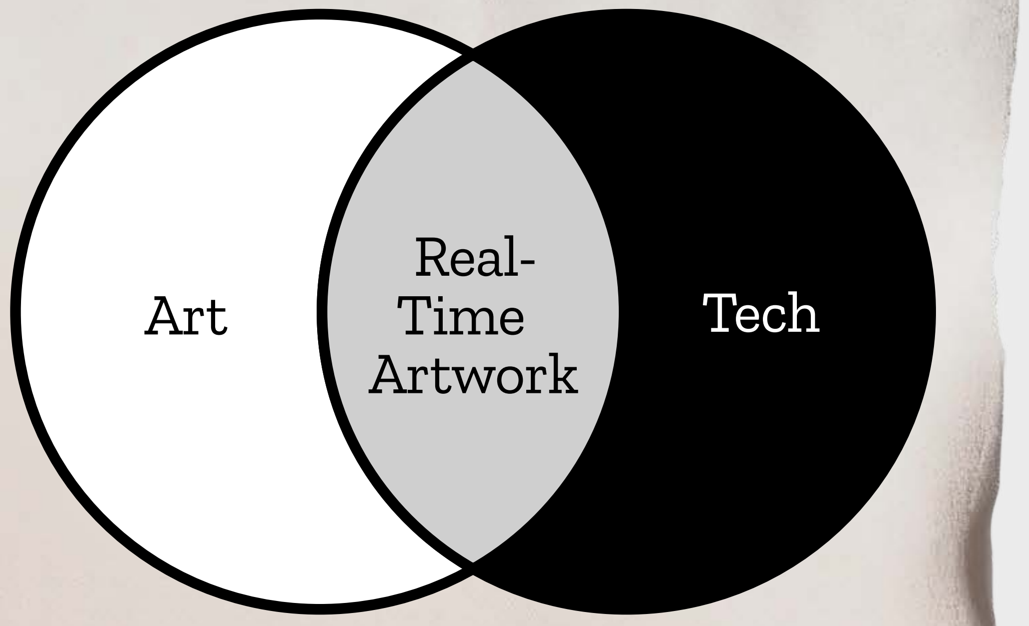


Media Semiotics Analysis

Multimodal analysis is a means of analysing a media text by examining the sounds, images, materials, etc., of the media text to reveal the intended meaning behind the text (Burn,2016).Using this framework to investigate the connection between weather systems and audience emotions, from static paintings to dynamic games.
The research is also based on other theories, such as Ryan's reader-response theory (Ryan,2003), to further reveals the emotional metaphors behind the weather system.

According to Andrew Burn's multimodal theory, games act as a mediated text that conveys information and content through sound, interaction, images and materials, and emotions are no exception.

The theory of multimodality explains how meanings and metaphors in games are realised through a technical approach to visual effects and mechanics.
This further provoked me to think about the role of technology and art: through the application of the theory, could we render complex landscapes through real-time rendering techniques and create new art by performing artistic manipulations, and along with these visions, and through a number of case studies, I produced this project



In the case of Red Dead Redemption II, the effects brought about by rain in the game include:

To Help the player immerse themselves and build an emotion connection in the virtual weather system by trying to keep things simple in terms of sound effects and removing unnecessary sound effects like background music and wind sound effect.

Sound

Triggered Sound: Players can hear the sound of raindrops falling on different surfaces and ambient sound effects.
Ambient Sound: Background Sound of raining and sound of thunder

Interaction

Game Mechanics:When it rains in a game, the character's movement speed is affected, and this negative impact on the game character also affects the player's emotional experience.

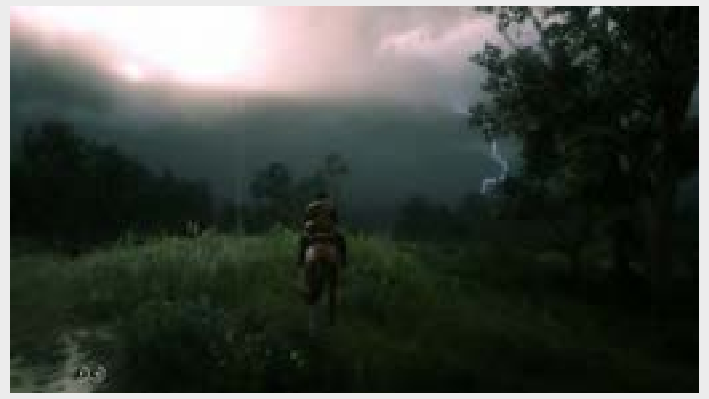
Images

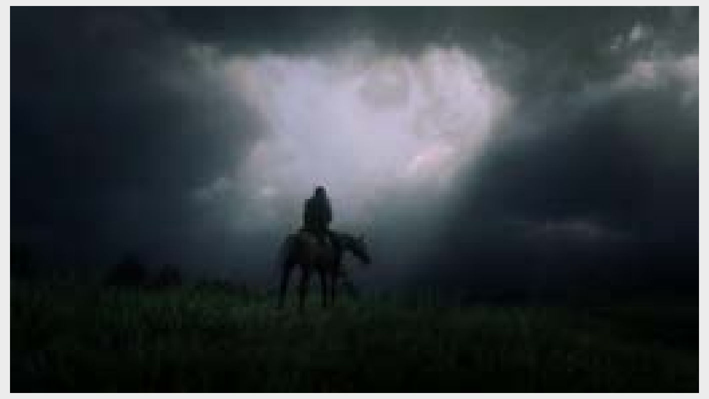
Environments: Puddles of water form on the ground in the game world, and mountains in the distance are shrouded in mist.
Colour: Colours become grey and deep.

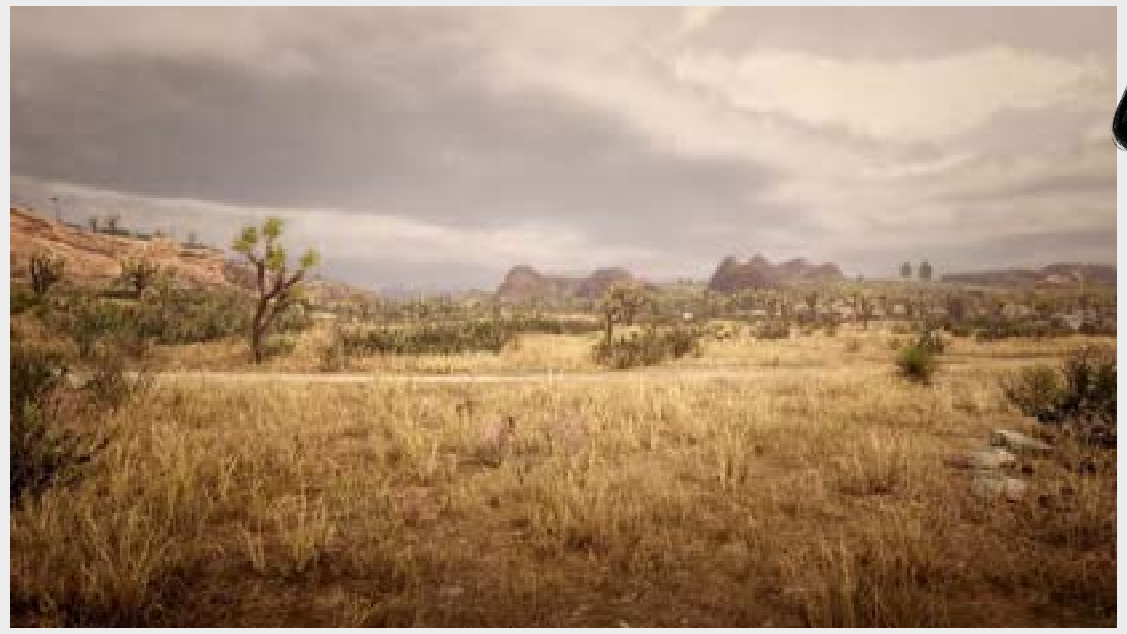
Materials

Wet materials: Materials in the environment become wet and there is a lot of reflection and refraction of light.

Together, these visual effects and colour changes and musical effects create a hazy, melancholic atmosphere. Bringing the player into a tense, sad mood.







Emotional Weather

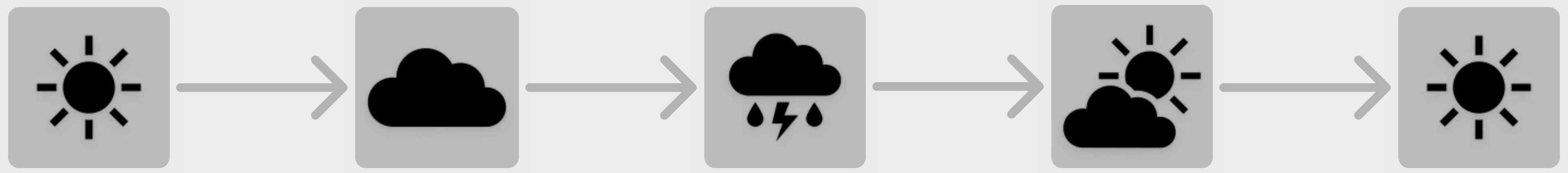
Design

Based on the textual research, I wish I could make a weather simulator that is **vivid, detailed**, with **artistic inspiration** and **emotional connection.**

I raised some question when I start design the system.

What Weather should be included?

There are a lot of different weather in the real world, in this system, I try to integrate the weather that user are familiar with and easy to present in digital world.
Therefore I have choose **sunny, cloudy and rainy** weather for this project.



However, the "weather" is not only the weather listed above, by adding a seamless transition, it also include weathers like "After Rain" "Before Storm", these weather also resonate with users.

Apart from weather, what else?

Time and the sun are also very important parts of the weather system.
Not only because by controlling the time in the weather system, more artistic effects can be created, such as 'God Light' (Tyndall effect), sunrise, and so on.
But also because "time" is also an element that affect and influence user's emotion: the morning time may bring a more energetic feeling, and the evening time may bring a sad and depress feeling.
Having a time controller in the system together with the weather controller have enhance the influence on user emotion.

Which Engine should I use?

Work with blueprint and custom nodes,Better rendering result.

Natively support HLSL to build shaders, development efficiency may be better.

For better visual effect, not using Unity engine but using UE engine, but Unity natively support HLSL, development efficiency may be better.

Because UE engine is better in real time rendering and more realistic. And for a simulation system that affects the user's mood by rendering the artistic experience in real time, rendering effect and visual presentation should be the first priority.

Present Real-time Rendered Art

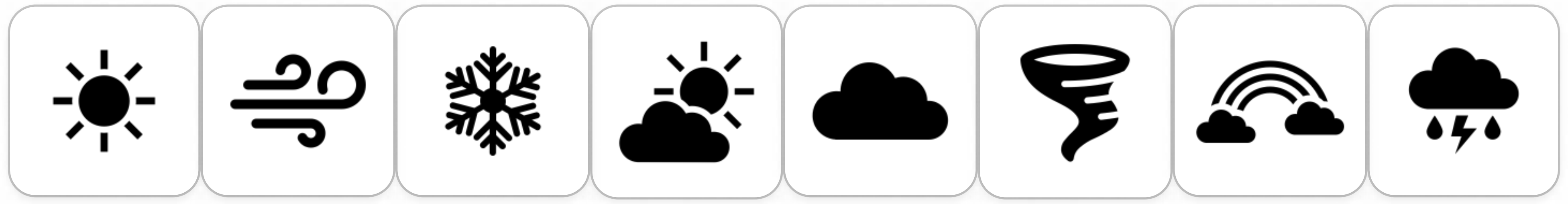
Code is neutral and cold, but the results rendered through it can have the power to affect emotions.
The aim of this project is to explore the intersections and boundaries between new and existing art produced by the artistic processing of real-time rendered landscapes through real-time rendering techniques.
The project generates new art by rendering real-time rendered landscapes onto canvas and then merging them with existing art. At the same time, this artistic tool has other roles: in the industry it could be part of a weather system for a game production, or it could be a tool for anthropological research.

Design

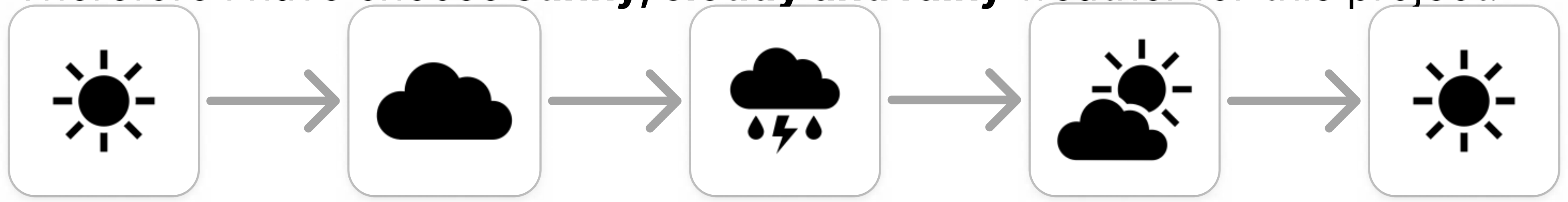
Based on the textual research, I wish I could make a weather simulator that is **vivid**, **detailed**, with **artistic inspiration** and **emotional connection**.

I raised some question when I start design the system.

What Weather should be included?



There are a lot of different weather in the real world, in this system, I try to integrate the weather that user are familiar with and easy to present in digital world.
Therefore I have choose **sunny, cloudy and rainy** weather for this project.



However, the "weather" is not only the weather listed above, by adding a seamless transition, it also include weathers like "After Rain" "Before Storm", these weather also resonate with users.

