Dissertation

* Introduction: scene setting
* State the original aims/hypothesis and objectives for your project, but elaborate on any changes that have occurred over the project’s duration.
* Outline of the dissertation (what material is where, and why)
* .Background technical material
  + This will be a review of relevant background research, technical, and/or commercial material depending on particular emphasis of the project.
* What you did and why you did it
  + This will include a high-level explanation rather than a low-level details
  + Software engineering aspects (design, implementation, testing strategies, etc.)as appropriate.
* Results and Evaluation
  + What has the project produced as an output
  + Evaluation of these results
  + Overall evaluation of the “information systems” or “computing science” aspects of the project (e.g. plans and engineering approach)
* Conclusions
  + About the original aims (hypothesis) and objectives
    - -how well were the original objectives met (and why weren’t they all met?)
* Both positive and negative
* Look back -what has been learnt
* Future, proposed follow-on work
* References (including any URLs)
* Appendices: additional material that might be referred to in the main body of the dissertation, for example, a user manual, questionnaire responses, transcripts of interviews, and where relevant source code. Only include material that might help the marker verify statements and claims that you make in the main body of the dissertation.

# Supervisor notes

**Introduction**

Good academic papers typically have an introduction which covers the following four things (typically in a paper introduction each of these items is a paragraph, but you have a bit more room in a dissertation):

* State the problem – why is the work interesting. For a dissertation this should describe the motivation and context as well as the technical challenge and scope
* Describe what has gone before – how has the problem been addressed in the past by other researchers, and, if relevant, why is this not yet a complete solution
* What you have done. How has this paper/dissertation built on the previous work. Aim and Objectives help here. For a dissertation, you are probably investigating an existing technique, or comparing a couple of existing algorithms in a specific context.
* How did it turn out? Say in the introduction what the outcome was, how it was measured and how that compares to existing work (ie this isn’t a mystery novel, we want to know how it ends right at the start)

**Background and Research**

This chapter should cover two things in detail:

* A review of existing academic works. Use Google Scholar, identify papers related to the field. Start broad (eg procedural generation in general) and then narrow down to your particular topic (eg city generation). In particular, state why each paper mentioned is relevant to your project and how it has influenced your decisions.
* The algorithms and technology used, in detail. If you are implementing an existing algorithm, this is where you describe the algorithm in detail in such a way that a competent programmer could also implement it.
* It’s okay to have a few references to games articles, but the bulk should be peer reviewed academic papers (journals and conferences)
* Then summarise the findings of the background chapter, relating to decisions made for your project. You should then refer back to these findings in the Implementation and Evaluation chapters (ie when describing implementation decisions, support them by referring back to findings from the research, similarly when identifying an evaluation method, or describing specific results, support them with evidence from other people’s research, or state that they are different and why that might be)

**Implementation**

* In detail, what you did and why you did it. This is your work. If you designed or added to an algorithm talk about that here. If you created a program to implement an existing algorithm, talk about that. Probably you combined various techniques, algorithms and technologies, so talk about how and why you did that.
* It is okay to discuss any blind alleys you spent time on – describe that the idea was and why it was abandoned or changed, in the broader context.
* As mentioned above refer decisions back to findings from the research chapter that support them.
* Code snippets can be useful if they support the text but not a requirement
* Screenshots are definitely useful if relevant. Show the progress of the project from early protype to final build.
* End with a summary of the implementation and mention how it will be evaluated

**Results and Evaluation**

* Clearly identify the evaluation technique
* This should be quantitative. Ideally there should be graphs where one parameter is changed and another is measured (eg density of vertices against performance)
* Push the system to breaking point – make it fail, show that on the graph, and describe why it fails at that point and what could be done to address that (and what the downside of that solution may be)

**Conclusions**

* Overview of the project. How did it go. Refer back to the initial objectives, and the findings of related works – how do your results compare
* Further work, what more could be done in a further project – how could others build on your work

# Introduction

## Motivation

*State the problem – why is the work interesting. For a dissertation this should describe the motivation and context as well as the technical challenge and scope*

One crucial element of modern video games is the atmosphere, the feel of the world where the player is introduced. Whether the sun is gleaming through the leaves in a small forest clearing where a fabled sword is waiting to be plucked from the stone by a brave hero, or whether the rippling of the rain is mingling with the tunes of jazz in a New York detective’s office, the weather can be a very powerful story telling element. Having the right weather at the right moments can invoke high emotional intensity within the player and make for a more memorable experience. In open world games, weather is one of the many elements that immerse the player into a world that feels real, while games that simulate car racing might benefit from altering physics based on weather, translating rain into more slippery roads.

