

WORCESTER POLYTECHNIC INSTITUTE

Implementing Procedural Narrative and Characters
to
Explore Player - NPC Relationships

by
Miquel Sans Cornet

A project submitted in partial fulfillment of the requirements for the
Degree of Master of Science
In
Interactive Media and Game Development

April 2022

APPROVED:

Prof. Gillian Margaret Smith, PhD, IMGD Department, WPI

Ben Schneider, IMGD Department, WPI

Abstract:

Procedural generation is a growing technological field which is used to generate content algorithmically without the need for human intervention once a content generator has been implemented. A number of video games have used this technique to widen their content variety and replay value in aspects such as level generation¹ or randomized loot. This project aims to study procedural generation in two aspects:

Firstly, it analyses how it can help making Non-Player Characters (NPC) feel more unique by developing a generator that creates characters with enough complexity to develop engaging relationships with the player. If successful, this project could show a potential way to add interesting characters into a video game while reducing the costs of manually having to detail all aspects of their personality and characteristics.

Secondly, this project also aims to evaluate how modular procedural narrative can be affected by such generated NPCs, modifying its content depending on their dynamic characteristics. The analysis on how player interaction with these procedural narrative events is received could give insight on future implementations of rich and replayable stories in video games.

To assess those two goals, this project implements a prototype narrative video game where the player is in control of a pirate crew the members of which they recruit and can freely explore narrative events in a map.

Keywords

Procedural
NPC
Replayability
Narrative
Modular
Engaging
Relationship
Gameplay
Events

Index

1. Introduction.....	7
1.1. Game structure	7
1.1.1. Problem: NPC predictability	7
1.1.2. Research questions	7
1.1.3. Game Scope.....	8
1.1.4. Reasoning behind the use of Unreal	8
1.2. Related video game comps	9
1.2.1. Crusader Kings – NPC relationships and trait modifiers	9
1.2.2. Nemesis system – Shadow of Mordor/War	9
1.2.3. Battle Brothers – Resource Management.....	10
1.2.4. Dungeons and Dragons – Skill checks	10
2. Game Design	11
2.1. Game Systems	11
2.1.1. Tavern.....	11
2.1.2. Map	11
2.1.2.1. Resource management	11
2.1.3.2. Event initialization and interaction	12
2.1.3.3. Officers Assignment	12
2.2. Procedural NPC generation.....	14
2.2.1. Cosmetic Elements.....	14
2.2.1.1. Gender and name.....	14
2.2.1.2. Clothing and facial accessories.....	14
2.2.2. Stats.....	15
2.2.3. Personality traits	15
2.2.4. Liked/Disliked personalities and relationships.....	16
2.2.5. 3D Character visualization.....	17
2.3. Procedural Narrative	18
2.3.1. Event structure.....	18
2.3.1.1. Base elements	19
2.3.1.2. Keywords.....	19
2.3.1.3. Choices	20
2.3.1.4. Event Textures:.....	20
2.3.1.5. C++ narrative processing.....	21
3. Tests	22
3.1 Target test subjects	23

3.2 Methodology.....	23
3.2.1 Introduction.....	23
3.2.2 Initial survey.....	23
3.2.3. Game session.....	23
3.2.4. Final survey.....	23
3.3. Development testing.....	24
3.3.1. Goals.....	24
3.3.2. Survey questions.....	24
3.4. Final testing.....	24
3.4.1. Goals.....	24
3.4.2. Survey questions.....	24
4 Results analysis.....	25
4.1. Initial survey results.....	25
4.1.1. Theory on having engaging relationships with NPCs.....	25
4.1.2. Ways in which to apply procedural generation.....	26
4.2. Development game test results.....	28
4.2.1. Game session notes.....	28
4.2.1.1. Initial signs of attachment in NPC creation.....	28
4.2.1.2. Map experience overly enjoyed.....	28
4.2.1.2. Crew numbers greatly affecting experience.....	28
4.2.1.3. UI modifications.....	29
4.2.2. Development survey results.....	29
4.2.2.1. Game experience cohesion.....	29
4.2.2.2. Prioritizing Improvements.....	29
4.3. Final test results.....	30
4.3.1. Game session notes.....	30
4.3.1.1. More homogenous experience.....	30
4.3.1.2. Playing favorites.....	30
4.3.1.3. Loyalty in NPCs being important to players.....	30
4.3.1.4. Narrative taking center stage.....	30
4.3.2. Final Survey results.....	31
4.3.2.1. Engaging player-NPC relationships.....	31
4.3.2.2. Gameplay elements affecting mental image in narrative.....	32
4.3.2.3. Gameplay elements affecting mental image in narrative.....	32
4.3.2.4. UI clearness.....	33
4.3.3. Comparing Development and Final tests.....	33

4.3.3.1. Comparing procedural predictability	33
4.3.3.2. Comparing overall Enjoyment.....	34
5. Conclusions.....	34
5.1. Initial Survey conclusions	34
5.2. Production goals' success assessment	34
5.3. Procedural Generation Question Conclusions	35
5.3.1. Engaging relationships with NPCs	35
5.3.1.1 Reading initial information.....	35
5.3.1.2. Growth, support and consequence.....	35
5.3.1.3. Stats are not equal – loyalty is key.....	36
5.3.1.4. Chance to die / die too quick	36
5.3.2. Narrative Mechanics	36
5.3.2.1. Making characters feel unique.....	36
5.3.2.2. Modular modifiers make for good replayability	36
5.3.2.3. Having visual information	37
6. Future Work	37
7. Annex.....	38
7.1. Initial test survey questions	38
7.2 Development test – Post game session survey questions	42
7.3 Final phase test – Post game session survey.....	45

1. Introduction

1.1. Game structure

1.1.1. Problem: NPC predictability

There are many videogames which either put the player in command or makes them fight a group of people in an organization. A big limitation in many of these systems is that the Non-Player Characters (NPCs) in said group may feel limited by being identical and disposable, therefore making the player not care much about their fate. Instead, they are seen more like props than characters, their function being mainly to fill an otherwise empty space. This can be enough in some cases, but in others this simplicity can hurt the experience. For example, Assassin's Creed Black Flag's narrative is based on the concept of the player being a pirate captain, having a crew and managing their ship. This fantasy fades however, when in ship-to-ship battles dozens of NPCs do next to no damage despite their animations suggesting the opposite, making the player look like the only competent person there. This is clearly implemented to stop the player from being passive, but in doing so NPCs become much less interesting. The question is, is it possible to have it both ways, having all NPC have interesting characters while not making the experience feel unengaging or passive?

It would be hard to manually create an unlimited number of characters with name, goals, voice acting, etc. but recently there have been some games that have used Procedural Content Generation (PCG) techniques to give some depth to their NPCs, both affecting the relationship they can have with the player and the variety they provide to gameplay. However, in many cases the PCG content is an add-on, with NPCs interacting with the player and other NPCs in a limited capacity and the narrative being the same regardless. In this project, procedural narrative and characters are to take center stage in the core game experience, by having a simulation-centric narrative narrative defined by characters with rules, characteristics, relationships as well as personalities, similar to the paradigm described in by C. Martens².

1.1.2. Research questions

The goal of this project is to create a gaming experience in which procedural characters feel like they are believable and interesting enough to create a bond with the player, for them to develop an engaging relationship. Seeing which gameplay elements help design an effective procedural NPC generator is an important aspect to discern. Additionally, it is an element of interest to assess how the different characteristics between traditional NPC and procedural ones affect how the players interact with these characters, as well as the narrative in which they interact with. This is why these are the two main research questions this project aims to answer:

Q1: What mechanisms allow developers to create NPC generators with the capacity to have engaging relationships with the players?

Q2: What is the mental model of players when interacting with NPCs developed using procedural generation and narrative?

1.1.3. Game Scope

In order to answer those questions, the next step is to establish the scope of the game prototype in which this player experience is to take place.

- Design the data which will define a procedural NPC character. This information will need to be complex enough to allow for meaningful differentiation between the other NPCs. It will have to be determined how their characteristics affect gameplay.
- Implement a Procedural NPC generator that allows the player to endlessly create characters following the previously mentioned design, displays effectively their characteristics and lets them choose from them which are to join their crew.
- Implement procedural narrative events that present the player with choices which can be modified by the crew's configuration and abilities. The NPC characters need to feel reactive and act accordingly to their characteristics. The more the narrative content feels dynamic and receptive to the player's decisions, the more replayable it will feel.
- Implement an overall gaming experience that ties the separated narrative events into a coherent framework, such as being a pirate crew in a ship, free to navigate the map as the player's sees fit, as long as they have the resources for it.

1.1.4. Reasoning behind the use of Unreal

This project is developed using Unreal as game engine. There are two main reasons why it was decided to use Unreal as the framework:

Firstly, Unreal has a very extensive User Interface API in the form of Widgets, with a lot of functions that help modify text based on game state variables with little to no intensive programming. In a project with a lot of work needed in UI, these functionalities could really make the development process easier.

Secondly, it was my personal intention to learn more about C++ and to implement a project that used that language. Additionally, it made sense to divide front-end functionalities such as UI and player input using Blueprints with C++ back-end calculations, such as developing the procedural narrative by adding game state variables or generating procedural character stats.

1.2. Related video game comps

1.2.1. Crusader Kings III – NPC relationships and trait modifiers

In Crusader Kings III³ this game there are many ways to use your character: religious acts, warfare, plots to murder or seduce, administration, etc. and there is not a particular goal or order in which to do activities. However, the result of each depends on the stats of the character, its modifiers and the state of the game, making each playthrough different.

Character's traits and abilities determine success chance and options given a particular challenge. Every character has an "opinion" on the rest of them, which can also influence actions regarding them. Additionally, the more repetition there is on a task the more suited for it a character is, which in the end creates a system where initially every character is more or less a blank slate, which from there a playstyle is favored at the expense of others.

Takeaways:

- Simulation sandbox feel, not many explicit objectives
- Relationships between all characters, traits affecting reactions
- Modifying characters through choices and dilemmas in events



1.2.2. Nemesis system – Shadow of Mordor/War

The procedural NPCs in Shadow of War⁴ have their own strengths and weaknesses, forcing the player to fight them in different ways to defeat them, using different kinds of attacks and mechanics. The idea that weaknesses and strengths directly affect the effectiveness of an NPC in concrete situations is also an inspiration in this project, since one of the main goals is to make each character feel unique and give the player choices on how to use their crewmen.

Additionally, what the NPCs say also depends of their personality (example: a drunk orc may slur their words), fleshing out more their personalities. They will also react to player's actions in different ways, further developing their relationship with them (example: an obsessed will moan about longingly missing the player after they fled their previous encounter).

Takeaways:

- NPCs with distinct personalities
- Reactions to player actions
- Gameplay modified by traits



1.2.3. Battle Brothers – Resource Management

Battle Brothers⁵, a turn-based strategy game, has an open world in which the player can recruit NPCs into their mercenary company. As such, they can go where they please as long as there are enough resources to keep them paid and happy. This gives the player a lot of agency and organic narrative, allowing them to get resources by engaging in different activities such as accepting contracts, exploring map locations, trading, robbing civilians, etc.