Apart from weather, what else?

Time and the sun are also very important parts of the weather system.
Not only because by controlling the time in the weather system, more artistic effects can be created, such as 'God Light' (Tyndall effect), sunrise, and so on.
But also because "time" is also an element that affect and influence user's emotion: the morning time may bring a more energetic feeling, and the evening time may bring a sad and depress feeling.
Having a time controller in the system together with the weather controller have enhance the influence on user emotion.

Which Engine should I use?

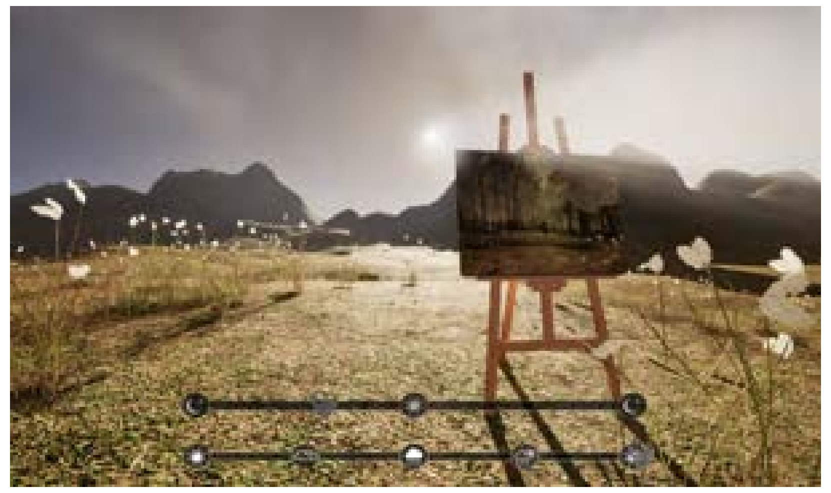
Natively support HLSL to build shaders, development efficiency may be better.

Work with blueprint and custom nodes,Better rendering result.

For better visual effect, not using Unity engine but using UE engine, but Unity natively support HLSL, development efficiency may be better. Because UE engine is better in real time rendering and more realistic. And for a simulation system that affects the user's mood by rendering the artistic experience in real time, rendering effect and visual presentation should be the first priority.

How to present real-time rendered art?

Render Art on a Canvas.



CANVAS 图

Emotional Weather

Project Production

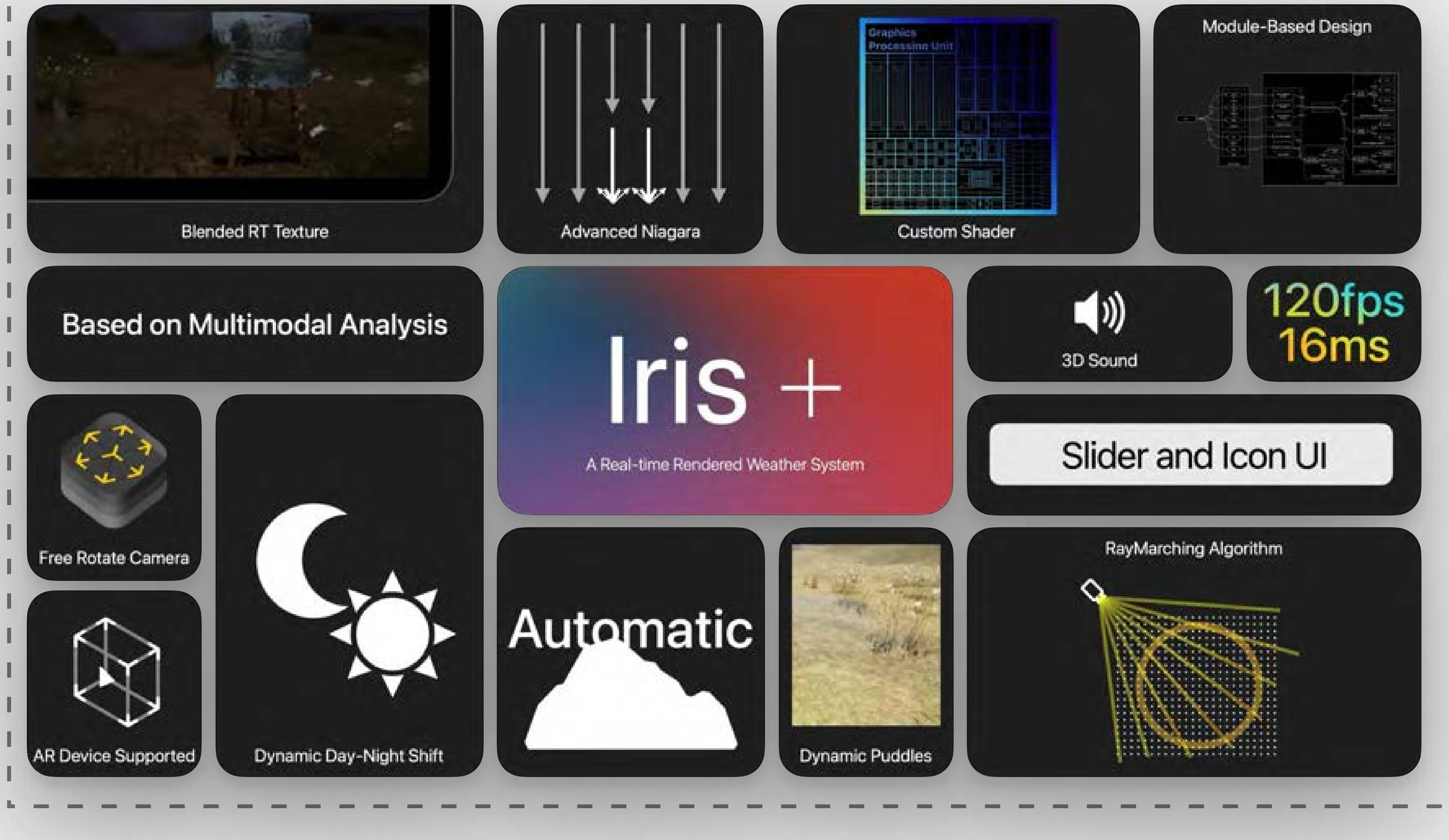
As a weather simulator, the technical workflow start with what should be in side the system: the Landscape and the sky. Apart from that, in order to achieve the rainy weather that we want to made, a Niagara system is needed.
Upon make sure what to be included, a detailed thinking on the technical workload has been processed.

All Technical Workload Overview

Land Scape	Seamless Materials	Terrain Selection Material	Puddle Effect	Wind Effect
Skyboxes and Cloud	Sky Sphere	Volumetric Cloud	Light and Environment Mixer	Day-Night Shift
Niagara System	Liquid Material	Camera-Follow Niagara	Rain Niagara System	Lightning Niagara Ststem
Control and UI	Render Target	Data Storage	Widget Blueprint	

What is included in the project

The weather system have include a lot of advanced techniques like Niagara system and day-night shift and rendering solutions like ray marching. In this project, I have finally achieved a lot of function to make this project sophisticated, vivid, visually pleasing and emotional impactful.

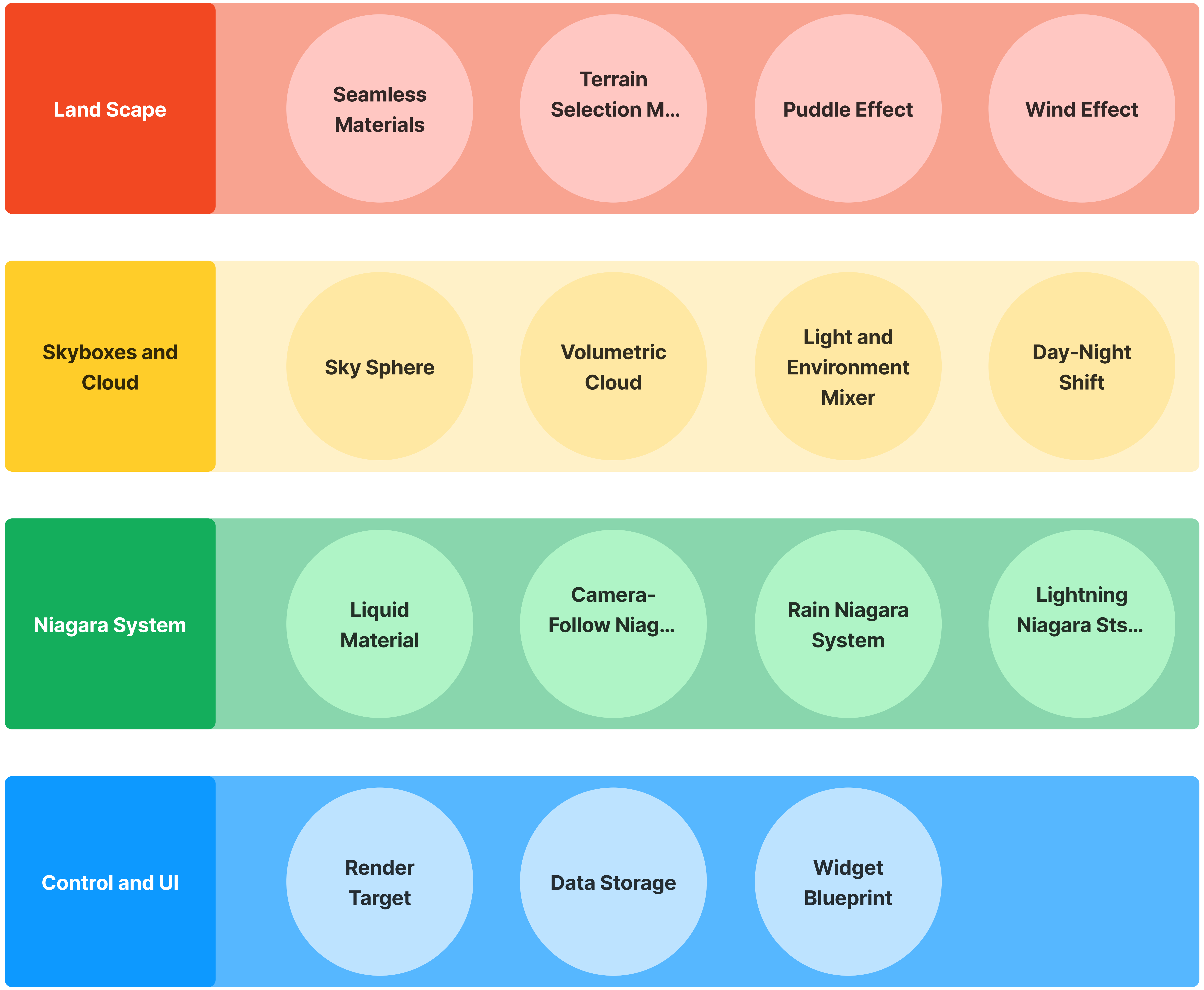


Project Production

All Technical Workload Overview

As a weather simulator, the technical workflow start with what should be in side the system: the Landscape and the sky. Apart from that, in order to achieve the rainy weather that we want to made, a Niagara system is needed.

Upon make sure what to be included, a detailed thinking on the technical workload has been processed.

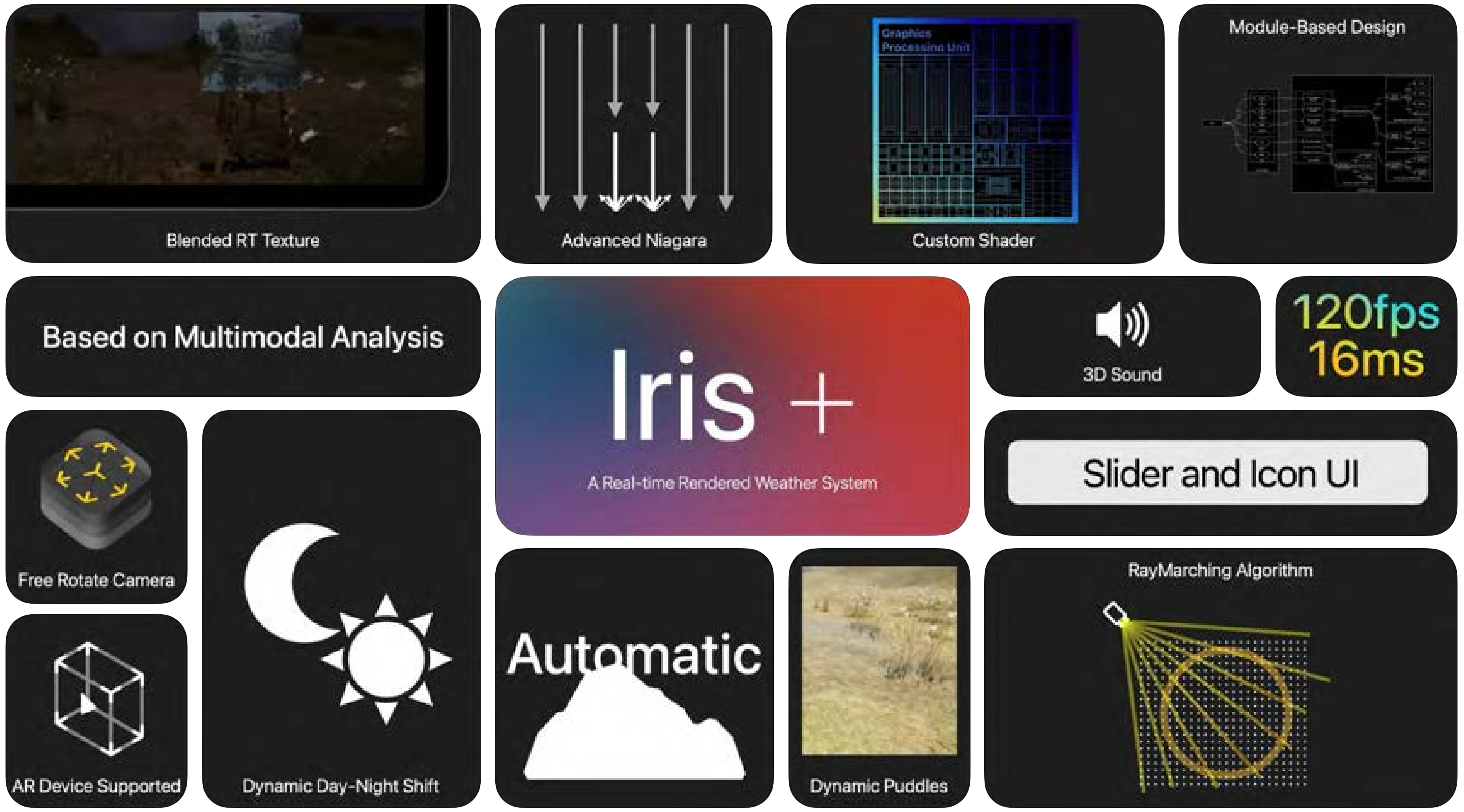


Land Scape: Seamless Materials, Terrain Selection M..., Puddle Effect, Wind Effect

Skyboxes and Cloud: Sky Sphere, Volumetric Cloud, Light and Environment Mixer, Day-Night Shift

Niagara System: Liquid Material, Camera-Follow Niag..., Rain Niagara System, Lightning Niagara Sts...