In a piece of artistic production, where nothing happens without reason, there must be some symbolic value attached to the fact that "It was a dark and stormy night." This naturally pertains to creating the atmosphere. Floods and storms create a sombre looking scene while sun and rainbows give way to more positive impressions.

Weather can also be used as a device to advance the story. Would Noah still build the ark if it were not for the flood? Most likely not.

In a more symbolic sense, a character standing in the rain can be cleansed of past misdeeds or happenings, or contrarily, they can fall in the mud and get dirty.

Triple-A(AAA) games have showcased deeply realistic and detailed weather, which is, of no small part, due to their budget and resources. A sterling example of how much weather can do to bring a game to life was displayed in Ghost of Tsushima by Sucker Punch. Aside from the stunning effects of fog, rain and clouds the weather element that stand out in particular is the wind. On the island of Tsushima where the action is set, the wind blows in the direction of the quest encouraging the player to immerse themselves into the scenery reducing the need to open a map and break the illusion of realism. Aside from particles, there are trees, grass, cloth and ropes that were tuned to move appropriately under different wind conditions. A set global wind direction is integrated into most effects in the game; when a bomb goes off or a fire is lit, the smoke drifts in the correct direction. https://blog.playstation.com/2021/01/12/how-stunning-visual-effects-bring-ghost-of-tsushima-to-life/The speed of the wind is also being sampled to add turbulence as the wind speed increases.

Adding petrichor inducing rainfall is not an easy task for the average game developer however, as there are many elements that go into a well-designed weather system. Any who dare dream of adding this particular layer of realism into their games must first acquire learning of particle generation, shader coding and texture rendering among many others.

Capturing the essence of all the elements that make weather so intrinsically beautiful and translating them into code is a hurdle that stops many independent game developers from integrating weather into their games.

Learning resources specific to the problem of creating a weather system have been found to be very limited and what material there is, was found to be outdated and difficult to put in practice.

Documenting the development of a weather manager in this thesis will hopefully shed some light on the process, so that in the future, more game developers choose to not give up on weather effects.

## Current weather systems

*Describe what has gone before – how has the problem been addressed in the past by other researchers, and, if relevant, why is this not yet a complete solution*

## Aim and objectives

*What you have done. How has this paper/dissertation built on the previous work. Aim and Objectives help here. For a dissertation, you are probably investigating an existing technique, or comparing a couple of existing algorithms in a specific context.*

The overall aim of this project is to study the development of an immersive weather simulation framework.

To achieve this, the following objectives must be followed:

* Identify and research existing key technologies and practices that best replicate weather and atmosphere in video games.
* Design and incrementally implement the determined individual weather elements.
* Design and implement a high-level user-adjustable manager component.

Creating a fully-fledged weather simulator is both a technical and artistic challenge. Having poorly implemented components, such as unrealistically moving particles or badly rendered textures, would run the risk of ruining the player’s immersion. Therefore, identifying and understanding the features that have successfully conveyed realism in previous weather simulators is a crucial step towards the completion of this project.

The weather phenomena will initially be implemented individually and incrementally added to the main system. Meaning, once a component works as intended by itself, it will be implemented into the weather system; when the current version of the weather system works well, the development of the next individual element begins. Tackling one feature at a time, will allow for more efficient testing and bug resolution, as any issues that may arise during the initial stage will be resolved outside of the context of the collective framework.

Once the weather system’s functionality has been fully completed, the issue of integrability and accessibility will be addressed. One of the main motivations of this project is to make weather integration available to developers regardless of their programming skill. Thus, the integration of the weather system into games will be done through a user interface.