Additionally, relatively simple events spontaneously get triggered from time to time, with choices based on the NPC's traits. The combination of player freedom when deciding on what activities to engage in combined with narrative events create a very replayable experience and narrative which greatly depends on the players actions.

Takeaways:

- Feasible event UI quality
- Map exploration
- Events based on traits



1.2.4. Dungeons and Dragons – Skill checks

D&D's whole gameplay consists on narrative events narrated by the Game Master in which the players can make decisions and interact with characters. It is one of the classics of tabletop roleplaying. The way characters are defined with stats, how their actions succeed or don't and how the consequences are applied are elements from which this project takes inspiration.

For example, the way a character is mechanically differentiated from another is based on which stats are high and which are low, creating different archetypes or classes in character sheets. Generally, all characters have a similar number of skill points allocated in different stats to keep the classes balanced. Skill checks allow a player to take actions which can succeed or fail by making a skill check. A skill check consists of a random roll die modified by a relevant skill, (ex: strength check could to try to break a wall). This approach is interesting because it combines the excitement of uncertainty while maintaining some player control by being able to hedge the odds in their favor with the right NPC's abilities.

Takeaways:

- Stats-based NPC builds
- Interactable narrative events
- Skill checks



2. Game Design

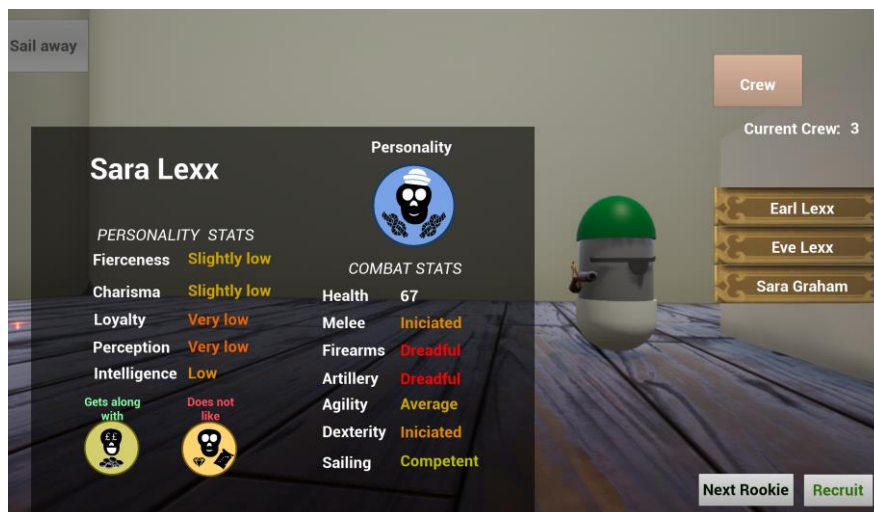
2.1. Game Systems

2.1.1. Tavern

The player initially finds themselves in the interior of a building representing the tavern in which they can swap through characters and choose to recruit those that they prefer into their crew. The stats and characteristics of the NPC can be read with ease by using color-coding and icons.

It is also possible to check the characteristics of the crew members like their personality trait, and which personalities they like and dislike. Aside from the stats information, a 3D representation of the pirate is shown so that the player can relate the name and character information to a 3D character.

Once recruited, a dropdown menu appears on the right, allowing the player to inspect the new crewmembers, which is useful when wanting to try to get a new recruit that balances out the weaknesses of the already existing crewmates. Once at least eight characters have been recruited, it is possible to click on the “Sail away” button in order to leave the tavern with the newly acquired crew



Tavern scene

2.1.2. Map

The map is where the player can decide where their ship is to sail next, and from where they can manage their crew. Procedurally generated events appear in the form of markers, some of which can move while others remain in place. By clicking at a point in the sea, the ship moves to that position. Time is represented by having a day counter, each day passing after a fixed number of seconds.

2.1.2.1. Resource management

As shown in the figure below, the top left corner of the screen is dedicated to displaying the crew's resources, them being food, gold, wood, gunpowder and rum. Following the game experience detailed in section 1.a.iii, the gold and bread resources drop each day proportionally to the number of people in the crew. While game balance was not a priority in this project, the

concept of having to balance having a powerful crew and not spending too many resources was something intended to be visible for the player in a prototype fashion.



Map scene

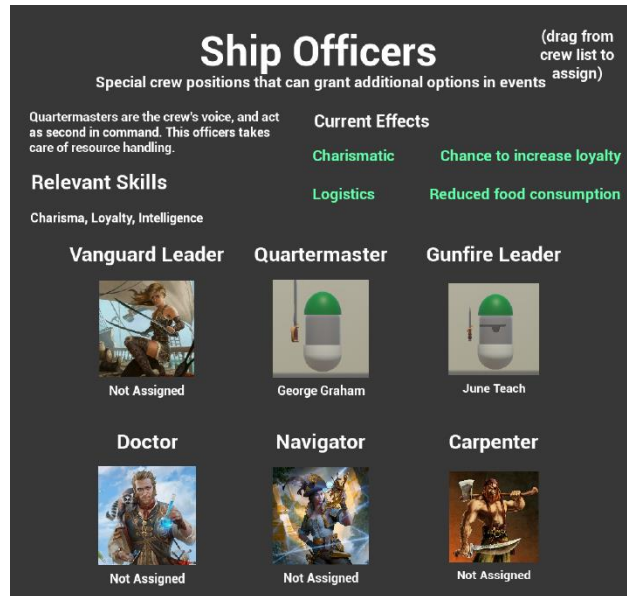
2.1.3.2. Event initialization and interaction

The intention of this project is that in each playthrough the event distribution is randomly different. Initially, event markers are scattered around the map with only some parameters to constrain the kind of events they can represent, such as whether the event has to happen on land or at sea, or whether it has to be static. At the start of the map scene all map event markers assign themselves with an event corresponding with those parameters.

2.1.3.3. Officers Assignment

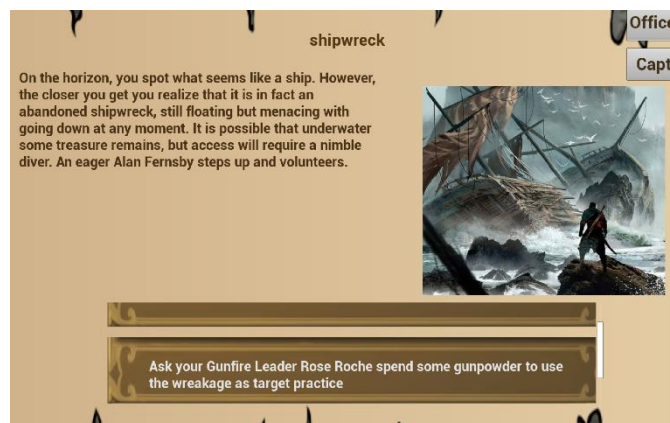
The crew can be further managed by assigning crewmen to particular positions. These special positions are called officers, and they provide additional options in narrative events that require some expert in a particular area like navigation or artillery to be able to help, granting additional choices in some related narrative events. By giving them titles and concrete tasks that require specific abilities, crewmen can feel more unique and useful, and further develop the mental narrative that the players have about the NPC's trajectory in the crew. On the Officer's window the player can see all the information by hovering on each of the officer positions. Each one will showcase:

- A brief description of the position and why it is important to the crew
- A list of relevant skills. Crewmembers with expertise in those skills will make a better candidate for that particular officer position.
- A list of beneficial effects on the crew by having the position filled. Initially, there was the intention to have gameplay bonuses or penalties depending on the skill of the officer (i.e. increased movement if the player assigned a good navigator, decreased if they were an ill-suited one). However, because of scope reasons, those effects were not implemented. However, the effect descriptions were kept in order to give the playtesters a feeling of the prototype's intentions.



Officer's window

As an example of the officers' effect on the narrative, in the figure below it is possible to see how a new option emerges because of the crew having a Gunfire Leader, allowing him to use the shipwreck as target practice for the men creating a new narrative branch. The way this modular narrative works is further discussed in section 2.3.



Officer exclusive narrative choice

2.2. Procedural NPC generation

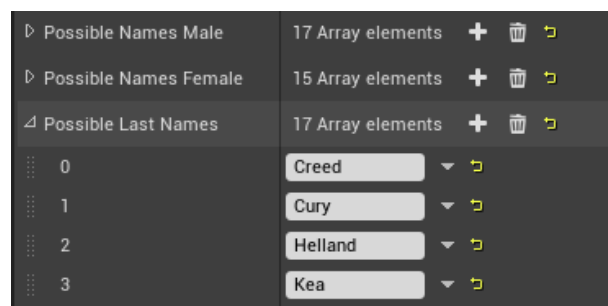
Pirate crewmembers can be recruited in the tavern scene, where all their information is displayed, as well as a random cosmetic generation is implemented.

2.2.1. Cosmetic Elements

When tasking the procedural generation system with the creation of an NPC, there are several steps involved:

2.2.1.1. Gender and name

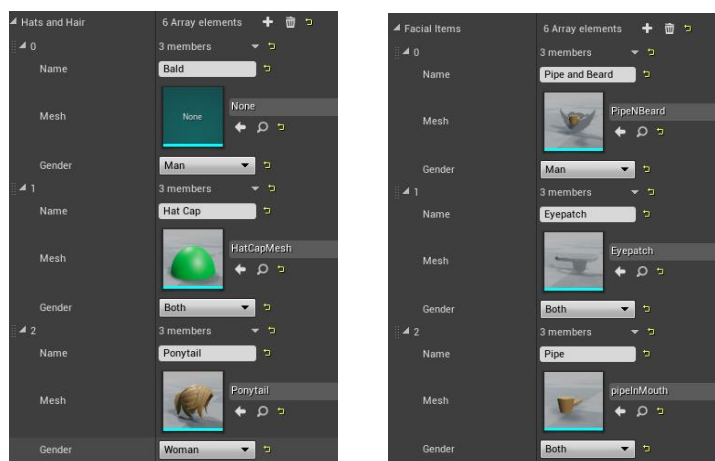
The character is named based on randomly assigning its gender, then choosing a random name from the corresponding pool, followed by a random last name, shared between the two genders.



2.2.1.2. Clothing and facial accessories

There is a level of visual customization of the pirate, based on randomly choosing an item to set as hat/hair, a facial item and a weapon. This visual customization, while light, allows for the pirates to have an easier time making a visual image of the character and not have it just described by text. Although initially the weapon kind affected combat when a real time combat system was introduced, since the final product focused on the narrative experience, weapons became just another cosmetic item.

The system is able to modularly add and remove cosmetic items into such as hats, hair and facial items, which can be assigned to male, female or both genders. Each item had a 3D mesh and a name assigned. After having created the tool, it is really easy to add and remove elements, the hardest part would be to create the 3D mesh.