Control and UI: Render Target, Data Storage, Widget Blueprint

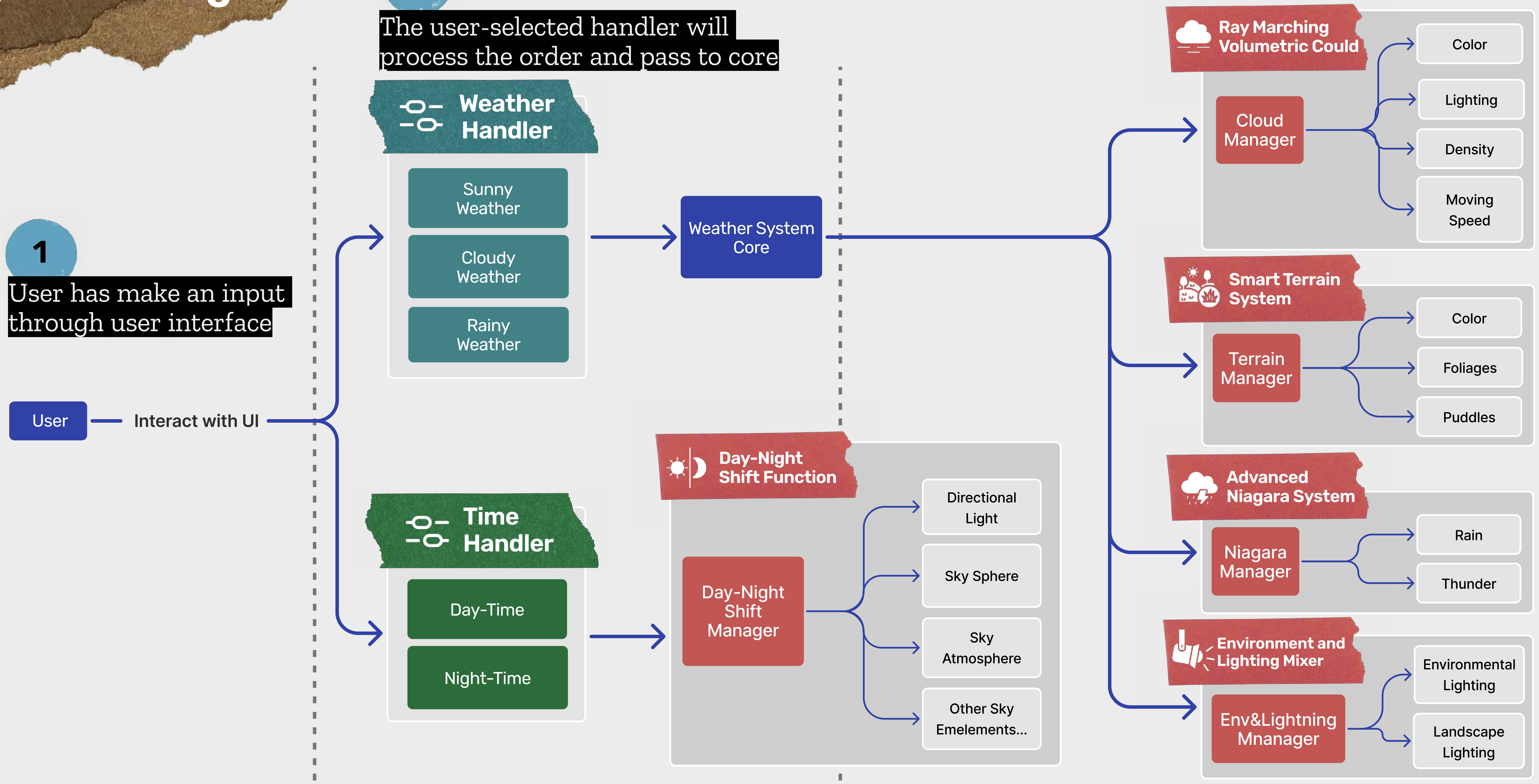


What is included in the system- a overview

The weather system have include a lot of advanced techniques like Niagara system and day-night shift and rendering solutions like ray marching. In this project, I have finally achieved a lot of function to make this project sophisticated, vivid, visually pleasing and emotional impactful.

Project Production

Modular Software System Design



1

User has make an input through user interface

User — Interact with UI

2

The user-selected handler will process the order and pass to core

3

The corresponding manager in the scene translate the user input into output in the system, and render the dedicated effect

Weather Handler

- Sunny Weather
- Cloudy Weather
- Rainy Weather

Time Handler

- Day-Time
- Night-Time

Weather System Core

Day-Night Shift Function

Day-Night Shift Manager

- Directional Light
- Sky Sphere
- Sky Atmosphere
- Other Sky Emelements...

Ray Marching Volumetric Could

Cloud Manager

- Color
- Lighting
- Density
- Moving Speed

Smart Terrain System

Terrain Manager

- Color
- Foliages
- Puddles

Advanced Niagara System

Niagara Manager

- Rain
- Thunder

Environment and Lighting Mixer

Env&Lightning Mnanager

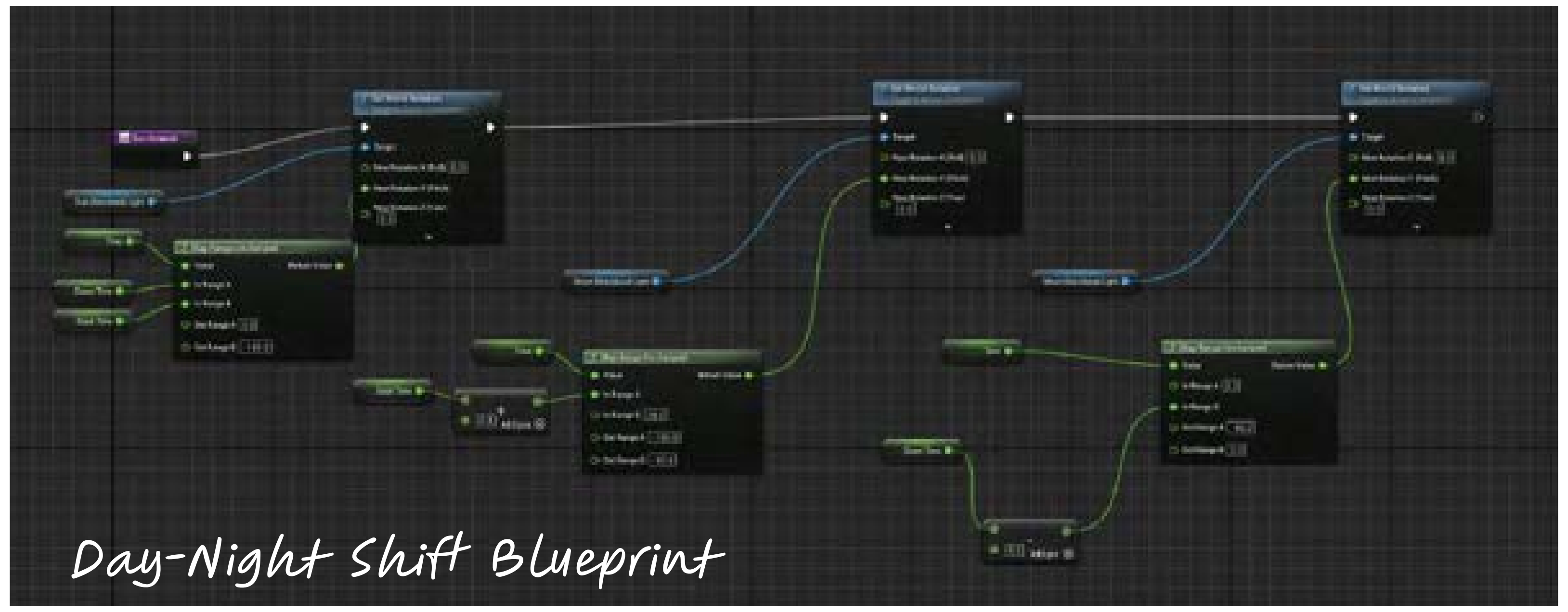
- Environmental Lighting
- Landscape Lighting

Project Production

Day&Night &Cloud

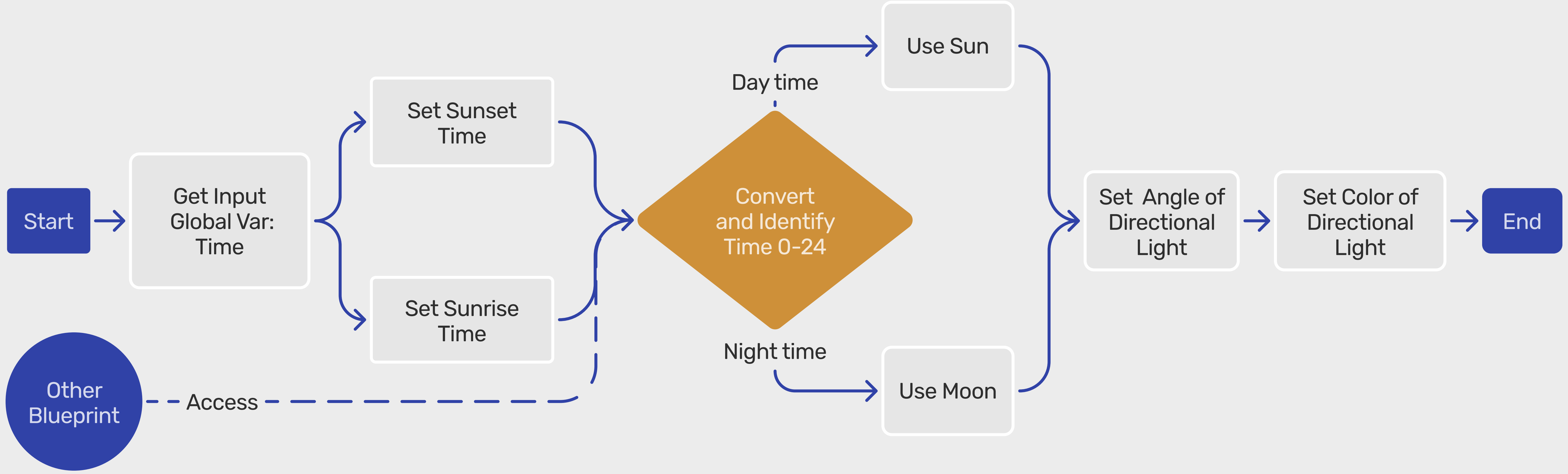
Day-Night Shift System

To ensure users can enjoy the scene of sunrise, sunset ,starry stars and other touching moment that could affect user emotion, a dynamic say-night shift system that allow users control the time in the system has been implemented.



Day-Night Shift Blueprint

Powered by blueprint, Day-Night shift function converts user input into numeric time between 0-24. It visually present a smooth transition from day to night time by changing the skybox and directional lights.



Start
Get Input Global Var: Time
Set Sunset Time
Set Sunrise Time
Other Blueprint
Access
Day time
Convert and Identify Time 0-24
Night time
Use Sun
Use Moon
Set Angle of Directional Light
Set Color of Directional Light
End

Volumetric Cloud

Why Cloud matters?

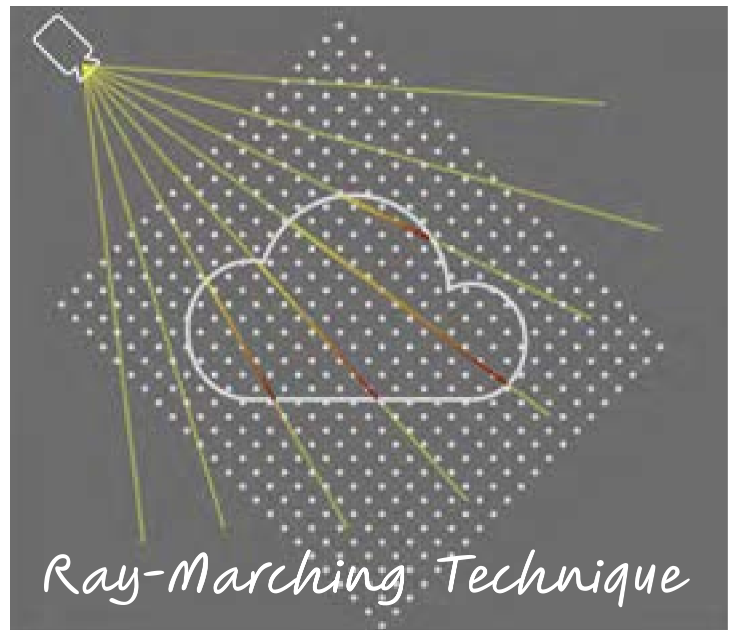
Clouds are the carriers of the weather, with different heights and colours and thicknesses for different weather conditions.