# Background Research

## Academic review

*A review of existing academic works. Use Google Scholar, identify papers related to the field. Start broad (eg procedural generation in general) and then narrow down to your particular topic (eg city generation). In particular, state why each paper mentioned is relevant to your project and how it has influenced your decisions*

### Player immersion in video games

There are several experiences that can occur whilst playing digital games. Terms such as engagement (Brockmyer et al., 2009), fun (Huizinga, 2003), flow (Chen, 2007), playability (Bernhaupt et al., 2008), and immersion (Brown and Cairns, 2004) – amongst others – have been used by digital games players, designers, and player researchers to describe the experience of playing digital games. https://core.ac.uk/download/pdf/29030308.pdf

Colloquially, immersion is understood to be the sense of being “in the game” and can be defined as a psychological absorption, where players invest their entire attention, thoughts, and goals into the game as opposed to their surroundings (Sanders and Cairns, 2010). Immersion has also been viewed as a component that hooks in players, and draws them into the gameplay making the activity more engaging (Jennett et al., 2008). Indeed, digital games are known to allow players to lose themselves in a “different” world, to the extent that the players do not notice things around them (Jennett et al., 2008). This feeling is important because it is viewed as a critical factor to game enjoyment, and is one of the outcomes of a positive gaming experience (Jennett et al., 2008). Jennett et al. (2008) state that immersion is affected/influenced by five components when gaming: cognitive involvement, emotional involvement, real-world dissociation, challenge, and control. Its consequences include specific characteristics where gamers are less aware of their surroundings, become more involved with the games, and also experience losing track of time (Haywood and Cairns, 2005). These consequences can be observed when players find themselves absorbed in the game and engaged with gaming session

https://core.ac.uk/download/pdf/29030308.pdf

### Weather impact on mood in general

Weather can play an immense role in cognitively involving the player in the game, but it can also be used as a story telling element. When weather appears in fiction, it is for the purpose of accomplishing certain narrative goals. Albeit being considered part of the world setting, <https://anartfulsequenceofwords.com/2017/04/17/weather-in-writing/> its enigmatic and changeable nature permit it to influence characterisation, plot and more.

In her article about the role weather plays in fiction, [Kathryn Schultz](http://www.newyorker.com/magazine/2015/11/23/writers-in-the-storm) discusses how the perception of weather “mythical to metaphorical”, “with atmospheric conditions…stand[ing] in for the human condition”. https://www.newyorker.com/magazine/2015/11/23/writers-in-the-storm. According to Schultz, such representations may to refer to characters, interactions, or societies.

In *Long Day’s Journey into Night,* a play by Eugene O’Neil, Mary Thorne who struggles with morphine addiction states that she adores the fog for its capacity to conceal the world. Fog is an allusion to the addicted state into which Mary takes refuge from unpleasant realities that she faces. Through much of Richard Wright’s novel *Native Son,* Bigger Thomas finds himself in a snow-covered Chicago, which represents the isolation that he feels as a person of colour living amongst white society.

Symbolism aside, weather in fictional works helps authors create the right atmosphere. The way weather is characterised will have an impact on the elicited emotional response. At the start of *The Storyteller* by Saki, the narrator establishes that it was “hot afternoon” and that “the railway carriage was correspondingly sultry” (129). The reader already feels the irritating air even before learning that the main character will spend a train journey of an hour, trapped in the company of three rambunctious children and their aunt, a woman who is ill adept at making conversation.

Weather was shown to have an impact on human psychology in myriad subtle ways; the reason why this is, not being immediately obvious. The effects of weather on mood may be entirely physiological; excess heat causes discomfort, and this causes irritability, sun exposure produces vitamin D, which promotes serotonin production improving disposition. Cunningham, M. R. (1979). Weather, mood, and helping behavior: Quasi experiments with the sunshine samaritan. *Journal of Personality and Social Psychology, 37*(11), 1947–1956. [https://doi.org/10.1037/0022-3514.37.11.1947](https://psycnet.apa.org/doi/10.1037/0022-3514.37.11.1947)

Another possibility is that the effects of weather on mood also stem from social and psychological aspects. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5918636/Like the weather, our emotions sometimes seem like fickle forces of nature: unstable, enveloping and uncontrollable.