2.2.2. Stats

One of the main ways in which characters differentiate from each other is the stat distribution. This is what dictates the strengths and weaknesses of the character, which in time affects how the players assign them roles they deem fit for their profile. As it can be observed in the figures below, the stats values are segmented in tiers, nine to be exact, and those tiers are color coded to give a better idea to the player of how good or bad they are, even with just a glance on the coloring. Additionally, instead of just being represented by numbers, the proficiency in an ability is determined by self-explanatory tiers. This makes readability of a large variety of abilities much easier.

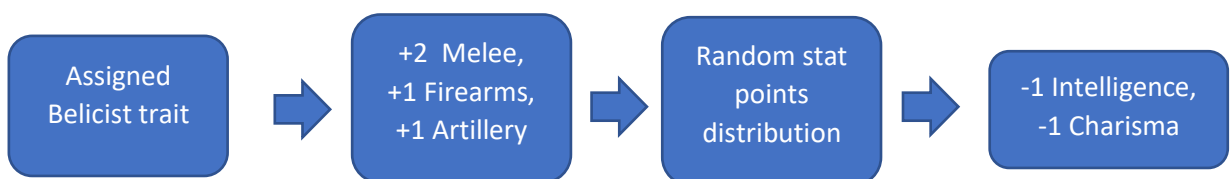
PERSONALITY STATS		COMBAT STATS	
Fierceness	Low	Health	69
Charisma	Abysmal	Melee	Average
Loyalty	Low	Firearms	Beginner
Perception	Very low	Artillery	Beginner
Intelligence	Low	Agility	Beginner
		Dexterity	Average
		Sailing	Average

The generation of those stat values is not completely random, because that amount of variability could result in characters having very unbalanced stats configurations could technically make it possible for an NPC to have all stats to max or to zero. To fix that an approach similar to D&D was taken, in which initially a generated character has zero points in all stats and then a fixed number points are distributed randomly between the stats, ensuring the NPC's variability in configurations while keeping balance between them.

2.2.3. Personality traits

Aside from the stats, there is another NPC characteristic that modifies their behavior and abilities in narrative events, which is their personality trait. All characters have a random trait assigned from a pool which can give them some positive stat bonuses before the previously random distribution and/or some negative ones after. Each trait is represented by an icon, which when hovering on it give a description of its meaning, and in case that the personality icon is hovered, it also gives a description of the stat bonuses.

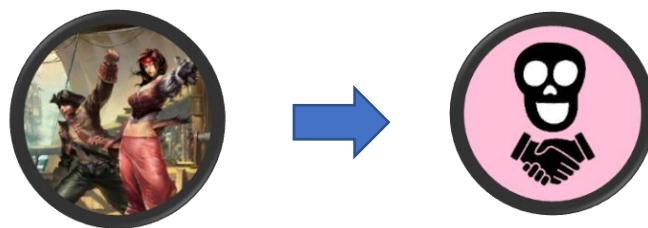
Example of personality trait assignment in procedural generation:



This is a table with the possible personality traits:

Adventurer	+1 Fierceness, +1 Perception, +1 Agility, -1 Dexterity
Drunk	+2 Melee, +1 Fierceness, +1 Loyalty, -1 Ranged, -1 Charisma
Coward	+1 Agility, +1 Artillery, +1 Thievery, +1 Firearms, -1 Fierceness, -1 Melee
SeasonedSailor	+2 Sailing, +1 Perception, -1 Artillery
Friendly	+2 Charisma, +1 Loyalty, -1 Thievery
Belicist	+2 Melee, +1 Ranged, +1 Artillery, -1 Intelligence, -1 Charisma
Greedy	+2 Thievery, +1 Intelligence, +1 Loyalty, -1 Charisma
Rogue	+2 Thievery, +2 Agility, +1 Melee, -2 Loyalty, -1 Fierceness
Gentleman	+2 Charisma, +1 Intelligence, -1 Fierceness
Sharpshooter	+2 Firearms, +1 Artillery, +1 Perception, -2 Melee

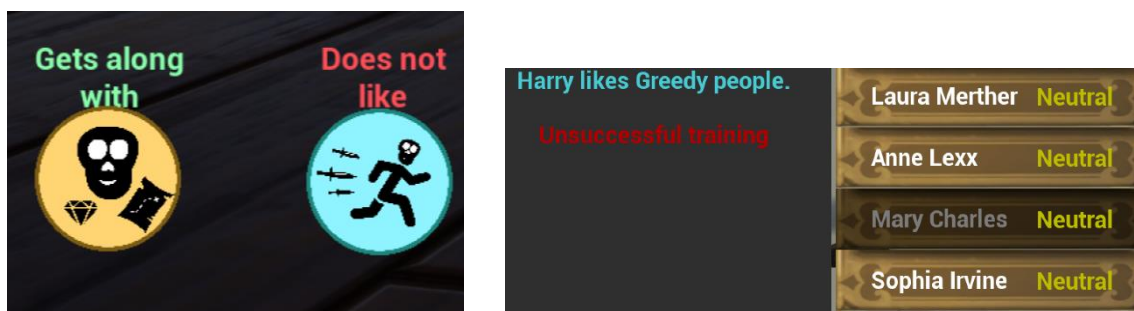
Initially personality trait's icons shown in the UI were created using artistic illustrations found online with the intention to represent the idea while being visually engaging. However, some development testers found them somewhat confusing, both because the images were too complex to analyze in a small icon and because the gender of characters in the illustration may not match that of the NPC. As it can be seen in the image below, this led to the introduction of simplified icons that conveyed the same message.



2.2.4. Liked/Disliked personalities and relationships

On top of the personality trait assigned to modify the NPC's stats, two more are randomly drawn to be assigned to the liked and disliked traits. These preferences affect how the relationships between the NPCs start as, giving the NPCs with the liked personalities a small bonus while the disliked get a small penalty, while keeping the rest neutral.

All crewmen have an opinion about each other, basically a relationship that gets developed over time, with multiple narrative events being able to affect them, such as having a crewman warn another about a plot that is being carried out against them or successfully training another crewman in a particular skill. It is possible to see the state of these relationships, also being color-coded into tiers and having a record of the events that have affected each particular relationship.



2.2.5. 3D Character visualization

Once the internal information such as name, gender, stats and personality traits are chosen, those are visualized on a 3D capsule-like body to represent the generated pirate. The reason for this game to have 3D character initially was the implementation of a real-time combat system that would be combined with the procedural narrative bits.

However, despite some initial progress was done in that direction, the focus was put solely on the narrative aspect in order to give it more depth, therefore dropping the real time combat aspect of the game experience. However, the 3D character creation was kept, allowing for portraits which could be shown when choosing a pirate. This way, the NPCs were not only differentiated by text, but also with a portrait of themselves.

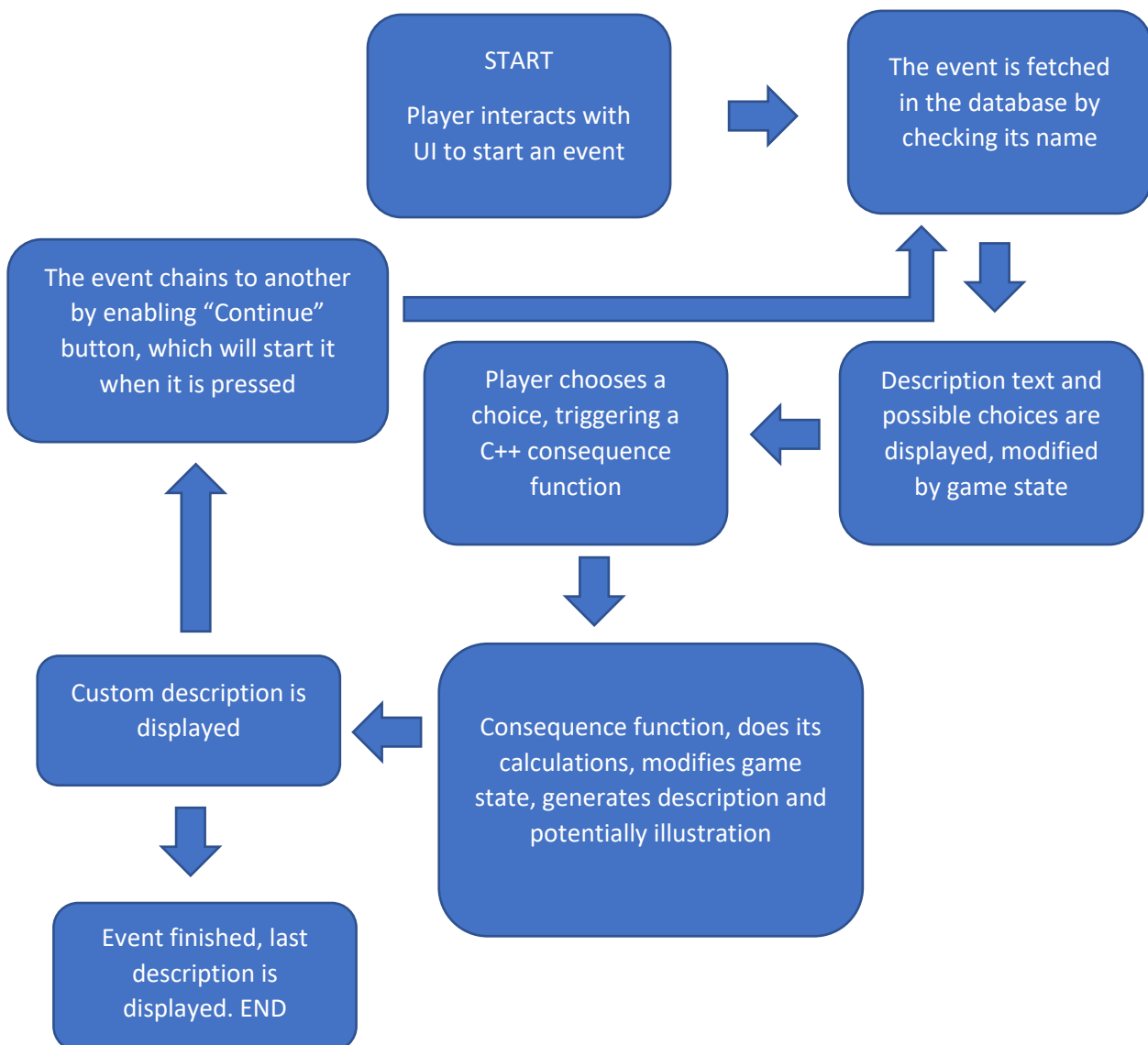


2.3. Procedural Narrative

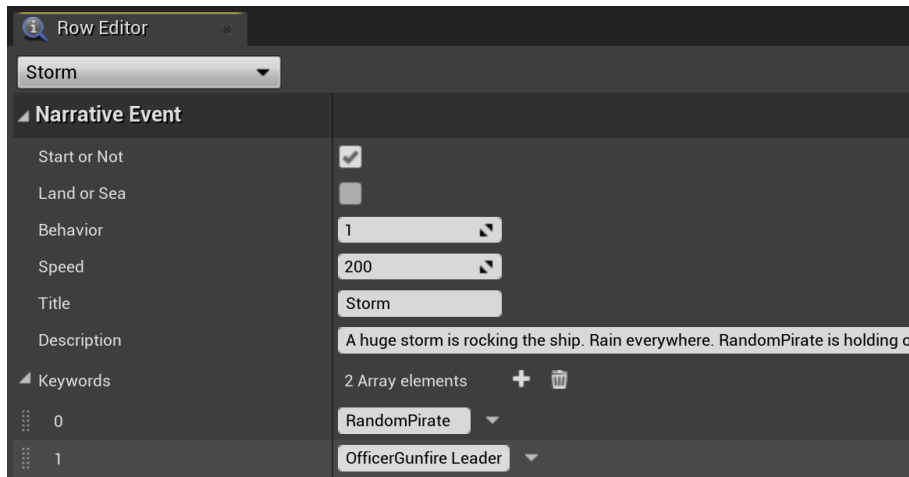
Most of the action that happens in the pirate crew is based on narrative events. There are two main types: captain actions and map events.