Volumetric Cloud Effect in the project

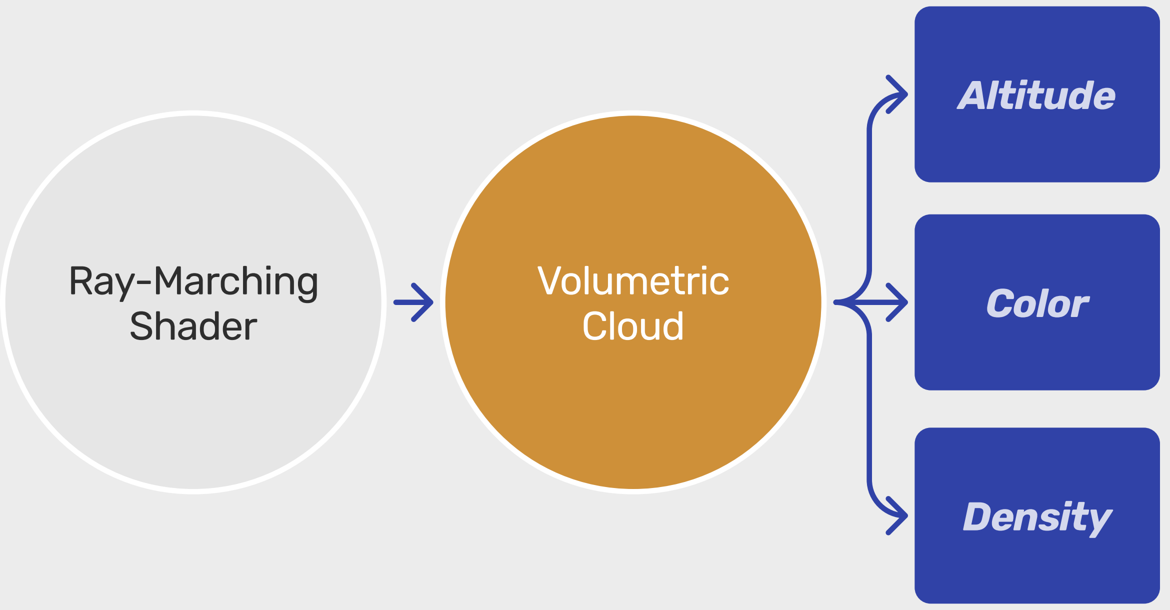
Through the refraction of light and clouds, spectacular scenes like the Tyndall effect can occur.

Tyndall effect in the project

Therefore, I decided to write a separate cloud shader for volumetric cloud rendering by means of light stepping.



Ray-Marching Technique



Ray-marching is a rendering technique in which light is projected from the camera to each pixel, travelling in small steps through 3D space. With each step, an algorithm samples the density of the cloud. These densities are accumulated to determine the final colour and opacity of each pixel, resulting in a volumetric cloud effect.

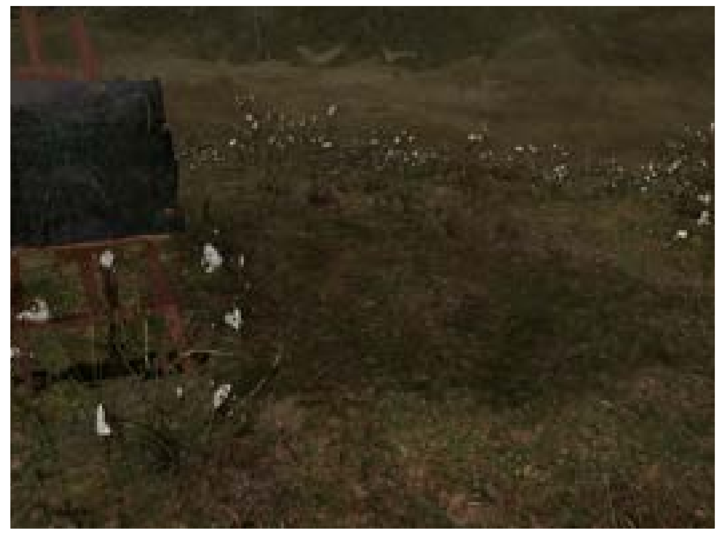
Project Production

Terrain

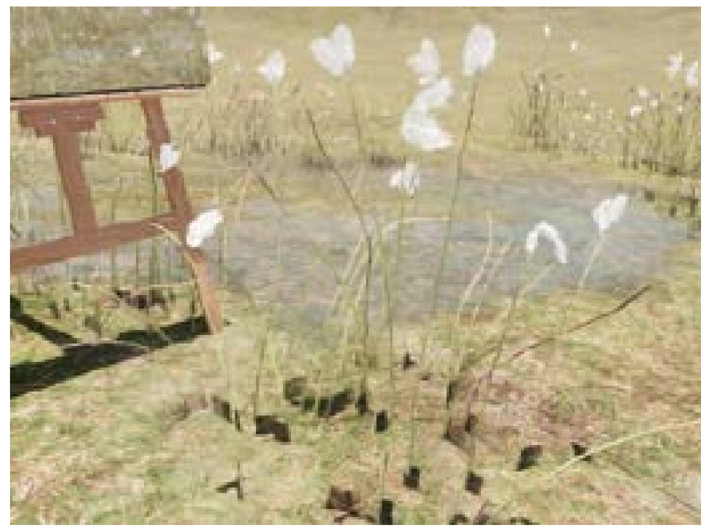
Dynamic Effects on Terrain

Puddles

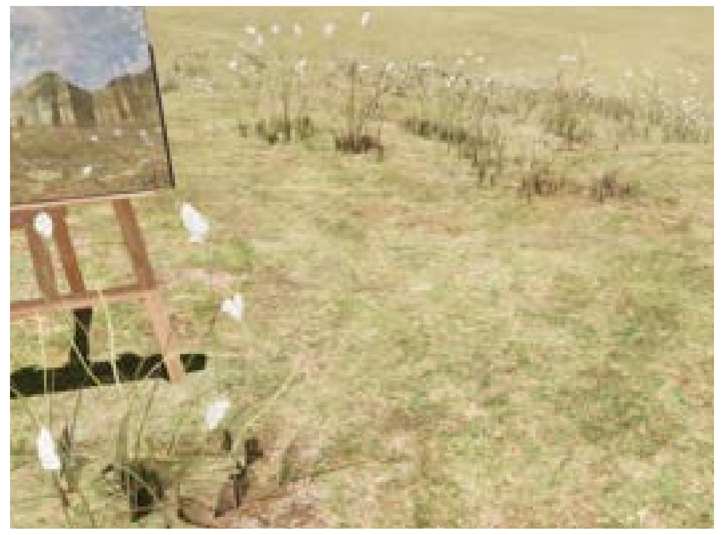
A tangible representation of rain is the puddle, and by creating puddles on the terrain and then varying the size of the puddle, both rainy and rainy weather can be shaped more realistically.



Raining-Large Puddle



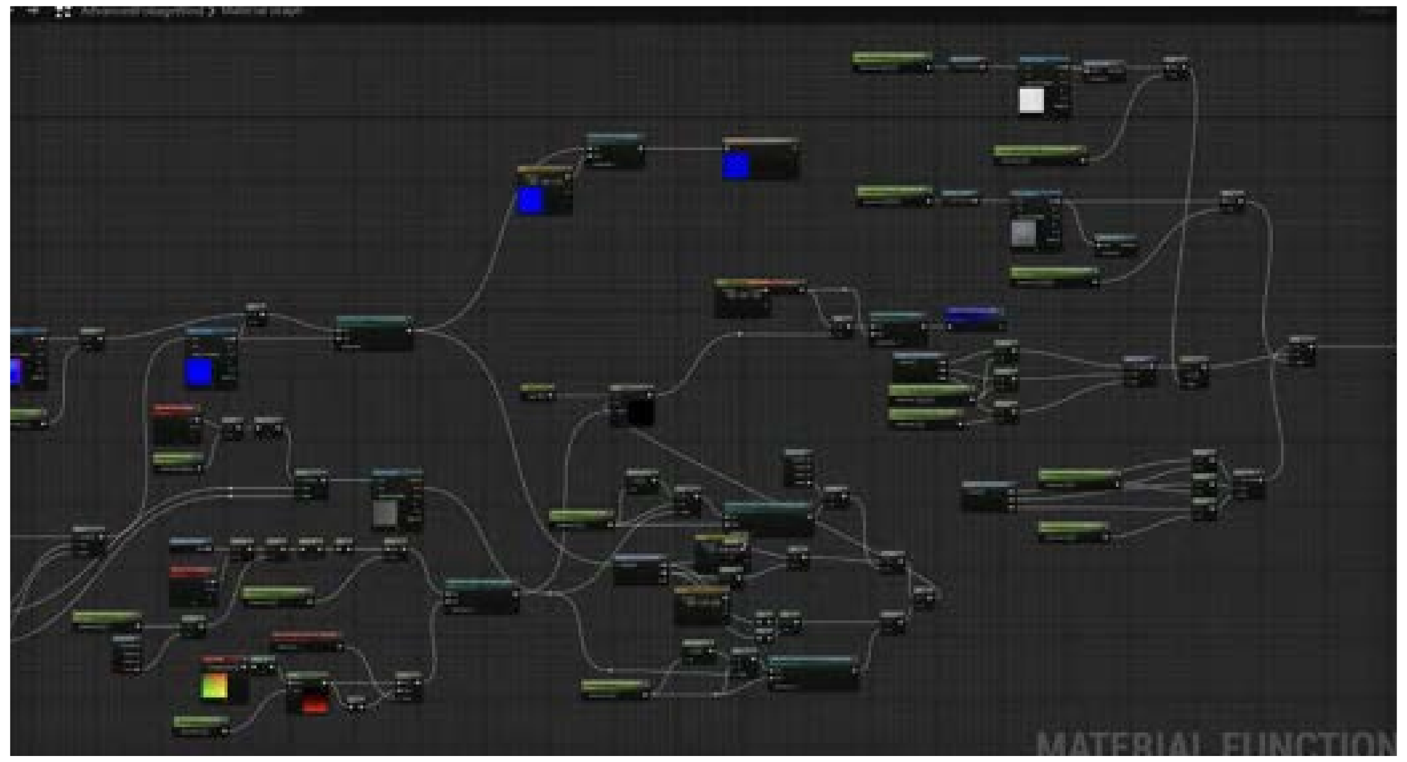
After Rain-Small Puddle



Sunny-No Puddle

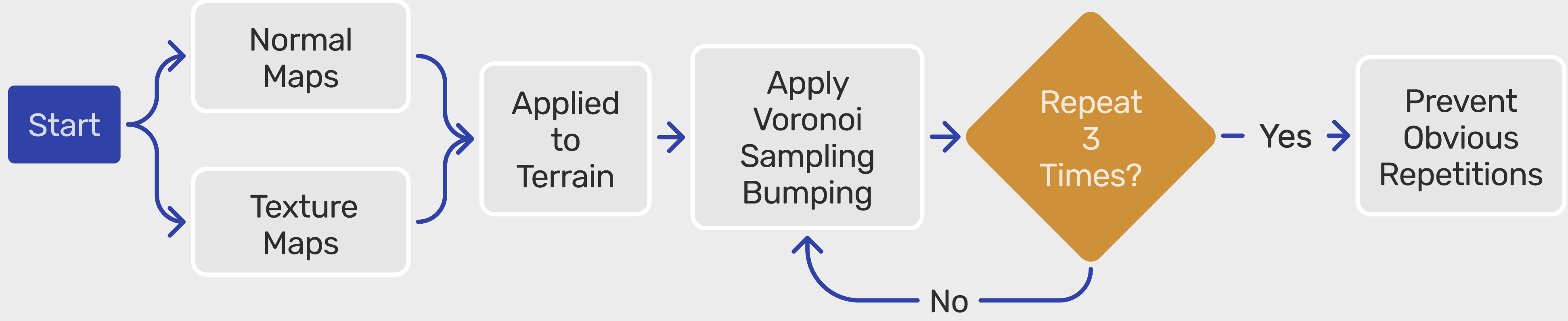
Foliages

Wind is a very important part of weather, but wind cannot be seen directly. At first I planned to represent the wind in the perspective of rain, but then I decided to add Foliages to the project and make these Foliages blended by vertex shader to visualize the wind.