### Weather in video games

Climate reproduction exists in various types of computer games, the degree of complexity varying with the size and budget of the company producing the game.

Weather is a crucial component within Rockstar’s Red Dead Redemption game series. It adds depth and realism to the world environment, while also creating atmosphere. Weather is deployed based on a time and location cycle; dust storms only happen in the arid deserts, while snow only falls in the mountainous regions. Temperature dictates the type of clothing that should be worn and can be accessed by the player via UI (User Interface). Wearing thin clothing in cold environments will lower the player’s health as will wearing thick coats in hot climates.

Smaller studios have also tried raising to the demand of a more realistic experience. Stardew Valley is an open-ended country-life RPG(role-playing game) developed by ConcernedApe featuring weather that changes daily and with the seasons. In Stardew Valley, weather plays a big role as a mechanic, meaning that the current weather has various impact upon gameplay. On a sunny day, the player must water their crops in other for them to grow, while on rainy days there is no need to; during winter crops do not grow at all.

Particle systems

Shaders

Visual Effects (VFX)

## Technology used

*The algorithms and technology used, in detail. If you are implementing an existing algorithm, this is where you describe the algorithm in detail in such a way that a competent programmer could also implement it.*

*It’s okay to have a few references to games articles, but the bulk should be peer reviewed academic papers (journals and conferences)*

### The Unity engine

The Unity game engine is one of the most versatile and widely used game development frameworks, having been installed over 33 billion times in the past year alone.

The engine allows developers to deploy games to over 25 platforms, from Android and PS4 (PlayStation 4), to the lesser-known Windows Mixed Reality or Android TV. Game builds can be iterated from a lead format and ported to any of these platforms, making the creation of multiplatform games effortless. https://www.gamesindustry.biz/articles/2020-01-16-what-is-the-best-game-engine-is-unity-the-right-game-engine-for-you

Unity allows for a very fast iteration

## Findings of the background chapter

*Then summarise the findings of the background chapter, relating to decisions made for your project. You should then refer back to these findings in the Implementation and Evaluation chapters (ie when describing implementation decisions, support them by referring back to findings from the research, similarly when identifying an evaluation method, or describing specific results, support them with evidence from other people’s research, or state that they are different and why that might be)*

Weather’s effect on human emotion is obvious when put into perspective by fictional works.

Beyond its role in ambiance, it can add depth and complexity to the narrative of a game, underscoring the true intentions of the creative minds behind it. Thus, replicating weather faithfully becomes truly necessary to preserve the emotional effect it has on us.

One of the core motivations of this project is to share the weather system and the process of its creation with any who may seek it, regardless of whether they are a seasoned game developer or a complete beginner. Giving its popularity and versatility between platforms, the Unity game engine was deemed as the best fit for the development of this project.

# Implementation

* *In detail, what you did and why you did it. This is your work. If you designed or added to an algorithm talk about that here. If you created a program to implement an existing algorithm, talk about that. Probably you combined various techniques, algorithms and technologies, so talk about how and why you did that.*
* *It is okay to discuss any blind alleys you spent time on – describe that the idea was and why it was abandoned or changed, in the broader context.*
* *As mentioned above refer decisions back to findings from the research chapter that support them.*
* *Code snippets can be useful if they support the text but not a requirement*
* *Screenshots are definitely useful if relevant. Show the progress of the project from early protype to final build.*
* *End with a summary of the implementation and mention how it will be evaluated*

This chapter is an overview of the development process and implementation of the fluid solver. The methods, technologies, and hardware used will be reviewed along with their advantages/disadvantages justified in context. The structure for this chapter will reflect the development process taken during the project. As such, the CPU implementation will be discussed first then the GPU version will be shown

This chapter will follow the design process and development of the weather system. The methods, technologies and algorithms used will be discussed along with their strong and weak points. The structuring of this section will be mirroring the implementation process.

## Production of the weather system

### Setup

The first step in designing the weather system was to decide on an initial set of features to be included. Several case studies were undertaken on various video games to determine the necessary effects that would need to be depicted to create a convincing illusion of weather. Some obvious candidates were first selected: c, clouds, snow, fog, and lighting (ambient and volumetric).