2.3.1. Event structure

This diagram helps exemplifying the interaction in a narrative event works:



2.3.1.1. Base elements



Start or Not – This Boolean variable determines whether or not this event is the start of a chain of events. This way, it is possible to have complex situations combining multiple events while being sure that only those events which have a *true* value set in this variable will be eligible to be assigned in the map events at the beginning of the game. Additionally, in order to be able to store information during the course of chained events, this information is not reseted when starting an event with a *false* StartOrNot value.

Behavior and Speed – Map events have the ability to show different behaviors, such as remaining still, moving randomly, fleeing when close to the player and pursuing the player. This way, events can act in a way coherent with their content. For example, events such as shipwrecks and temples have behavior 0, which keeps them in place, while a storm has behavior 1, which make it move randomly across the map at speed 200.

Title, Description – The event’s setting is described by the title and description content, which is shown in their corresponding sections in the UI.

2.3.1.2. Keywords

While most of the description text is fixed, it is possible to combine it with dynamic text processed in C++ by using keywords. Keywords represent any particular variable of the state of the game. They are stated in the keywords array and then before showing the UI to the player, the program iterates through each element in Keywords, fetches the corresponding game information and then substitutes the keyword with the same name in the description with the obtained data. For example, for the “RandomPirate” keyword, the algorithm will select randomly a pirate from the crew and return its name. Keywords are kept during the whole event, allowing to modify choice descriptions, as well as storing them for further use in a chained sequence of events.

Examples of Keywords used in the project:

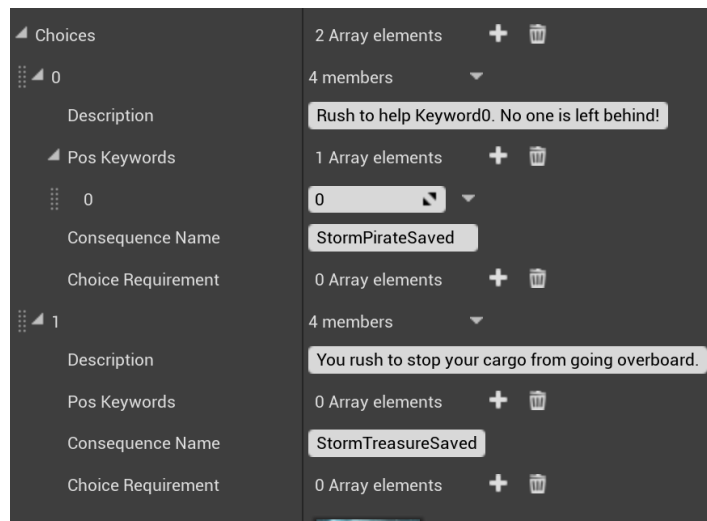
- RandomPirate -> Choose a random pirate.
- OfficerXXX -> Find the name of the officer with XXX position, or return “None”.
- EventLocation -> Return the name of the location where the event is happening.
- RandomTraitPirateXXX -> Return a random pirate from the subset that have the personality trat XXX, or a random one if there is none.
- PirateTargetOrder -> The name of the pirate that was chosen as target for a plot

A great element of keywords is the fact that they are completely customizable. If other events needed to fetch other information, it would be as simple as adding the C++ function to return that element to substitute a predetermined keyword.

2.3.1.3. Choices

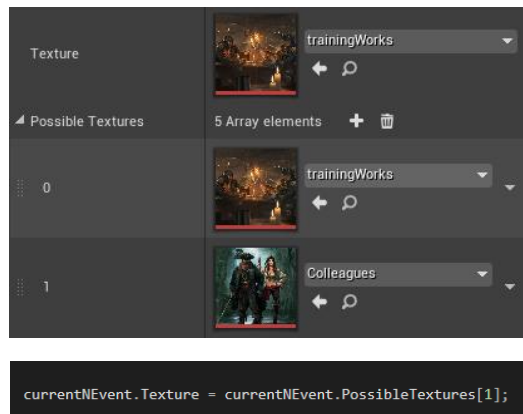
After being presented with the event description, the player can choose from an array of choices to choose from, each one with a description which can be modified similarly to the event's description, by replacing any word starting with "Keyword" by the corresponding keyword from the event's array described above. Once a choice is made, a C++ function is called using a custom reflection method, which allows to execute a function by sending its name as a parameter. This way, each choice can have its own custom behavior, modifying any relevant game state information. Finally, a choice will only be shown to the player if the requirements for it are fulfilled. Such requirements can be:

- An officer position is filled
- There is a minimum amount of a resource available
- If a particular skill check has succeeded



2.3.1.4. Event Textures:

To give some more visual representation to the narrative event that is unfolding, each event has a texture and an array of possible textures. This way, not only an illustration can accompany the narrative text, but this texture can change by picking one or another from the possible texture pool in the C++ consequence functions, as it can be seen below, allowing for the visual story to also change based on its dynamic evolution.



2.3.1.5. C++ narrative processing

Combining fixed text with variables - Once a choice is made and a C++ consequence function is called, the algorithm executed the calculations that such function requires, and elaborates a text that will be shown as the following event description to the player.

An example of this processing can be seen in the figure below, where after a successful training, the function to increase a stat is sent to the Global Information object by sending the name of the pirate as a parameter, and then a static text is combined with variables such as the name of the trainee, the skill they have successfully trained and its current value, while internally removing the resources that were used for that learning.

```
int aux = GI->ModifyStat(pirateToGiveOrderTo, statToTrain, 1);
currentNEvent.Description.Append(".\n\n ");
currentNEvent.Description.Append(pirateToGiveOrderTo);
currentNEvent.Description.Append("'s ");
currentNEvent.Description.Append(skillToTrain);
currentNEvent.Description.Append(" is now ");
currentNEvent.Description.Append(GI->PersonalityStatsNames[aux]);
currentNEvent.Description.Append("\n 5 rum and 5 gold spent. ");
GI->Ship.alcohol -= 5;
GI->Ship.gold -= 5;
```

Game-state dependent narrative snippets – Additionally to the scripted text, there are modular details that can be hidden or displayed based on conditions being met, with the overall narrative making sense regardless of whether these snippets are added or not. This way it is possible to make the NPC characters feel more believable and reactive by showing how player decisions relevant to their character affect them.

Example 1: If the player intends to explore a temple in ruins, those crewmen with the personality trait “Adventurer” react to that fact, by presenting themselves as volunteers to enter the dangerous place, as well as improving their loyalty, as you can see in the following figure.

The adventurers in your crew gets really excited about this expedition!
Harry Charles's loyalty is now Average.
Sam Irvine's loyalty is now Sightly high.

Example 2: When in a duel, a Sharpshooter pirate will have a bigger chance of success when shooting a pistol. When the firearms skill check is made, there is a small modifier giving an edge.

Being a Sharpshooter, George has some advantage at this part of the duel.

Additionally, it is important that the players can understand the rules and inner workings of the game so that they can feel like they are in control and that they can strategize around those rules. This is why when a Skill check is made, the outcome is stated before narrating the consequences, so that the player can trace back the reason for an event to play out in a particular way.

(Loyalty check failed) Sophia's loyalty is to the whole crew, and the fact that you would plot to have one of your own murdered is too much. They refuse and disclose your plans to the whole crew!

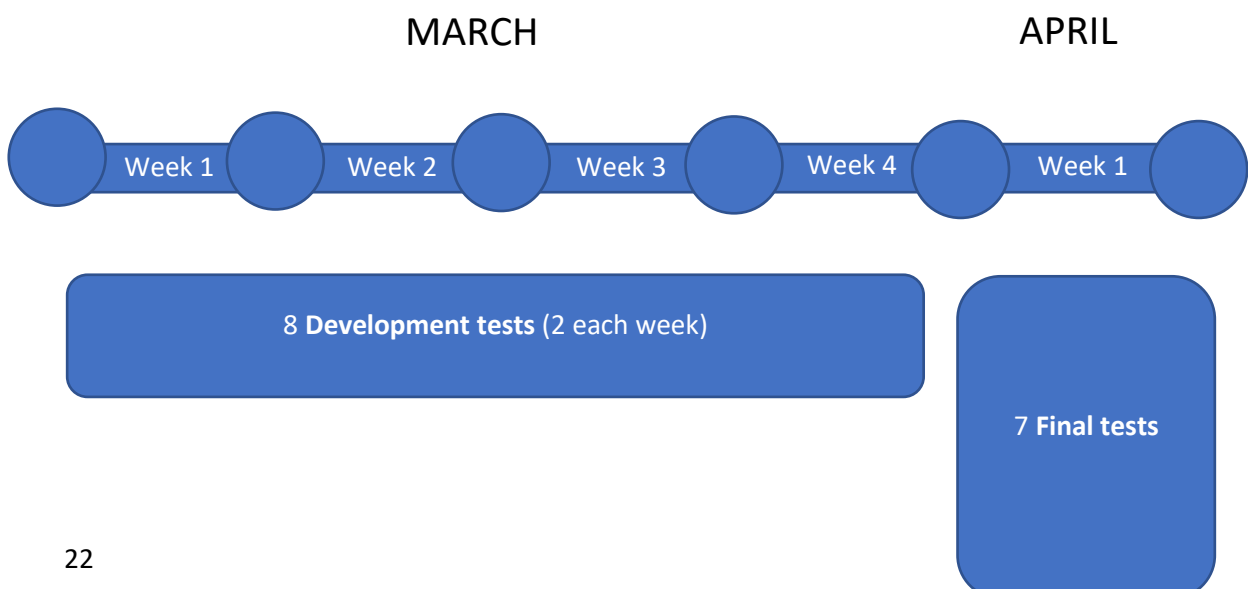
Concluding the text processing - At the end of the processing, the C++ consequence function can call two blueprint functions to update the UI:

- Continue the event, which will set the “Continue” event button to call the next event in the chain when pressed and finish the event, which will call the narrative event in the parameters when pressed.
- End the event, which will enable the “EndButton”, show the final text description.