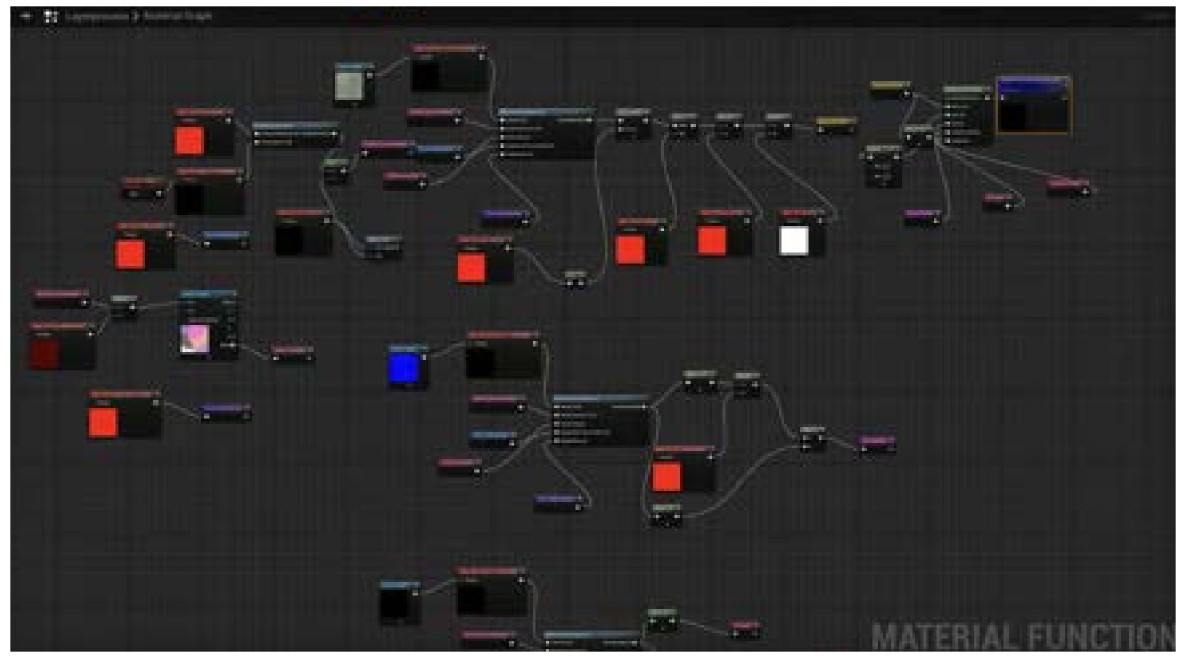


Terrain Material

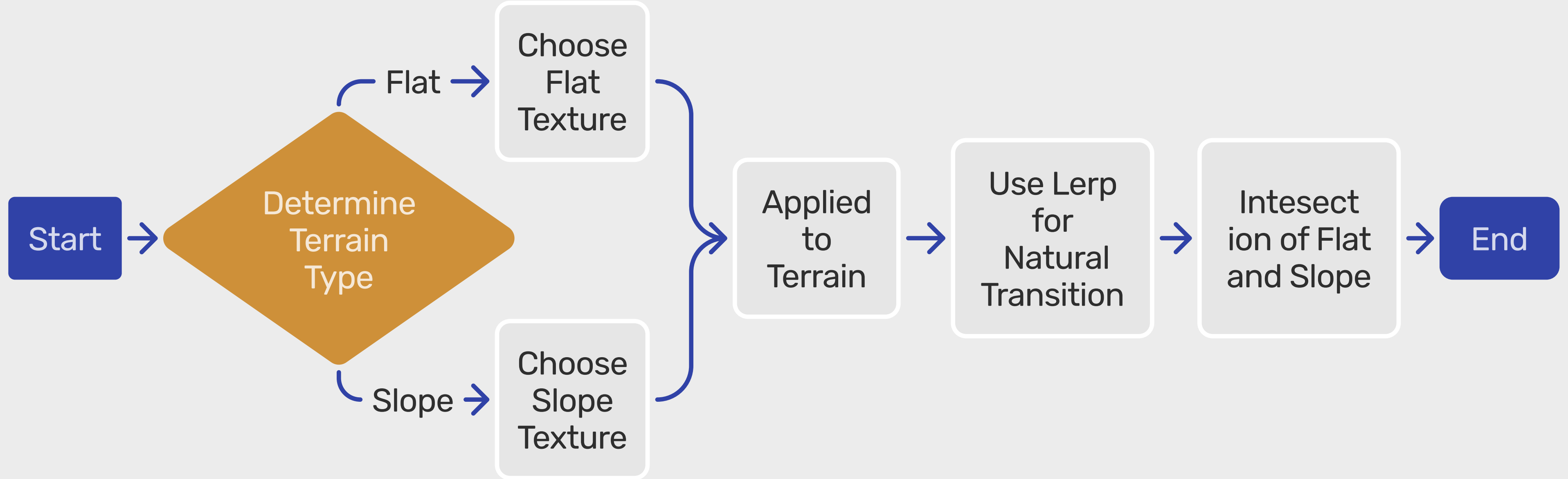
Prevent Material Duplication



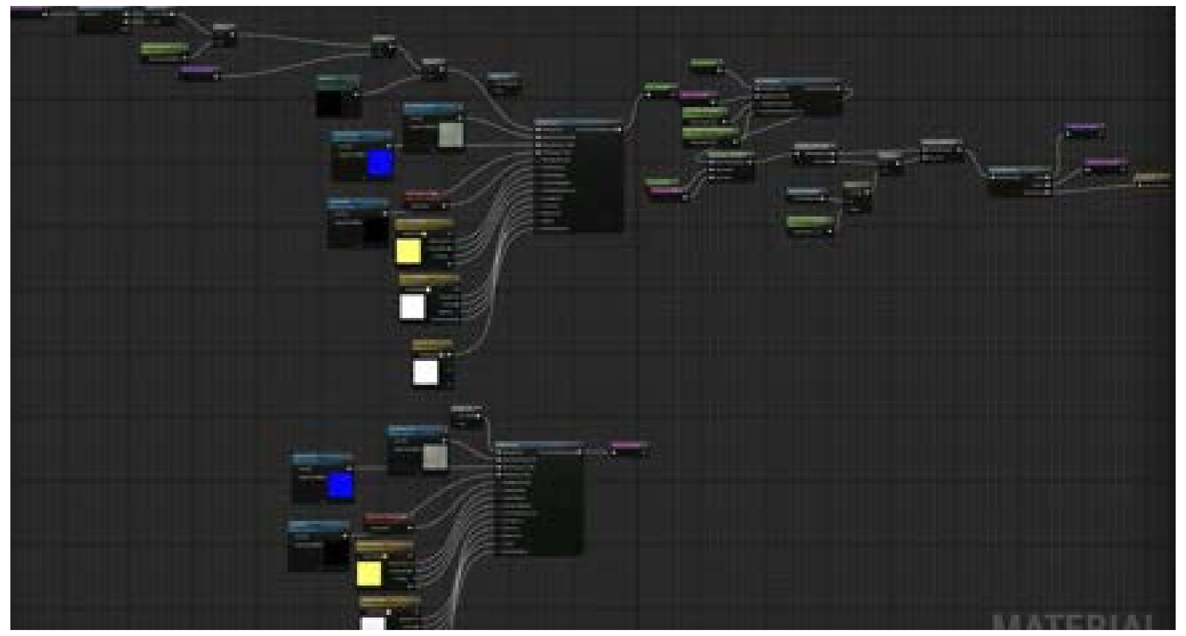
Blended material gets the current weather information from the slider in the UI, and then decides which textures to blend (via the Lerp function for blending effects).
e.g. in sunny to cloudy weather, the material blends the sunny painting, the cloudy painting, and the real-time rendered scenery. This approach ensures that changes in the painting are not rigid and inspires more possibilities .



Select Texture based on Vector of terrain



According to the vector of the location of the terrain, area to determine whether the terrain is flat or slope, and then choose a different texture for it, and through the lerp way to make the slope and the intersection of flat texture natural transition



Project Production

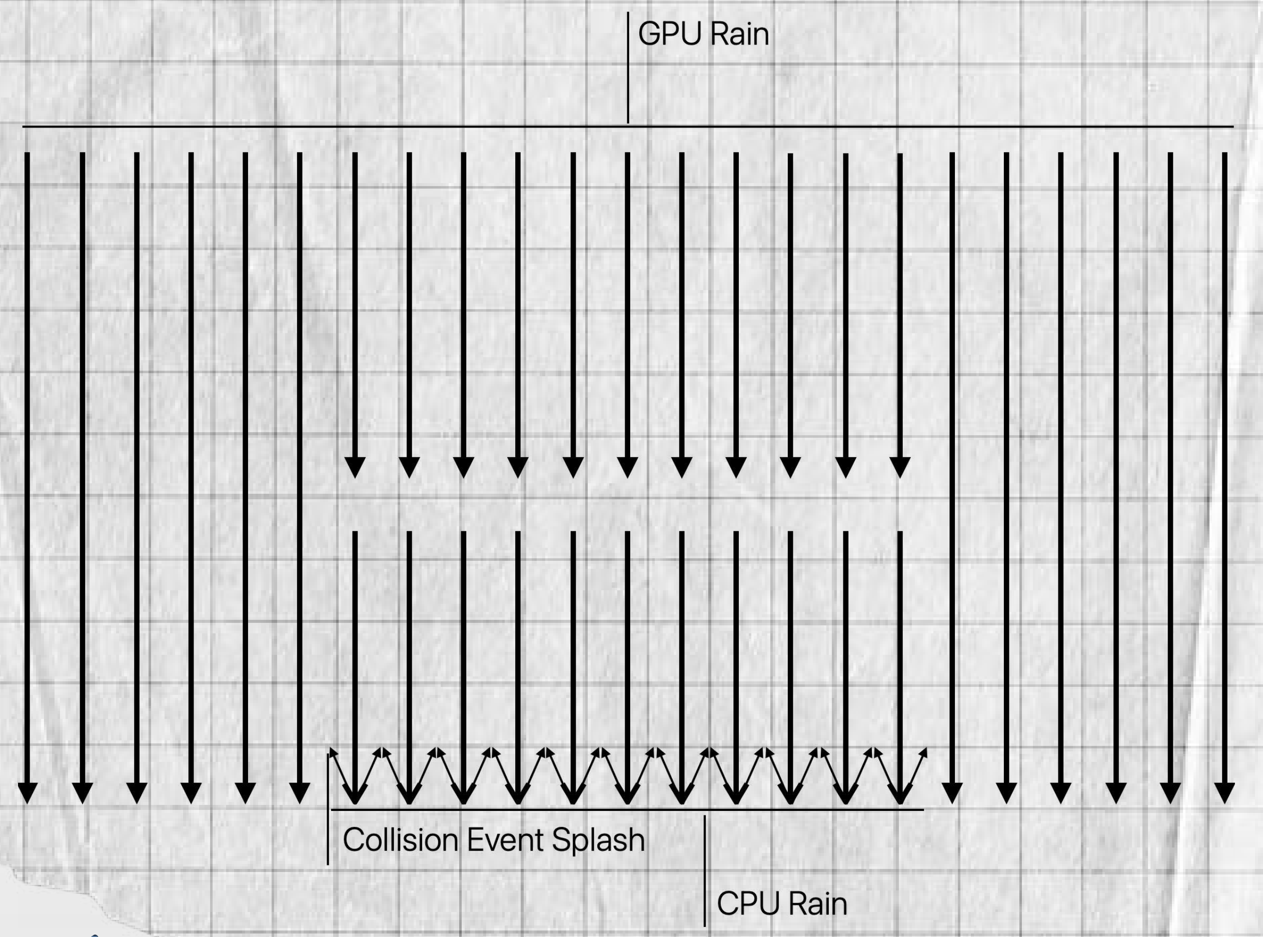
VFX Effects

Rain

Problem1 :
Rain in Whole Scene is Expensive

Solution:
Create Niagara Template that follows camera therefore Rains only Renders in front of Camera

Problem 2 :
Render Efficiency with Rain Splash



GPU Rain Cheaper but cannot trigger collision event

CPU Rain Expensive, powerful, can create colllision event to trigger rain splash

Solution:
Combine CPU and GPU Rain to Improve Efficiency

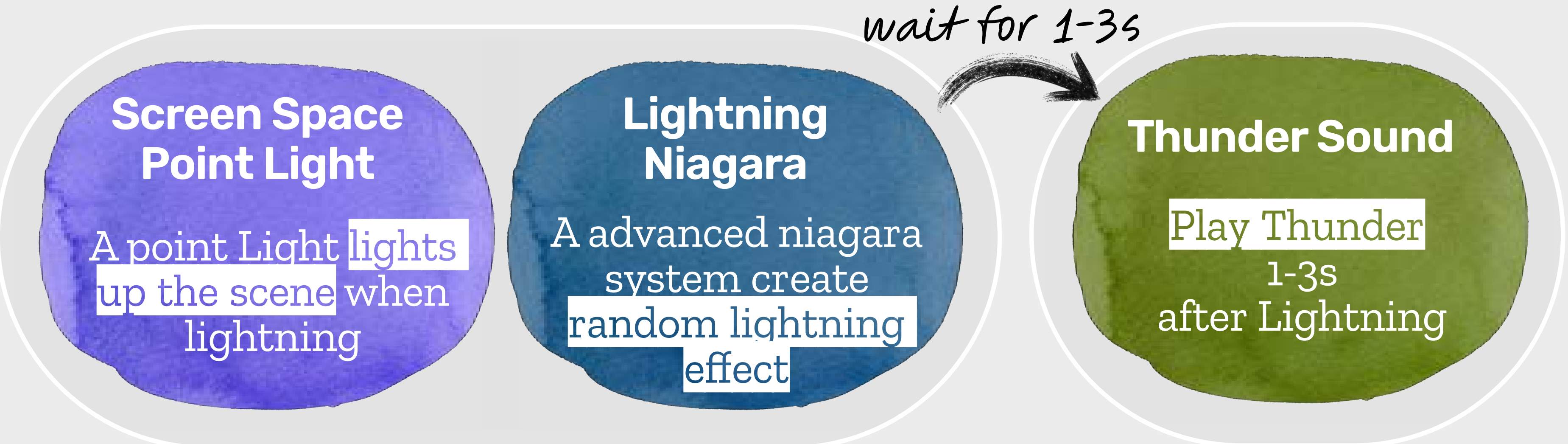
CPU Rain
in the foreground

The Rainwater Niagara system consists of three components: GPU Rainwater, CPU Rainwater, and Splash.