Implementation began with creating a basic particle system.

It is possible to create biomes, or climate zones, by setting up weather triggers.

The manager makes it possible for the user to modify weather

### Implementation

The weather system works by generating a set of boxed effects that are following the player. Therefore, the player camera must be assigned in the

Unity features a robust Particle System where you can simulate moving liquids, smoke, clouds, flames, magic spells, and a whole slew of other effects. Unity features a robust Particle System where you can simulate moving liquids, smoke, clouds, flames, magic spells, and a whole slew of other effects.

#### Rain

During real rainfall, larger droplets appear as elongated cylindrical shapes, while smaller droplets are swept in spurts by the wind, creating a drizzle. This becomes more apparent the heavier the rainfall. They also tend to create ripples or splashes when they hit a surface. All these aspects were replicated by using multiple particle systems.

Firstly, a box shaped particle system was created. The particles were made to fall downwards by adding a varying negative Y value to the *Velocity over Lifetime* modifier. This can also be done by enabling *Gravity to* affect the particle system. While this approach might have been useful for physics-based games, to make the raindrops fall at the expected speed, the particle system would have had to be set up higher on the y axis, which in turn would call for a longer particle lifetime. As this would impact performance, the first method was chosen for implementation.

To replicate larger raindrops, a texture of a cylindrical singular shape was created and applied to the particle system. It was rendered in the stretched billboard mode, meaning that the particles will be pointing in the direction of movement. This will make the rain appear that it is moving with the wind, a crucial aspect to the overall illusion.

To further mimic the appearance of rain, the *3D Start Size, Opacity* and *Emission Rate* modifiers were altered accordingly. The 3D Start Size variable, as the name might imply, controls the three-dimensional size of the individual particles. As real raindrops have slightly different sizes, this variable was set to output varying sized particles. The *Opacity* modifier, which regulates the transparency of the particles, was lowered. The *Emission Rate* is the rate at which new particles are created; regarding the rain effect, it affects how heavy the rain appears to be.

The steps undertaken so far had created a very simple rain effect. To further enhance realism, a second particle system was added to replicate drizzle.

The main differing aspect from the first particle system lays in the texture of the particles. A texture of multiple cylindrical shapes was applied to replicate groups of smaller, lighter drops that are more easily swept by the wind.

A third particle system was added to replicate the droplets splashing when hitting a surface. For a fuller, more organic feeling rain, an animated splash texture was created and applied. This time, the texture was rendered in the vertical billboard mode and the *Start Speed* variable was set to zero. This means that the particles will be stretching upwards from the contact surface but will not travel.

When it rains in warmer climates, the higher temperature causes the rain drops to begin evaporating and remain suspended in the atmosphere. <https://www.sciencedirect.com/science/article/abs/pii/S0012825221000283> Another particle system was made to imitate this phenomenon.

#### Clouds

#### Snow

#### Fog

#### Ambient Lighting

#### Volumetric Lighting

Lighting is key factor for making visually appealing scenes. Colloquially known as *God rays* or *light shafts*, volumetric lights are an incredibly versatile effect that will enchant any scenery it is added to. From a technical standpoint, the effect of volumetric lighting is created by allowing light sources to affect volumetric fog.

# Results and Evaluation

* Clearly identify the evaluation technique
* This should be quantitative. Ideally there should be graphs where one parameter is changed and another is measured (eg density of vertices against performance)
* Push the system to breaking point – make it fail, show that on the graph, and describe why it fails at that point and what could be done to address that (and what the downside of that solution may be)

# Conclusions

## Summary

*Overview of the project. How did it go. Refer back to the initial objectives, and the findings of related works – how do your results compare*

## Further work

*Further work, what more could be done in a further project – how could others build on your work*

## Outcome

*How did it turn out? Say in the introduction what the outcome was, how it was measured and how that compares to existing work (ie this isn’t a mystery novel, we want to know how it ends right at the start)*

# References

|  |  |
| --- | --- |
| [1] | “Time and weather changes | Exploring the game's world,” [Online]. Available: https://guides.gamepressure.com/mgs5thephantompain/guide.asp?ID=31505. [Accessed 02 03 2021]. |