3. Tests

During the course of this project, fifteen tests were carried out over approximately five weeks. They were done in two test phases, differentiated both by their set of goals and survey questions, although the methodology and target test subjects were essentially the same.

The first phase consisted of eight development tests during March, phased out in two tests every week. Development tests were carried out to generally assess the gaming experience and to better prioritize improvements during the last weeks of development. The second phase featured seven final tests on the first week of April, where without further modifying the project build, the research goals were assessed.



3.1 Target test subjects

In both test phases the test subjects were Game Development college students or recent graduates (both from WPI and other universities) in order for them to have an easier time understanding and having an opinion on the project's goals concerning relatively complex concepts like procedural generation, NPCs and narrative, as well as game flow and player's experience.

3.2 Methodology

In both phases, the test is divided in three parts: initial survey, game session and final survey.

3.2.1 Introduction

Before the initial survey, the test subject is greeted, given the link to download the project's build, and the consent form with an explanations of the project's characteristics, goals and methodology. After answering any questions the subject may have and clearly stating that they can ask any further question at any moment, the test subject is given the link to the initial survey while the game downloads.

3.2.2 Initial survey

The test subject proceeds to fill out the initial survey, which is the same for both test phases, giving a general feel for the player's interests concerning games, as well as their initial opinion on the project's goals and theoretical assumptions. This way, for example, if the test subject feels it is at least theoretically possible to have engaging relationship with procedural NPCs but in the final survey they did not have that experience, it is clear the project's implementation is responsible for that.

3.2.3. Game session

The test subject starts playing the game, being encouraged to voice his train of thought out loud. Especially in the test development phase, they are encouraged to voice out improvement suggestions, etc. while in the final phase their experience is where the focus is. In both cases, the investigator takes notes while the test subject plays, and answers any question the test subject may have. The test subject can play for as long as they want, there is no time limit to the game session, but since there is no win/lose condition either, the investigator can let the test subject know when all playable content has been explored.

3.2.4. Final survey

Once finished the game session, the investigator shares the link to the final survey with the test subject, which allows them to share their gaming experience and how much the research goals of this projects have been attained. With that, the test concludes.

3.3. Development testing

These tests have been carried out during the four last weeks of development, once all basic tool implementation was finished and content was starting to be introduced, therefore an initial gaming experience was starting to exist.

3.3.1. Goals

By having some tests with some time to make corrections based on them, they could allow to:

- Determine the success of the recent additions to the game
- Ensure the cohesion between the different elements of the game experience
- Single out the elements of the game that were not interesting/fun
- Find possible bugs during the game experience
- Gather ideas of possible improvements and new additions
- Prioritize the importance of enhancements during the following weeks

3.3.2. Survey questions

- What is the one thing to keep no matter what and the one to remove
- How cohesive the different game systems felt in the experience as a whole
- How long did it take for procedural NPCs and narrative to feel repetitive
- An enquiry about which improvements should take more priority, from possibilities like enhance narrative events, improve NPCs, polish bugs, make new features, etc.
- Assess how fun the overall experience was

3.4. Final testing

Once the prototype game is finished, a final round of playtests is conducted. These playtests include an initial survey, a game session and a post-game survey with these goals and questions.

3.4.1. Goals

- Determine how fun/immersive the experience as a whole was
- Determine the quality of the characters generated by the PGC⁹ system
- Determine how engaging the relationship with the characters were
- Evaluate how the differences between characters based on traits affect the player's gameplay choices.

3.4.2. Survey questions

- How long did it take for procedural NPCs and narrative to feel repetitive
- How much did the player have engaging relationships with the NPCs
- Mark those gameplay elements most relevant to develop a relationship with NPCs
- How immersive the procedural narrative was
- How much did the player extrapolate details and stories about the NPCs in their heads
- Evaluate the importance and effectiveness of various elements of the procedural narrative
- Evaluate the UI's effectiveness in conveying information
- Rate the enjoyment of the experience as a whole
- Differentiate which possible improvements would be most interesting as future work
- Any additional comments they may have

4 Results analysis

Once all the test are concluded, here are the highlights of what these experiences brought, which will be used in the following section to develop conclusions.

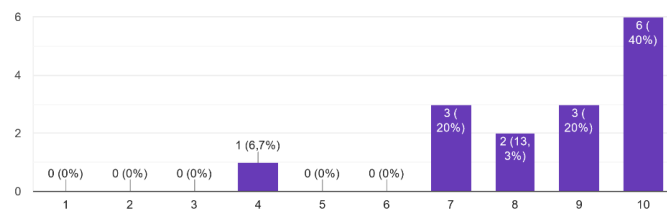
4.1. Initial survey results

Since both development tests and final tests had the same initial survey questions and their answers were not influenced by the game session, their results are joint in one.

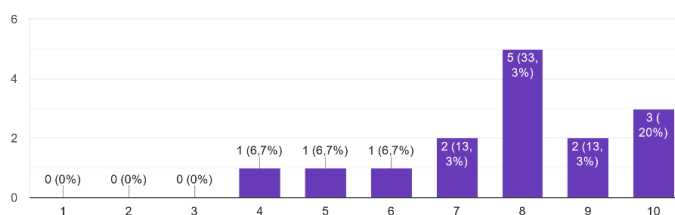
4.1.1. Theory on having engaging relationships with NPCs

Two questions were asked concerning engagement in characters and their relationship with the player. First no distinction between procedural and no procedural characters were made, and playtesters overwhelmingly agreed that players can have emotional connections with video game NPC characters, with an average scoring of an 8.5 out of 10. On the second question the distinction was made with procedural NPCs and whether they could be engaging, and while there was a larger variety in responses the average was still very positive, with a 7.8 out of 10 in average. From these questions it can asserted that most players believe procedural NPCs can be engaging, although scripter characters are believed to be slightly more engaging in general.

Do you think it is possible to have an emotional connection with an NPC character in a videogame?
15 responses

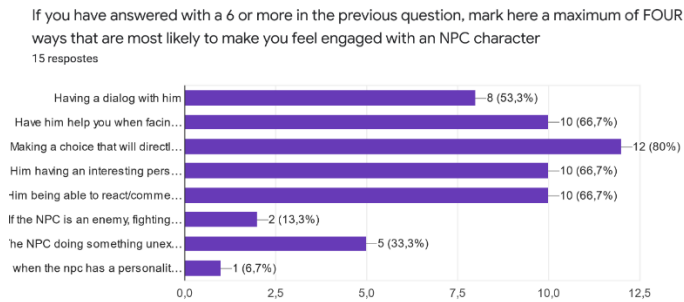


Do you think a procedurally generated NPCs can be engaging characters?
15 responses



Following up these two questions it was also asked to choose a maximum of four elements from a number of potential ways in which the playtester feels they can feel engaged with an NPC character. The choices were limited to four in order to make playtesters choose the most important ones, since potentially all elements could provide some benefit to the engagement between player and NPC, but that would not really provide much information. The intention for this question was inquiring about the most popular interactions between players and NPCs, both to prioritize tasks during development testing and to compare with the preferred NPC characteristics in the final post-game session surveys.

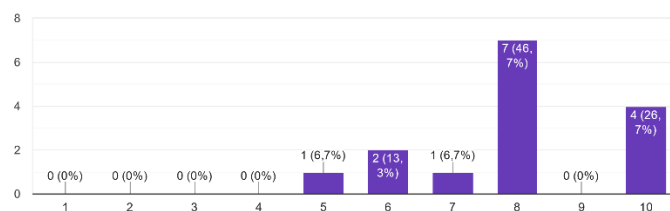
As shown in the figure below, most players felt like making a choice that would affect the NPC's life was the most important way in which they felt engaged with NPCs, basically having control over their actions and future. Additionally, NPCs being able to provide support, react to player choices and having an interesting personality were also chosen by a large majority of the playtesters. Interestingly, the least chosen options were fighting an enemy character, the NPC doing something unexpected and having a dialog with them.



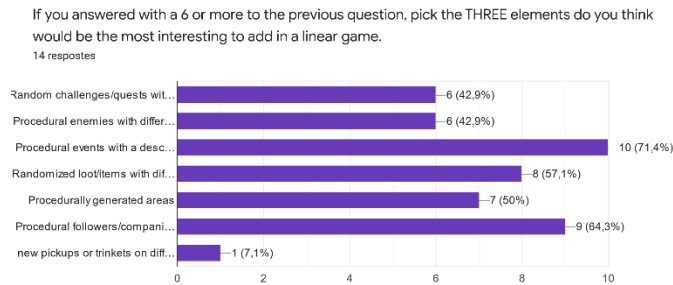
4.1.2. Ways in which to apply procedural generation

After having determined whether procedural NPC characters were interesting at all and exactly how NPCs could be engaging, three questions were posed to discuss the applicability of procedural generation's different techniques in games. Firstly, as shown in the figure below, playtesters agreed that procedural content could benefit replayability in linear games, with an average of 8 out of 10.

Do you think linear games with a fixed story might increase their replayability by adding procedural content?
15 responses



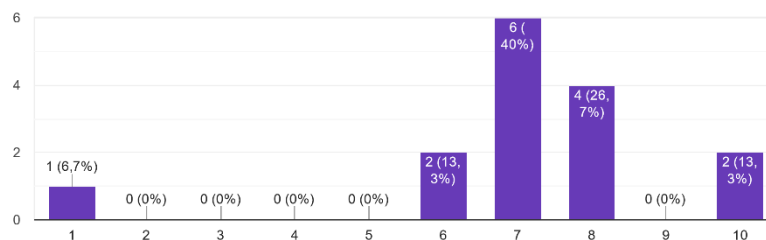
The follow-up question would narrow that affirmation down by giving playtesters the option to choose from a set of procedural application, again, limiting their choices to three, and only if they voted with at least a 6 on their previous question. Interestingly, the most voted choices are the implementation of procedural narrative events and the implementation of procedural companions, which they are the main procedural applications in this project. Additionally, randomized loot items and procedural game levels were also highly chosen, which are also techniques that many games implement. The least chosen elements were procedural quests and enemies, although they were still chosen by almost half the playtesters.



Finally, a more concrete question was posed concerning the internal procedural generation algorithm, on whether randomness should play a large role on generating challenges, or whether past player actions should act as the main parameter. Complete randomness vs. more defined procedural generation and which combination makes for better replayability is a topic that is still very discussed.⁶ Playtesters voted on an in-between while tending more towards non-randomness with an average of 7.1 out of 10, where 1 is completely random and 10 is not random at all. In other words, the average playtester likes the game to tailor the challenge to the player's choices and game-state, but they don't want that to be the sole factor, presumably because that would make the whole system quite predictable.

If a game had a procedural generative system to create enemies, would you prefer complete randomness in their characteristics or would you prefer number of archers to favor a change of tactics.