The logic is that GPU rain is rendered in the background in larger quantities, CPU rain is rendered in the foreground and the splash effect is triggered by a collision event, and the transparency of the rain is reduced according to the distance from the camera by calculating the distance between the particles and the camera, which allows the rain not to obscure the camera's view.

Lightning

Thunder and lightning are common elements in stormy weather, and their appearance accompanied by loud noises can be a good way to mobilise the player's emotions.

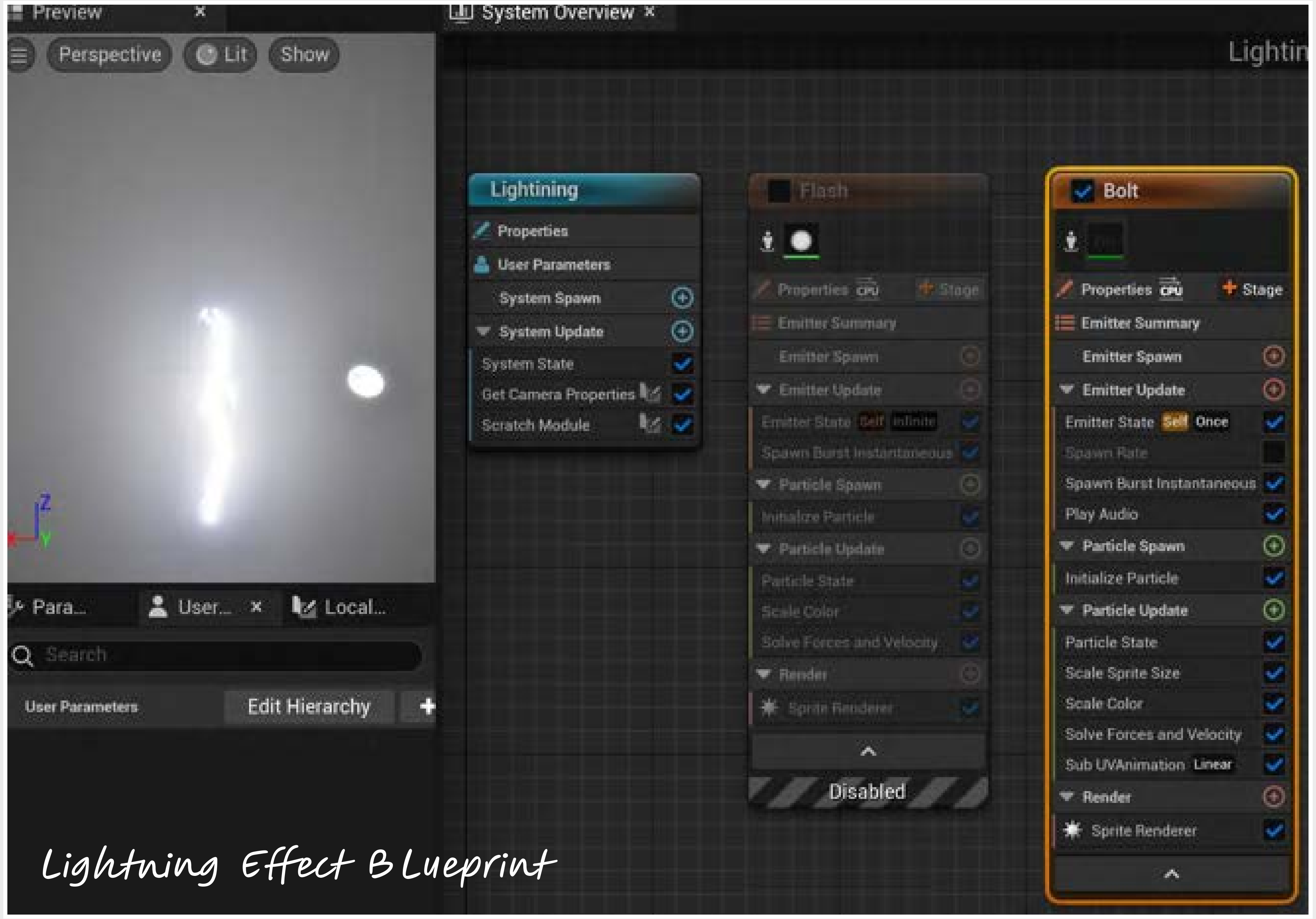


Screen Space Point Light
A point Light lights up the scene when lightning

Lightning Niagara
A advanced niagara system create random lightning effect

wait for 1-3s

Thunder Sound
Play Thunder 1-3s after Lightning



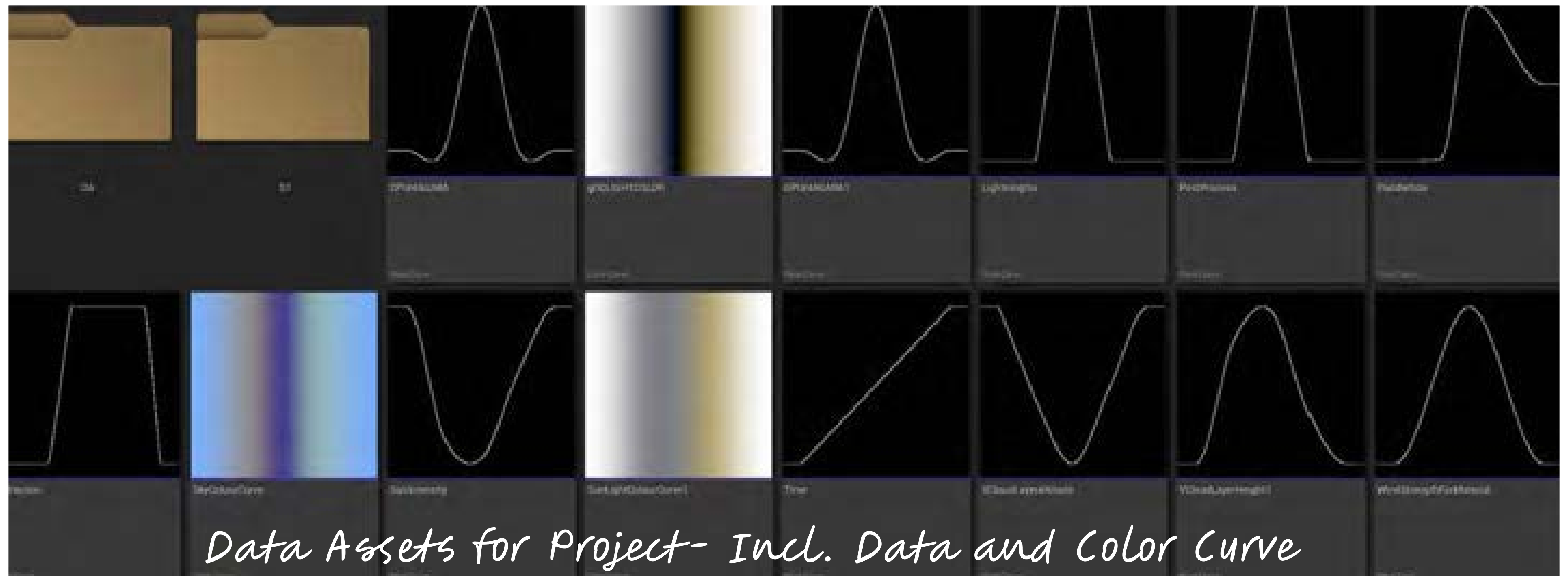
Lightning Effect Blueprint

In order to better present the effect of thunder and lightning, the thunder and lightning particle system underwent several iterations. The initial system simply included a lightning bolt but no other effects, so a point light source was added to light up the whole scene for further simulation.
In addition to this, there is a time lag between the lightning and the thunder in order to make the effect more realistic.

Project Production

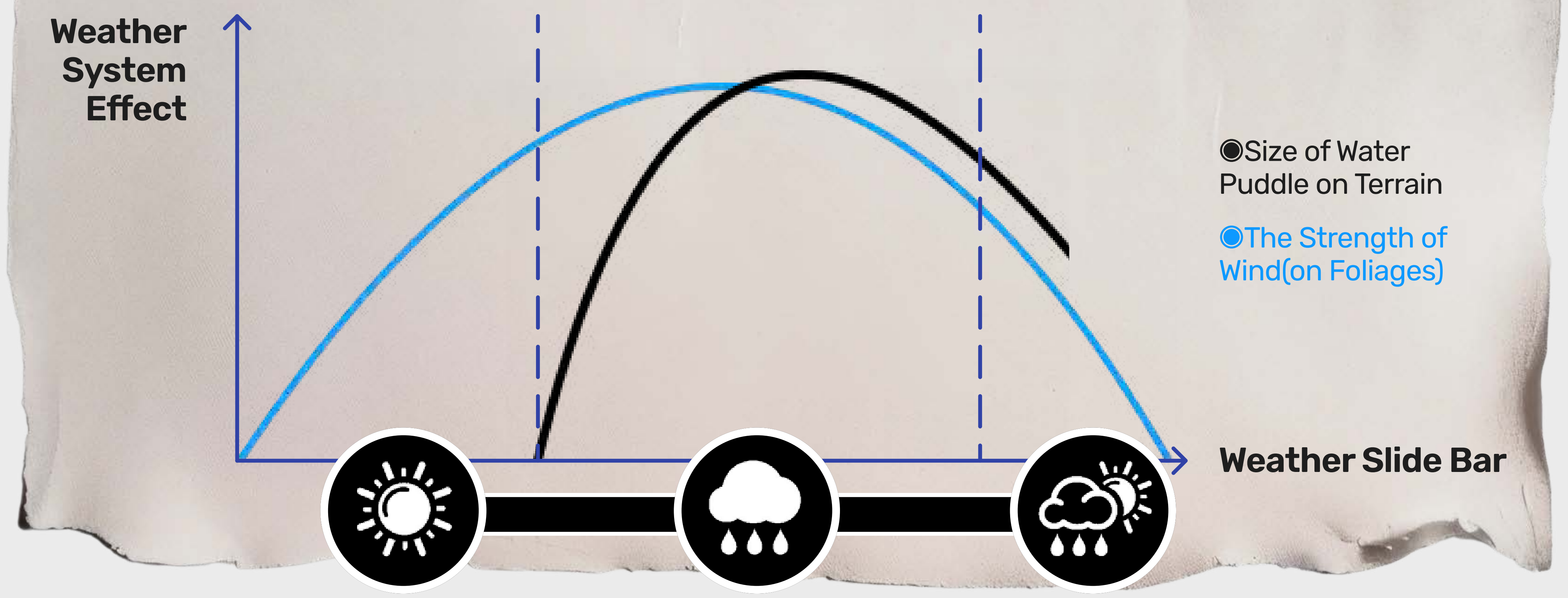
Data&UI &Sound

Data Storage


Data Assets for Project- Incl. Data and Color Curve

The project controls presets through data Asset and Enum. It provides a silky smooth translation between weather by using user inputs to X-axis values and setting individual elements of the weather system to Y-axis values.

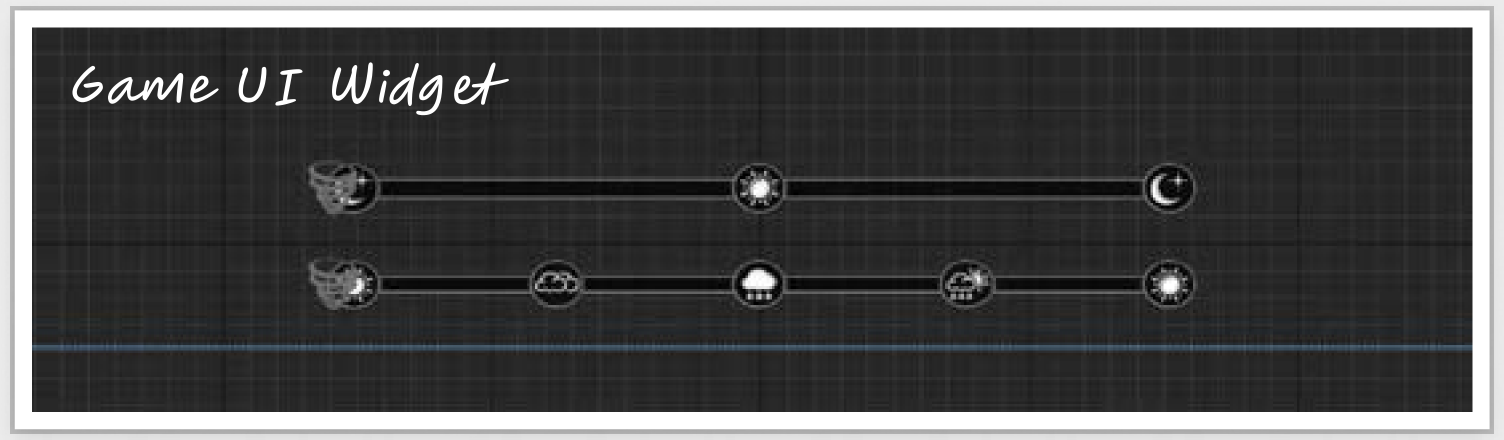
Weather System Data Assets- Take Terrain Puddle as An Example