15 responses



4.2. Development game test results

4.2.1. Game session notes

These tests were done while in production, therefore making each playthrough very different because the content was changing drastically each week and bugs were being fixed continuously. However, some qualitative notes can be made concerning how the game prototype was received during development tests.

4.2.1.1. Initial signs of attachment in NPC creation

Especially in during the first two weeks of development testing there was not a lot of narrative content, since it was still being refined and implemented. However, this did not stop playtesters from starting to react and engage with the character that the generator was creating in the tavern scene. Additionally, they would spend a lot of time going through potential recruit candidates, comparing their stats with the already recruited crewmembers in order to try to end up with a balanced crew. This is a clear sign that playtesters could see and understand the differences between the NPC and considering them important enough to take them into account when choosing whether to recruit or not.

Only by reading their strengths and weaknesses, as well as the personality traits and preferences, players were starting to create a mental image of different characters in their head, making assumptions and judgements such as “I don’t trust this guy”, “I want this pirate in my crew” or “It makes sense that X character is good at Y given that their personality is Z”, just to name a few. It was clear that those small data details were getting processed by the players and turned into a personality by filling the blanks in between. This is why in the second half of the development test as well as in the final testing phase a question was included to measure how long this initial attachment faded and the procedural system started to feel predictable.

4.2.1.2. Map experience overly enjoyed

Several playtesters enjoyed the feel on the map interaction, where the player can click on sea areas, making the ship figurine to move towards it, avoiding the land obstacles. The fact that the light casted a shadow of the ship on the map was appreciated, since it gave the overall feeling that indeed a physical figurine was moving on a map.

Seeing events being able to move following different behaviors was also enjoyed, since it gave the activities a much more dynamic feel as well as gave the players the small challenge of having to click correctly to intercept those events they wanted to interact with.

4.2.1.2. Crew numbers greatly affecting experience

One of the main factors that determined how long did it take for a playtester to get bored or feel like they had seen it all was how many crewmembers they had recruited in the tavern scene. This is so because most narrative events have modifiers that depend on having an NPC in the crew with a specific personality trait, skill or officer position. This is why with higher numbers the odds of having more modifiers appearing in events increased, therefore give the player more interesting content. Additionally, in some cases where playtesters had only chosen recruited three or four crewmembers, crewmen deaths meant they basically would run out of crewmembers to interact with.

For this reason, in the final test a modification was made in the tavern scene where players were only allowed to set sail after having recruited at least eight crewmen, so that all of them had a considerable chance of seeing at least some modified text events.

4.2.1.3. UI modifications

Multiple improvements were applied to the UI to make the experience more accessible to the players, despite not all of the remarks being able to be implemented in time. Some of them were:

- The inclusion of short explanations of the controls in several screens
- The change in style of the personality icons
- The removal of the personality trait stats modifier description in liked/disliked personalities section, to better differentiate from the actual modifications coming from the NPC personality section

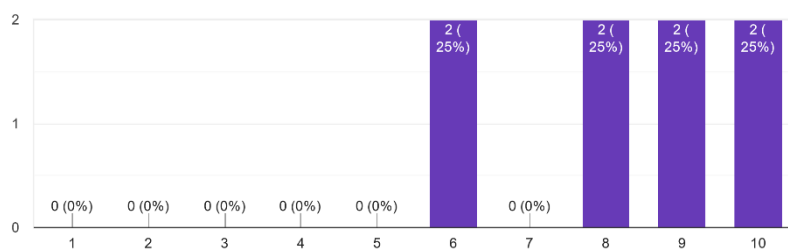
4.2.2. Development survey results

On top of many qualitative comments that gave an approximate idea of how the game experience was received by the players, there are a number of very clear results that can be taken from the scored questions in the survey.

4.2.2.1. Game experience cohesion

This project has many different game systems that are supposed to work together to give the player the experience of running a pirate crew with interesting characters. From recruiting randomly generated pirates, to moving the ship around the map, interacting with narrative events, checking out the different NPC information in the crew section and assigning officers, it was a real risk that the game could feel more like a combination of unrelated mini-tasks instead of parts of a greater experience. However, as it can be observed in the following figure, even while in development the overall cohesion feel of the game experience was very positive. It was possible then, even in early test phases to assume that the different systems were working well together, each fulfilling its function.

Do you feel the different elements of the experience so far (recruiting pirates, managing resources, exploring the world, engaging in narrative exents) create a cohesive experience?
8 responses



4.2.2.2. Prioritizing Improvements

One of the main uses of the development testing was to assess which were the most and less urgent improvements that could go into production during the last weeks. There were many directions in which this time could be spent and it was important to use it in the way it would have the most impact in the game's overall quality. Indirectly, this part of the survey also gave

an insight on which parts of the game felt complete enough to not require additional work. The following figure can be used as a reference to understand the colors in the graphics mean.

For example, the most highly voted option was polish bugs. This was one of the main reasons why in the final weeks of production the focus was set on polishing the existing experience instead of adding more mechanics and varied kinds of ways of interacting with the NPCs, such as NPC-triggered events or a notification tab, leaving that for future work. Similarly, improving current events was slightly higher scored than increasing event variety, which also led to prioritizing making complex, branching narrative events that could showcase the prototype's potential rather than making many one-screen events.

4.3. Final test results

4.3.1. Game session notes

Many of the development test session notes can be also applied to these game sessions, such as the immediate attachment to NPCs in the tavern screen or how in general the game prototype experience was well received. In this phase however, there could be more narrative content shown comparing to the one that was available in the development tests and that fact also shows in the results.

4.3.1.1. More homogenous experience

Thanks to the modification of forcing a minimum number of crewmates in the tavern level, the final test playthroughs were much more homogenous, since the increased variety ensured that over the development of the narrative events there were often modified text snippets for the player to see how the crew's configuration could affect the outcome of an event. This also helped playtesters generally play longer before getting bored or seeing like they had seen everything there was.

4.3.1.2. Playing favorites

Very often players would recurrently use the same character when an event would ask the player to pick a crewmember for a task, this way further defining the relationship between them and the NPC. Sometimes this relationship would make the playtester doubted whether it would be worth it to have that valued crewmate take on risks in narrative event out of fear of losing them. These kind of player conflicts show that the NPCs generated

4.3.1.3. Loyalty in NPCs being important to players

An element that happened in almost all playthroughs is that while there are eleven different stats which conform an NPC's characteristics such as melee, intelligence or charisma, out of all of them loyalty was by far the one that playtesters paid the most attention to. They would often keep generating pirates until randomly the loyalty stat would at least be average, which shows how players could already feel that the relationship between them and the NPC required loyalty from the latter to work. Not loyal NPCs would be judged as not to be trusted and differentiated from the rest, which is a way of showing how simple data in stats can define how the personality of an NPC is perceived by the player

4.3.1.4. Narrative taking center stage

Playtesters thoroughly enjoyed feeling like their decisions defined the outcome of the narrative events and how those in time affected the game state and made the NPCs evolve over time. Reactivity from the NPC's to specific player actions (example: adventurers being happy when deciding to go on a land expedition to explore a temple) were a very positive surprise for the

players, as how they could see that the NPCs having agency and individual goals and interests. Additionally, the internal information given to them such as the result of a skill check was well understood and received.

4.3.2. Final Survey results

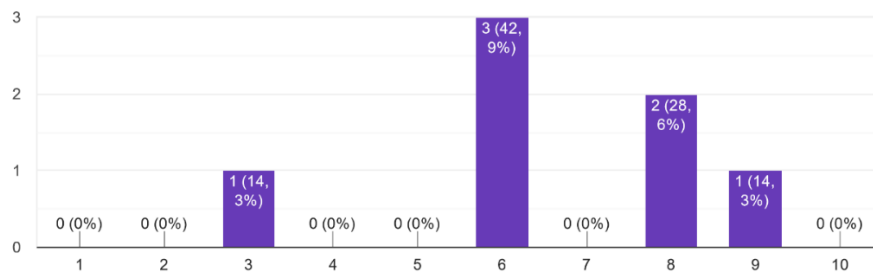
4.3.2.1. Engaging player-NPC relationships

As it can be observed in the figure below, the average scoring when determining how much playtesters developed engaging relationships with the NPCs is 6.58 out of 10, where 0 would mean they never did and 10 would mean the always did. Therefore, this question shows how all playtesters felt at least sometimes like they were creating engaging relationships with the NPCs, and several did so quite frequently. This is a positive result considering that the project itself is a prototype and with the implementation of future work this figure would probably increase.

Similarly, playtesters claimed to make up their own stories about the NPC crewmembers in their head while reading characteristics and narrative event descriptions on a 6.57 out of 10 on average, so it is possible to create basic data-driven information of procedural NPCs and their character will evolve in the mental model of the player.

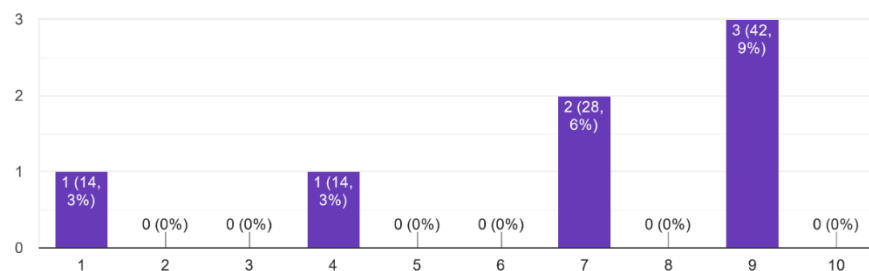
Overall, would you say you developed engaging relationships with crewmate NPCs over the course of your game session?

7 responses



Did you make up your own stories in your head about the crew or crewmember NPCs when reading up their characteristics or reading the narrative events?

7 responses



4.3.2.2. *Gameplay elements affecting mental image in narrative*

The question about which gameplay elements are most relevant when creating a mental image of their character brings a lot of information about how the players interact with procedural characters in narrative events. It is possible to see how the least relevant aspects are the ones that happen at the beginning and at the end of their relationship, them being NPCs getting recruited and them dying.

On the other side, players reading the procedural information as well as having the NPCs' characteristics evolving as consequences of events and affecting the narrative's outcome are two of the most relevant elements, showing how growth over time and consequences are elements that need to be taken into account when developing NPC narrative interactions.

Gameplay element	Average scoring (0-5)
Reading their stats	4
Them improving stats as consequence of events	3.9
Them affecting possible choices in events	3.6
Them succeeding/failing skill checks	3.3
Their characteristics affecting events	3.3
Making them interact with other NPCs	3.1
Them getting injured/dying in events	2.7
Recruiting them	2.4

4.3.2.3. *Gameplay elements affecting mental image in narrative*

Generally, all elements of the narrative mechanics were well received, such as the way the text could be modified by the game state, having choice availability be dependent on the game state and the general outcome of the narrative events. Additionally, elements outside of the strict dialog, such as the skill checks and the event illustrations were also positively viewed.