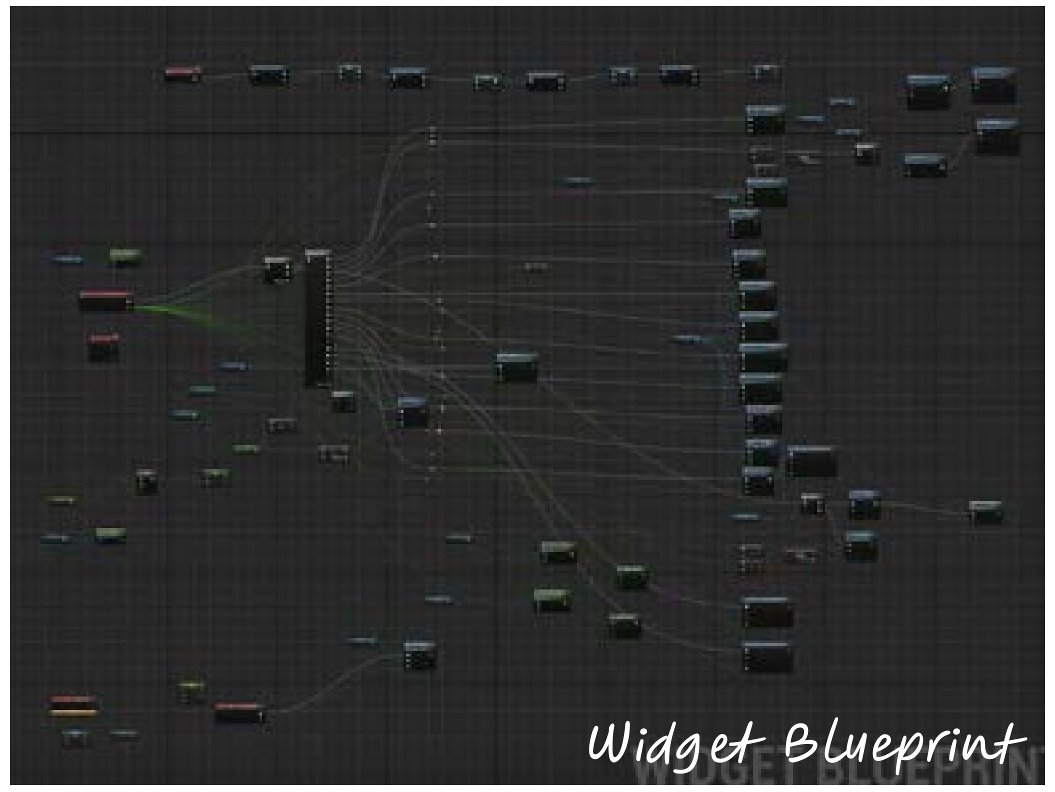


As the graph shown, base on the input of user(x-axis), the size of the puddle changed as the curve value.

UI Iteration

The first version was simple buttons but then the buttons were changed to sliders considering that the changes between the different weather were also part of the fascination of the project.


Game UI Widget


Widget Blueprint

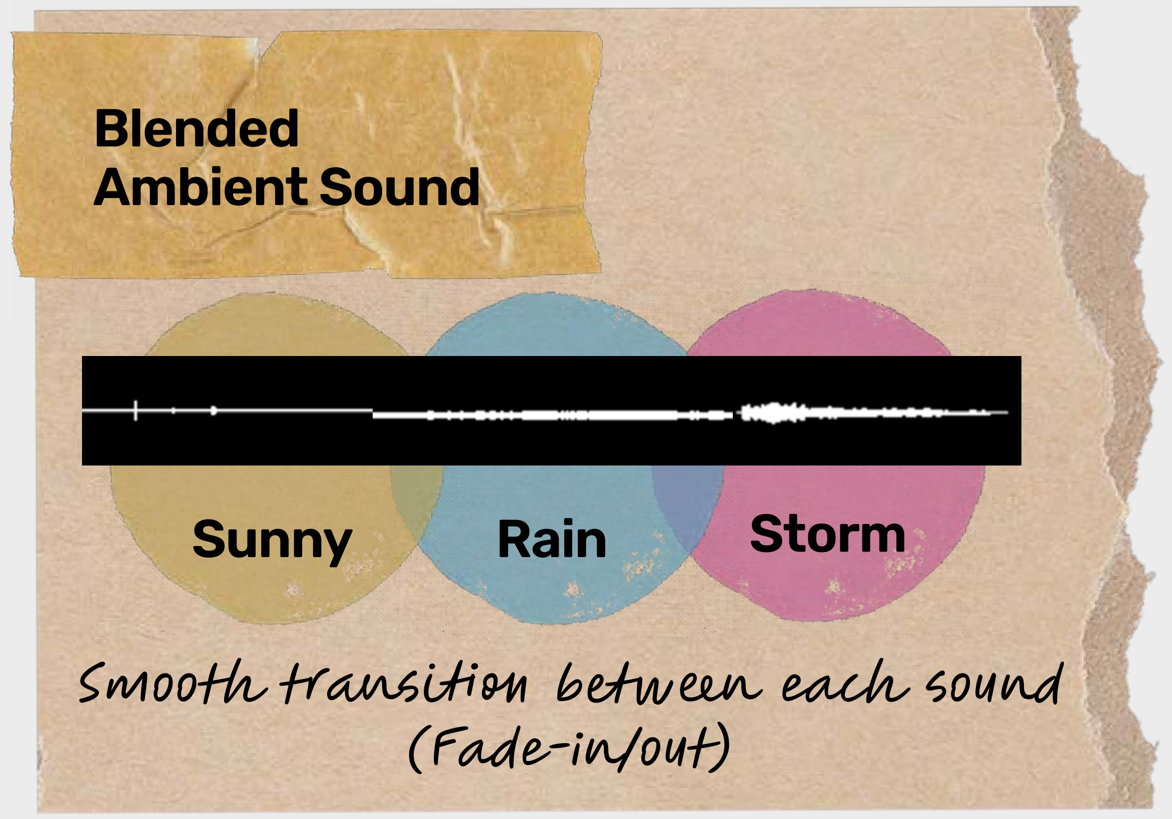
Finally, the project consists of two sliders: a time-control slider and a weather-control slider, both triggered by the on-value-changed method, with additional logic set to disable the time-control slider on rainy days.

the time-control slider implements the Day-Night shift. the weather-control slider implements the weather-control slider.

The main functionality of the weather control slider is to control the sunny-cloudy-rainy day shift and the generation of puddles on the ground and trigger the associated particle system.

Sound Design

To Help the player immerse themselves and build an emotion connection in the virtual weather system by trying to keep things simple in terms of sound effects and removing unnecessary sound effects like background music and wind sound effect.

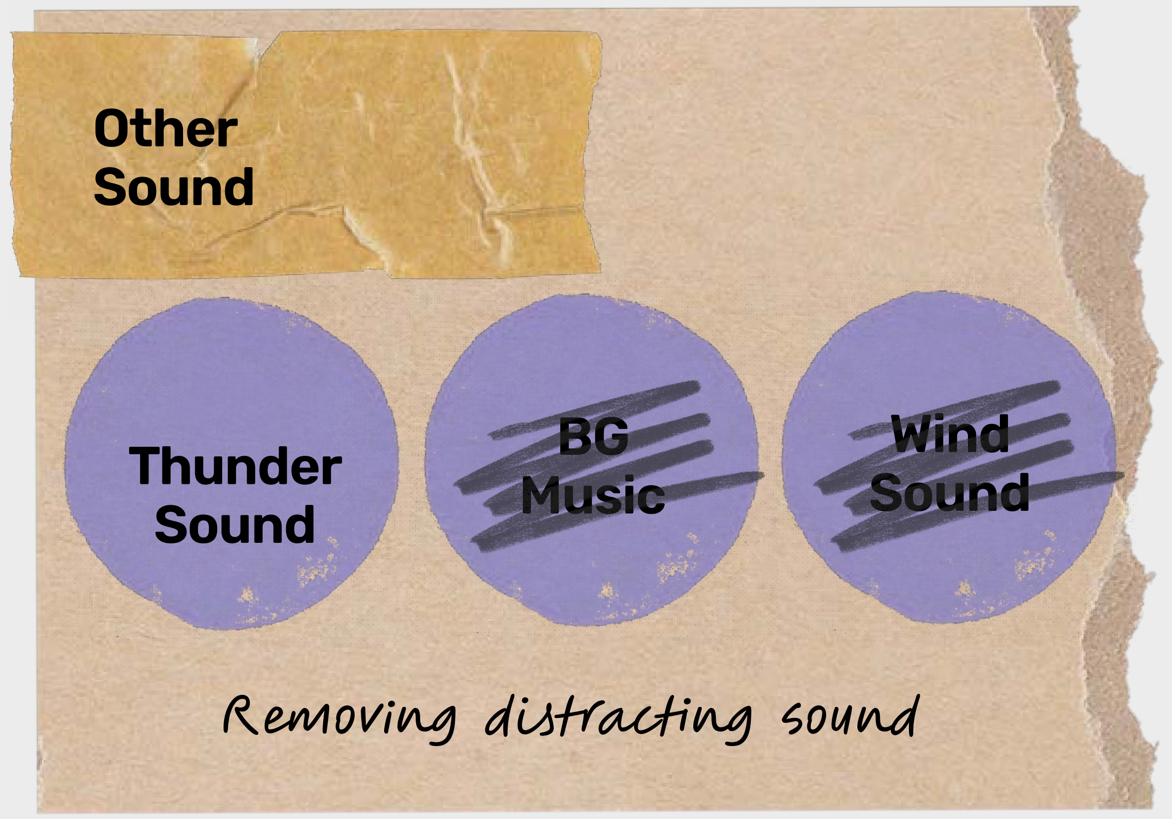

Blended Ambient Sound

Sunny

Rain

Storm

Smooth transition between each sound (Fade-in/out)


Other Sound

Thunder Sound

~~BG Music~~

~~Wind Sound~~

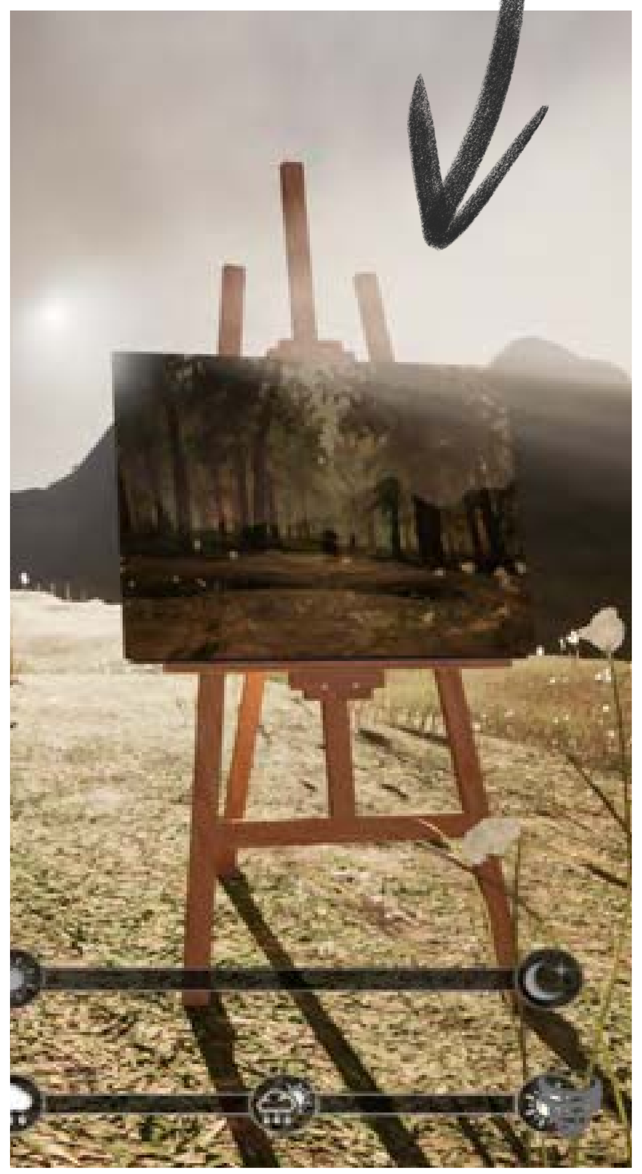
Removing distracting sound

Project Production

Art Render & Polish

Art and RealTime Render

The Art Work on Canvas



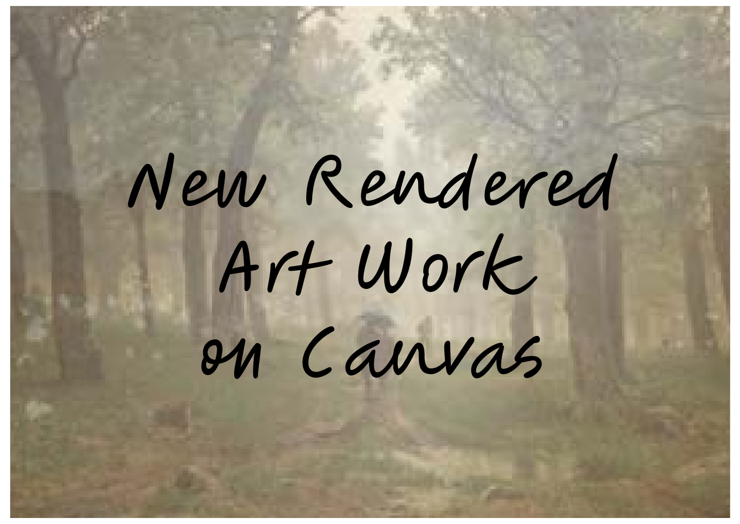
Apart from build a emotional connection with users, the project also reflect on using code as a tool to create art work, with this mission, the project featured a real-time rendered art work on the canvas in the scene. And it has be blended with some fine artwork.