Procedural narrative mechanics	Average scoring (0-5)
The event description being modified by the crew's characteristics was engaging	4.2
Having different choices based on the crew's characteristics was engaging	4.1
The skill checks enhanced the experience	4
The outcome of the choices of the procedural events were engaging	3.4
The event illustrations made the events more interesting	3.28
The outcome of the choices of the procedural events were frustrating	1.7

4.3.2.4. UI cleanness

When evaluating the UI's capacity to convey information to the player, it is clear that those elements related to narrative events were much more clear than elements concerning actions outside of events, such as the map scene or the resource management system. Probably a tutorialized introduction is needed to familiarize the player with the different activities they can engage with. Additionally, color-coding can be considered a good solution to convey a lot of information with the intention to quickly show weak and strong character points.

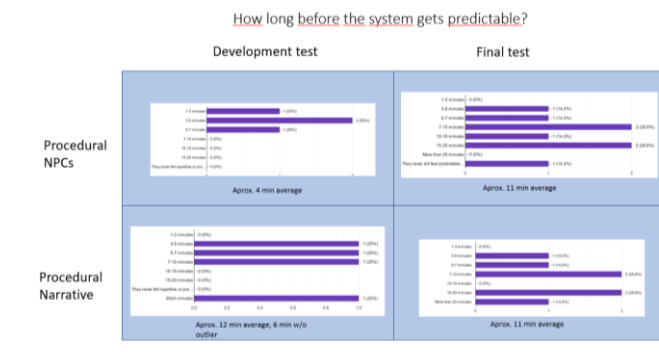
UI element	Average scoring (0-5)
NPC's color-coded tiers	4
Generally narrating the event situation	3.9
Showcasing specific narrative modifiers	3.3
Giving information about the crew's resources	3.1
Giving information about what actions the player can do	2.6

4.3.3. Comparing Development and Final tests

4.3.3.1. Comparing procedural predictability

After the first four development tests, it was clear that players were initially very engaged with the generator's possibilities. This is why in the following four development tests as well as in all the final tests two questions were introduced to test how long this initial feeling of engagement and surprise lasted. These questions were introduced because while having an initial positive reaction to the generated characters is already a positive element, there will always be a moment where the player will start to see repetitions and patterns in the system, making the generation feel less organic and immersive. It was the goal of these questions to see how long did it take before this point was reached and whether in final tests these points were reached further down the line.

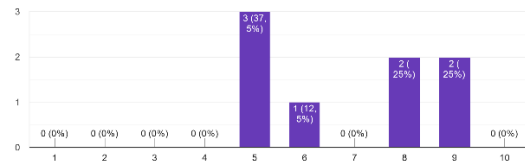
As it can be seen in the figure below, both in development testing the average time where the procedural NPCs generator became predictable was on average 4 minutes and the procedural narrative's time was on average 12 minutes, although an outlier could be excluded because of the difference in playthroughs experiences without a minimum of required crewmates. If it was excluded the average time would be 6 min. On the final tests both times increase to 11 minutes on average. So it is possible to assert that after the development tests both narrative and NPC procedural generators increased their complexity and playtime.



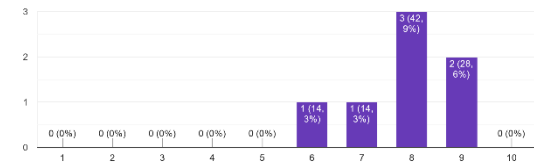
4.3.3.2. Comparing overall Enjoyment

It is also possible to compare how the playtesters rated the experience overall in terms of enjoyment. As shown in the two figure below, there is a clear improvement between the development test experience enjoyment (left figure), with a 6.9 on average and the final test experience enjoyment, with 7.9 on average. This difference could be attributed to improvement on bugs and other fixes in existing content result of the development test as well as the effect of the added narrative content.

Overall, how fun would you rate the game experience in its current state?
8 responses



How enjoyable was the game experience as a whole?
7 responses



5.1. Initial Survey conclusions

Even before having playtesters experiment with the prototype game, by seeing the initial survey results in section 4.1 it is possible to observe how the initial premise of this project was validated. The overwhelming majority of playtesters considered that it was possible to have engaging connections with NPC characters and that procedural NPCs could be engaging characters themselves. Additionally, they chose the procedural applications used in this project, procedural events and procedurally generated companions, as the most interesting ways to add PGC in a linear game.

Therefore, it is theoretically possible to enhance the experience with the production goals established in the introduction. It only remains to see whether the execution of this production has been successful.

5.2. Production goals' success assessment

As mentioned in the introduction, this project had as main production goals to successfully implement a procedural NPC generator whose generated characters would be complex enough for the player to develop engaging relationships with, and a procedural narrative game prototype which could have its content modified by the game state and the player's choices.

Firstly, it is possible to conclude that this project has been successful in implementing, as described in section 2.2, a procedural generator that creates NPCs with these characteristics:

- Random gender, name, cosmetic items
- Procedural stats distribution
- Random personality, liked and disliked traits
- Capacity to have an evolving relationship, recording events
- Capacity to remember relevant events with the player

Concerning the procedural narrative game prototype, a playable game has been developed over the course of these last months as described in section 2.3, with the following highlights:

- Two defined scenes, one to recruit NPCs and another to explore map and narrative
- Modular narrative event system, with the ability to spawn in random locations

- Choices, narrative description and consequences modified by the game state
- D&D-like skill checks to determine outcome
- Two kinds of events, map-interactable and captain actions

After interacting with these two elements in the game session, playtesters in the final testing phase overall found the experience enjoyable, with an average of 7.9 out of 10. That scoring, combined with the comments received from the survey and game session where players manifest liking the theme, execution and potential of the game, it can be concluded that the development of the goal experience has been a success. The core target experience, a feeling of freely managing a group of characters that can grow and deciding in engaging narrative events has been fully attained, although in a very prototype capacity, which could be expanded in the future.

5.3. Procedural Generation Question Conclusions

5.3.1. Engaging relationships with NPCs

As discussed in section 4.3.2.1 and 4.3.2.2, players did in general find engaging connections with the NPCs in their crew. Looking at them, as well as the game session notes in both development and final tests, it is possible to see a number of patterns repeating that give some insight in which gameplay mechanics affect most when developing those relationships, which was the first research question defined in section 1.1.2.

5.3.1.1 Reading initial information

At the beginning of the game, playtesters already spent a long time in the tavern area, having fun just generating NPCs and reading on their characteristics and quirks, making initial judgements on whether they approved of the newly created NPC. They could already even fill in the blanks and assign characteristics to the character that were not specifically stated in them just by extrapolating. Color-coding also helped make that process much easier and intuitive. It is important to make characteristics very easy to read for the players, since those are the base from which characters are built in the player's head. The capacity for players to make up characters with complex intelligence and personality based on simple random elements is comparable with Brian Reynold's perceptions that these kind of simple randomized data can be enough to elicit the perception of intelligence (Short, 2019, pg. 20)

5.3.1.2. Growth, support and consequence

The most successful mechanics besides reading stats are the way in which NPCs could improve their stats in narrative activities and their characteristics also affecting the choices available to the player in a given narrative situation. Interestingly, this fact can be related to the initial survey's question about what do players think makes an engaging character. In that question, two of the most voted answers were characters being able to improve their stats in activities and them providing support with the player, which is what they are doing in this prototype.

Characters being able to evolve makes them feel much more dynamic, just like humans are. This evolution is a reaction to events happening to them, and their behaviour not being static makes it more believable and relatable for the player. It is also a sign for the player that their decisions matter, since they have a direct impact on their crew, making those choices feel more important, giving the player control of the situation. Additionally, crewmen giving additional choices is the equivalent of giving support to the player, since those options would not exist without that particular NPC. This allows players to see a purpose to the existence of the NPC and reinforces

the feeling of being a team, where the player chooses and the NPC acts. It is interesting seeing how elements that playtesters felt were important in characters in general also clearly applies to procedural ones.

5.3.1.3. Stats are not equal – loyalty is key

In virtually all tests, the NPC stat players paid attention the most by far was loyalty. Many players would keep regenerating pirates until one popped with at least average loyalty before considering recruiting them. That makes sense because players want to feel like they have control, that the chance that a companion will turn on them is low. Additionally, NPCs take resources, so it is logical that players expect reliability in return. This is a good example of how not all stats should be necessarily treated equally. In this project, all stats have the same chance of being better or worse, but maybe characters in general would have been more engaging to the players if they had not looked so disloyal.

5.3.1.4. Chance to die / die too quick

The skill check mechanic was very well received, as a way in which crewmen could put their stats to the test in narrative activities. It probably was successful because while randomness has a considerable part in its implementation, the odds are also modified by the NPC's characteristics, giving the player a bit more control while keeping the tension of the possibility that failure could occur. This possibility makes players strategy and care more about their decisions as well as showing them how they have control over the narrative.

Interestingly, the least popular mechanic was having the NPCs die or get injured. Having witnessed the tests' game sessions, a possible explanation for this is that in many cases those deaths occurred to random crewmen, even those that had not had a lot of previous interaction with the player, making their demise much less impactful for the player. This is why a good mechanic to take into account is having the NPCs only die or die mostly after having had a number of interactions with the player, to make sure they have had enough time to connect emotionally with the NPC before losing it.

5.3.2. Narrative Mechanics

5.3.2.1. Making characters feel unique

One of the main risks of having procedural content generated is that there may be too much for the player to focus on or remember, making some of the characters overshadow the rest, simple because they have been chosen by the player before others. Having ways to easily differentiate them is key to make them not feel like permutations of the same generator, but actually unique characters. This is why the ability of assigning special roles was important and had good reception during the test, as well as the cosmetic items that allowed each character to have a more unique feel. While obviously these differentiations could be developed, they already laid the foundation of the system to avoid characters feeling like they were the same.

5.3.2.2. Modular modifiers make for good replayability

It is perfectly possible to implement a modular system of events with text that can be affected with keywords which then are replaced with game state variables. Playtesters enjoyed how the NPCs reacted to the narrative events in a manner that was consistent with their personalities. The fact that the same event could drastically change depending on the crew's configuration improved replayability in the sense that players wanted to try again to see what could happen when using other pirates

5.3.2.3. Having visual information

While reading can be a great way to give a lot of information quickly, it is also a bit difficult to represent engaging characters and situations on words alone, especially if people developing this project are far from skilled writers. Having a variety of media combined, such as a ship figurine to represent the crew's location or an illustration to visualize what is happening on the text are two good examples of how fatigue from reading in a narrative game can be avoided.

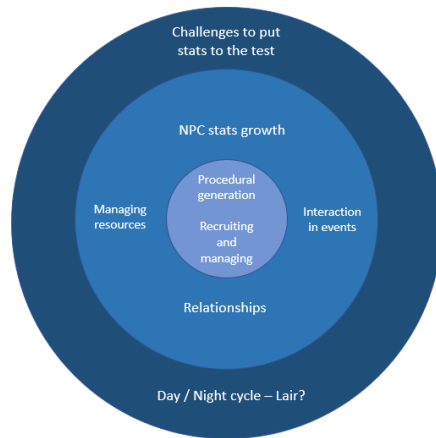
6. Future Work

As previously discussed, there are a lot of ways this project could be continued in future work, since it remains as of now a prototype. There were a lot of additions and modifications that had to be put on hold due to time constraints. In the following table it can be seen those features that were most interesting to playtesters as potential future additions to the game.