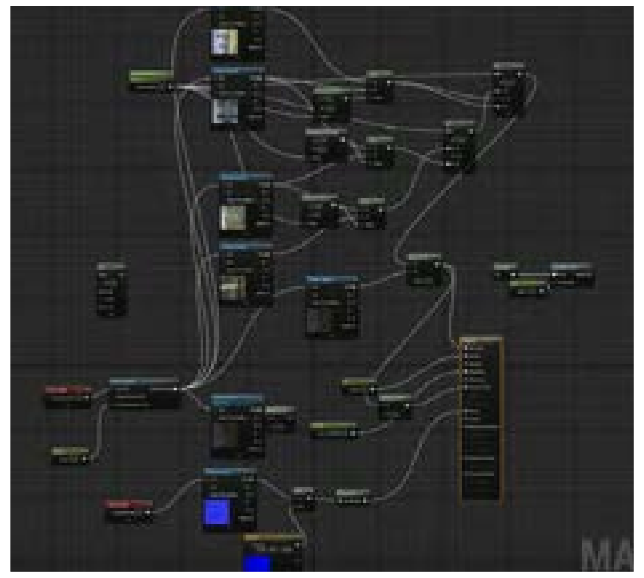
Weather in classic painting

+

Weather in Rendered Scene

=


New Rendered Art Work on Canvas



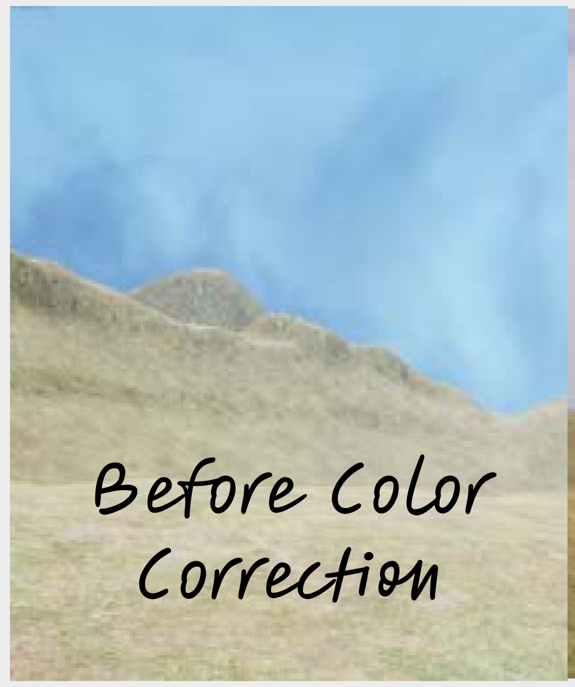
Blended Material

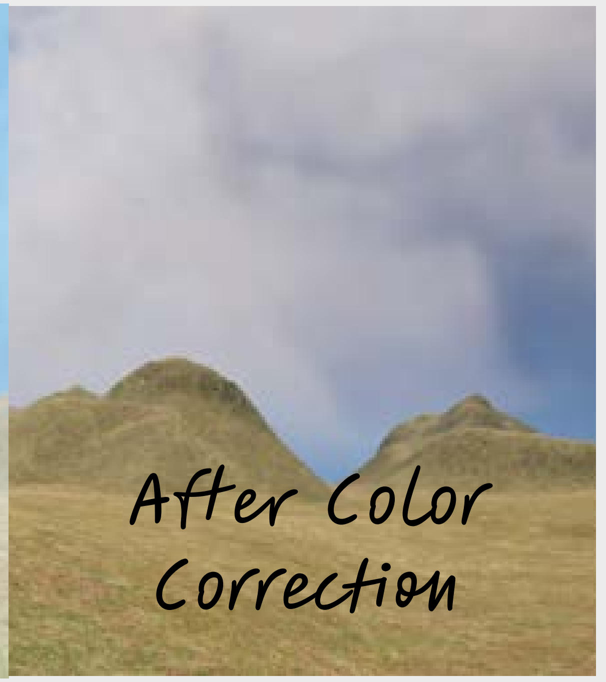
Blended material gets the current weather information from the slider in the UI, and then decides which textures to blend (via the Lerp function for blending effects), e.g. in sunny to cloudy weather, the material blends the sunny painting, the cloudy painting, and the real-time rendered scenery. This approach ensures that changes in the painting are not rigid and inspires more possibilities .

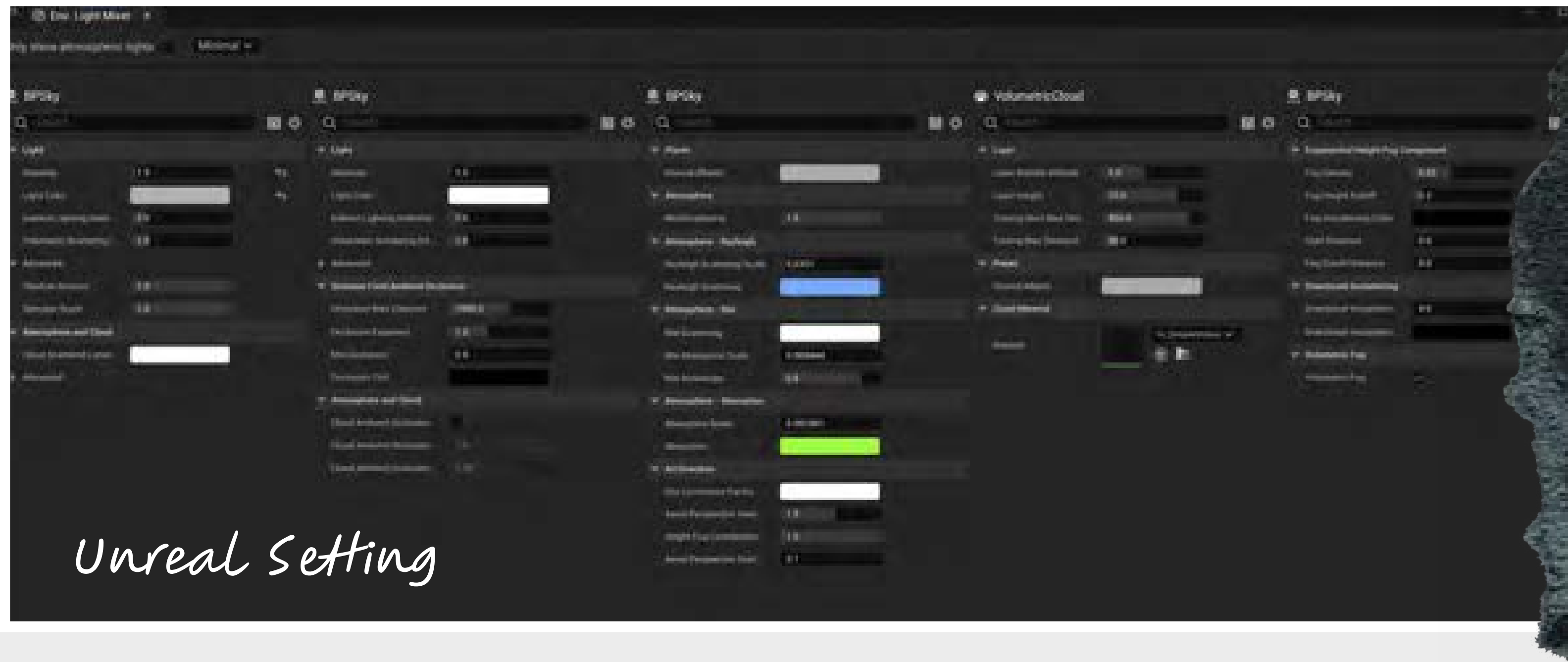
Ambient Creation and Final polishment

Compare with real-life weather scenery

Adjust colour presets for different weather through videos/pictures.


Before Color Correction


After Color Correction

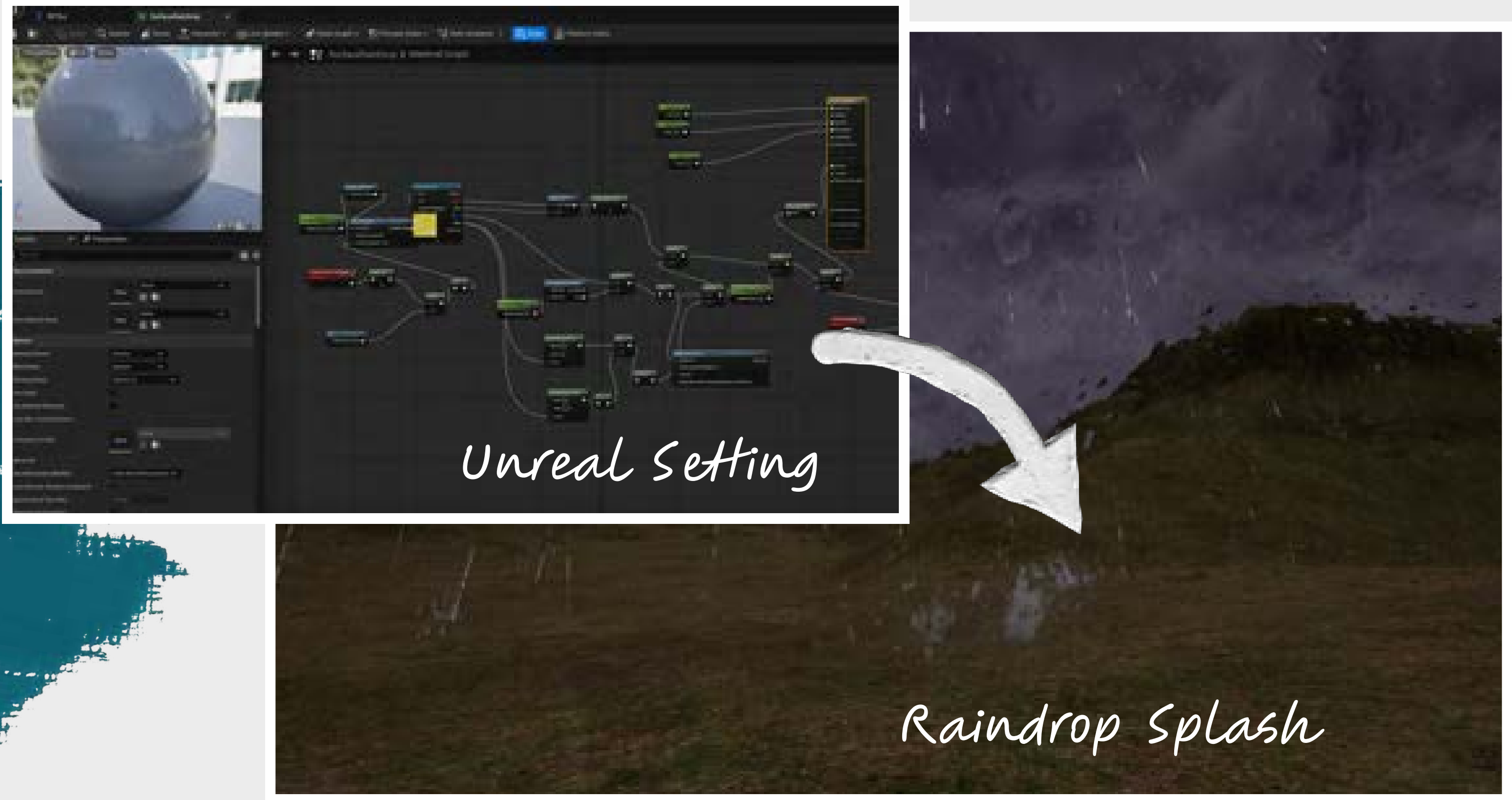

Unreal Setting

Light & Environment Mixer

By further changing the settings in Light and Environment Mixer, the scene's visual effect has been improved and the ambient has become more vivid.

Screen Raindrop Layer

To give the project a better rain effect, I added a glass rain effect to the rainy scene, so the viewer would see raindrops dripping down onto the lens and changing as the rain got heavier.


Unreal Setting

Raindrop Splash

Emotional Weather

Reflection

This project explored the relationship between user emotions and the weather system through a painting-like weather system that allows users ample room for imagination without guiding mechanisms. The project also incorporates fine art, exploring the infinite possibilities of computer graphics in creating art by combining weather systems and paintings. However, compared to weather systems in released games, this project needs improvement in rendering efficiency and completeness

Technically,many of the settings in the project were overly reliant on terrain with the Unreal Engine,which caused many problems, such as mountains created through sculpting being far less realistic than those generated in Blender through heightmap, as well as errors and biases in the Unreal Engine's calculation of terrain shadows. In future projects, trying to create mountains using modelling software and then creating materials via shaders would greatly enhance the project's visual impact.

Further Development

1.Sound and Narrative
By adding more mechanics and storylines, it can better complement the weather system for a complete narrative and further engage the player's emotions.

2. Artistic integration
The project tries to integrate real-time rendering technology with art, but the current integration is through algorithms for overlaying and blending, a better solution is to analyse the paintings of different painters to create different material function and style presets, and then use them to create new art work

3. Further Extension
The research questions could have been better addressed with the inclusion of an anthropological study. For instance, participants could have used the simulation system while their emotions were recorded using a participant observation approach. The exploration of integrating art and computer graphics technology through fusion rendering of classic paintings and real-time weather could have been enhanced by creating more weather variations and offering players the option to paint more.

Final Presentation