Potential improvement	Average scoring (0-3)
Implement spontaneous narrative events, activated when conditions are met (ex: mutiny if the loyalty is low)	2.6
Improve UI when showing NPC information	2.4
Increase number of events	2.4
Combine narrative events and crew management with combat scenes/missions (XCOM-like)	2.3
Make existing events more complex	2.1
Improve UI in narrative events	2.1
Have event figurines to distinguish different event types on the map	2
Be able to manage non-officer crewmen to man different parts of the ship (FTL-like)	1.6

As it can be seen, quality of events is not the most urgent improvement but rather the increase in variety. As is could be expected, a kind of event that does not need any player action to trigger, but that it is triggered by the NPCs themselves given certain situations was the most valued option. That would probably give the NPCs a much larger feeling of agency and independence and would create unforeseen challenges for the player, in comparison with the rather reactive world that is implemented so far. Additionally, improving UI when showing players their possible actions is also a high priority, since it was rather difficult for players to reach all the possibilities of content due to the UI design requiring too many clicks.

Finally, one additional improvement that was not listed in the survey questions but rather suggested by my reader, Ben Schneider, was the possibility of adding another layer of depth in the game experience, since in the state of the game right now players recruit crewmen (1st layer) and then interact with them and have them evolve and interact with each other (2nd level) but there is not a lot of actual challenge. A possible 3rd layer could be a day/night cycle with procedural challenges to solve and put the skills to the test.



7 References

[1] Vandrake, Procedural Generation in Game Development

<https://www.davidepesce.com/2020/02/24/procedural-generation-in-game-development/>

[2] GDC talk: C. Martens and R. Cardona-Rivera. (2017) Procedural narrative gen-

eration. <https://www.gdcvault.com/play/1024143/Procedural-Narrative>

[3] Crusader Kings III. Paradox Interactive, 2020.

<https://www.paradoxinteractive.com/games/crusader-kings-iii/about>

[4] Middle Earth: Shadow of War. Warner Bros, 2017. <https://www.shadowofwar.com/>

[5] Battle Brothers Game. Overhype Studios, 2015.

https://store.steampowered.com/app/365360/Battle_Brothers/

[6] Bycer, J, A Study Into Replayability -- Random vs. Procedural Generation, 2018

<https://www.gamedeveloper.com/design/a-study-into-replayability----random-vs-procedural-generation>

[7] T. Short, T.Adams, Procedural Storytelling in Game Design, 2019

8. Annex

8.1. Initial test survey questions

What genre of videogames do you tend to play the most?

- Casual
- Puzzle
- Strategy
- Shooter
- RPG
- Action-Adventure
- MOBA (Multiplayer online battle arena)
- Simulation
- Sports
- Survival or Horror
- Platformer
- Altres: _____

Do you think it is possible to have an emotional connection with an NPC character in a videogame?

	1	2	3	4	5	6	7	8	9	10	
No	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Yes

If you have answered with a 6 or more in the previous question, mark here a maximum of FOUR ways that are most likely to make you feel engaged with an NPC character

- Having a dialog with him
- Have him help you when facing challenges (fight alongside you, give you items, help you platform around the environment)
- Making a choice that will directly affect this NPC's life
- Him having an interesting personality
- Him being able to react/comment on your previous actions
- If the NPC is an enemy, fighting him and defeating him
- The NPC doing something unexpected (i.e coming back to life after seemingly dying, betraying you...)
- Altres: _____

Do you think a procedurally generated NPCs can be engaging characters? *

	1	2	3	4	5	6	7	8	9	10	
No	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Yes

Do you think linear games with a fixed story might increase their replayability by adding procedural content? *

	1	2	3	4	5	6	7	8	9	10	
No	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Yes

If you answered with a 6 or more to the previous question, pick the THREE elements do you think would be the most interesting to add in a linear game.

- Random challenges/quests with rewards that the player can choose to take on
- Procedural enemies with different equipment, strengths, weaknesses and personalities
- Procedural events with a description and a set of choices which the player must make, The number and nature of choices could be affected by the resources/skills he has at the time
- Randomized loot/items with different names, stats, etc.
- Procedurally generated areas
- Procedural followers/companions with different equipment, strengths, weaknesses and personalities which the player can recruit and command
- Autres: _____

If a game had a procedural generative system to create enemies, would you prefer complete randomness in their characteristics or would you prefer that the system also took into account your previous decisions to give you a tailored challenge? Example: You have been using mainly melee weapons to combat, therefore in the next dungeon there will be a larger number of archers to favor a change of tactics.

1 2 3 4 5 6 7 8 9 10

Completely random ○○○○○○○○○○ Completely based on previous actions

8.2 Development test – Post game session survey questions

What is the one thing that you would keep no matter what from this game? What is the one thing you would remove? Why?

Text d'una resposta llarga

Which game systems/controls do you think need to be better clarified to the player?

Text d'una resposta breu

Do you feel the different elements of the experience so far (recruiting pirates, managing resources, exploring the world, engaging in narrative exents) create a cohesive experience?

	1	2	3	4	5	6	7	8	9	10	
No	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Yes

Have you had any technical problems during your game session?

Text d'una resposta llarga

How long did it take for you to feel like the procedural NPCs could not surprise you anymore, like you had discovered everything there was in terms of generator complexity? In other words, how long did it take for the procedural elements of the game feel predictable or repetitive?

- 1-3 minutes
- 3-5 minutes
- 5-7 minutes
- 7-10 minutes
- 10-15 minutes
- 15-20 minutes
- They never felt repetitive or predictable
- Autres...

How long did it take for you to feel like the procedural narrative events could not surprise you anymore, like you had discovered everything there was in terms of generator complexity? In other words, how long did it take for the procedural elements of the game feel predictable or repetitive?

- 1-3 minutes
- 3-5 minutes
- 5-7 minutes
- 7-10 minutes
- 10-15 minutes
- 15-20 minutes
- They never felt repetitive or predictable
- Autres...

How much priority should these possible improvements take over the next couple of weeks? (Please put a maximum of 3 elements in each priority column)

	Don't implement it	Very low priority	Low priority	Medium priority	High priority	Must be implemented
Improve current events	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Increase event variety	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
More actions outside of narrative events (real time combat, obstacles/dangers when exploring)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
More ways to interact with the NPCs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Give NPCs non-gameplay affecting details (nicknames, cosmetics, random backstory...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Increment NPC complexity (goals they will try to achieve, reactions...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Improve visual quality (illustrations, UI, animations)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Polish bugs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Improve information given to the player	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Could you think of a possible addition not listed on the previous question?

La vostra risposta _____

Overall, how fun would you rate the game experience in its current state?

1	2	3	4	5	6	7	8	9	10
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

8.3 Final phase test – Post game session survey

What would you say was the most and least engaging part of the experience?

La vostra risposta _____

How long did it take for you to feel like the procedural NPCs could not surprise you anymore, like you had discovered everything there was in terms of generator complexity? In other words, how long did it take for the procedural elements of the game feel predictable or repetitive?

- 1-3 minutes
- 3-5 minutes
- 5-7 minutes
- 7-10 minutes
- 10-15 minutes
- 15-20 minutes
- More than 20 minutes
- Altres: _____

How long did it take for you to feel like the procedural narrative events could not surprise you anymore, like you had discovered everything there was in terms of generator complexity? In other words, how long did it take for the procedural elements of the game feel predictable or repetitive?

- 1-3 minutes
- 3-5 minutes
- 5-7 minutes
- 7-10 minutes
- 10-15 minutes
- 15-20 minutes
- More than 20 minutes
- Altres: _____

Overall, would you say you developed engaging relationships with crewmate NPCs over the course of your game session?

	1	2	3	4	5	6	7	8	9	10	
Never	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Constantly

When interacting with NPCs, describe how relevant was each gameplay element in developing a relationship with them, creating a mental image of their character.

	It was not relevant	It helped very little	It somewhat helped	It helped considerably	It was very relevant	It was fundamental
Reading their stats / characteristics	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Recruiting them	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Making them interact with other NPCs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Them improving stats as consequence of events	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Them getting injured/dying in events	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Them succeeding / failing Skill Checks	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Them affecting possible choices in events	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Their characteristics affecting events	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Is there an additional comment you would like to add concerning your experience interacting with the procedural NPCs?

La vostra risposta

Did you feel immersed in the procedural narrative story based on events? *

	1	2	3	4	5	6	7	8	9	10	
No	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Yes

Did you make up your own stories in your head about the crew or crewmember NPCs when reading up their characteristics or reading the narrative events?

	1	2	3	4	5	6	7	8	9	10	
Never	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Constantly

Can you think of any moments where you stopped thinking about your crew as system/gameplay and just thought about them as people/relationships?

La vostra risposta

Please evaluate these statements about procedural narrative mechanics in narrative events:

	Strongly disagree	Disagree	Somewhat disagree	Somewhat agree	Agree	Strongly agree
The event's description being modified by the crew's characteristics was engaging	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Having different choices based on the crew's characteristics was engaging	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The skill checks enhanced the experience	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The outcome of the choices of the procedural events were engaging	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The outcome of the choices of the procedural events were frustrating	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The event illustrations made the events more interesting	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

How clear was the UI in giving information to you in these elements?

	Not clear at all	Quite unclear	Somewhat unclear	Somewhat clear	Considerably clear	Very clear
NPC's colorcoded stat tiers	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Giving information about the crew's resources	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Giving information about what actions you could do as a player	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Generally narrating the event situation	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Showcasing specific narrative modifiers (ex: a particular personality affecting an skill check in an event)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Would you agree that having the "capsule" portraits with cosmetic items helped making the characters feel more unique than if they were only described with text?

1 2 3 4 5 6

Strongly disagree Strongly agree

How enjoyable was the game experience as a whole?

1 2 3 4 5 6 7 8 9 10

Can you elaborate on the reason for your previous scoring?

La vostra risposta _____

How much priority should these potential improvements have if they could be implemented as future work? (Please put a maximum of 4 elements in each priority column)

	Don't implement it	Could be implemented	Should be implemented	Must be implemented
Have event figurines to distinguish different event types on the map	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Be able to manage non-officer crewmembers to man different parts of the ship, giving different bonuses (FTL-style)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Combine narrative events and crew management with combat scenes/missions (XCOM-style)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Increase number of Events	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Make existing events more complex	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Improve UI when showing NPC information	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Improve UI in narrative events	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Add spontaneous narrative events, activated by the NPCs given a number of conditions in the game state (example: mutiny if loyalty is too low)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Is there another addition you would add that is not listed in the previous question?

La vostra risposta

Do you have any other comment to make about your game experience?

La vostra risposta
