Joshua Gorton

Supervisor: **Peter Blenkharn**

Due **20/04/26**

Can generated POIs intuitively communicate resource affordances?

Word count: 6500

A dissertation submitted in partial fulfilment of the University of Greenwich's

BSc (Hons) Games Design and Development (Programming)

ABSTRACT

This report explores how effectively Points of Interest (POIs) can intuitively inform the player/participant on what resources/content they might find there, or even what possible risk and reward might be waiting for them. I draw on Architectural Level Design to define POIs, and James Gibson's theory on visual perception that I believe to be a big part of this project. In this project, I'll be making a tool that generates islands and their props, and places them into an open water Unity scene. There will be two groups, a control and the main group: the control group will see only coloured spheres for islands, while the main group will have all the visual information that I can offer without the use of UI. The player is able to pilot a small boat and is tasked to go to islands to survive. I'll collect gameplay data on first choices, travel speed, average direction, and record post-play answers including the GUESS-18 with the GUESS-55 exploration subset questionnaire. The goal for this project is to prove if POIs can communicate a broad amount of information to the player without the use of UI, and if so, just how much of a difference does it make.

Table of contents

1 INTRODUCTION

1.1 Background

In this report, I will be exploring how the use of Points of Interest can influence a player's decision-making within an environment when given an abundance of choice.

So, what is a POI? A POI, or Point of Interest, is what Kevin Lynch in image of a city refers to as a node or landmark. It is something that visually stands out to the player within an environment. In games, these are referred to as POIs because they often indicate to the player that there is content, loot, or an objective to be found at that location encouraging them to travel there. POIs are a powerful tool for level designers, as they can subtly guide players in a general direction without directly instructing them to do so. That is the greatest strength of a well-executed POI it shows rather than tells.

With the rise of open-world environments, I think it's vital to look further into this tool and explore how players interact with POIs. In this research report, I will be running a test to help distinguish how much of a difference POIs can make and how effective they are at communicating information.

I'll be creating a small open-world sea game where players/participants control a small boat. Then I will be generating an abundance of different islands to explore some larger, smaller, rigid, rocky, green, darker, flat, ex… Then I'll see if I can correctly and effectively communicate what they may find there.

1.2 Research, aims and objectives

Drawing on Lynch's framework (covered in detail in section 2.2.1), I treat a POI as a hybrid of his landmark and district concepts: both an external reference point and a destination.

The core aim of this project is to determine if players can intuitively understand a game environment without needing explicit text or user interface (UI) markers. This is especially useful to look at as people are finding Diegetic UI more desirable year after year, I mean look at sons of the forest or dead space

1.2.1 Aims

Aim: To develop an island generation tool that can effectively convey resource affordance and perceived difficulty based on the tools generated visual traits.

Question: Can procedurally generated POIs effectively & intuitively communicate resource affordances to players?

1.2.2 Objectives

Objectives 1 : I'll need to Research theories on Points of interest as well as resource affordance and perceived difficulty. This will assist me in both setting up the test but also help me better understand the results I obtain .

Objectives 2 : To Develop a C# procedural generation tool in Unity to create various islands, that the player can travel to give me an item to measure.

Objectives 3 : Implement a resource-management gameplay loop (Food, Health, Fuel) to create "survival pressure." this will also act as my measure of whether or not resource affordance is a factor on player decision making

Objectives 4 : Conduct a blind A/B experiment (Visuals vs. Control) to measure survival rates and decision accuracy.

Objectives 5 : Evaluate whether procedural geometry alone can effectively guide player behaviour.

2 Literature Review

2.1 Theories of Spatial Navigation

2.1.1 Kevin Lynch: The Image of the City

The first key text I am relying on is The Image of the City by Kevin Lynch (1960). Although Lynch was an urban planner studying real cities like Boston, his work is considered the gold standard for understanding how humans navigate space. Lynch argues that for a space to be navigable, it must be "legible," meaning the layout is easy to understand and organise in the mind. He breaks cities down into five elements, but the one that matters most for my project is the Landmark.

Lynch (1960) defines landmarks as external reference points that the observer does not enter but uses to orient themselves. This perfectly describes the gameplay loop I am building: the player is in a boat, looking at distant 3D meshes on the horizon. Lynch states that for a landmark to work, it must possess "singularity", it needs to contrast sharply with its background. This theory directly informs my procedural generation code. The ocean in my Unity scene is a flat, repetitive background. If I generate islands that are too flat or generic, they will lack singularity, and the player won't notice them. By using Lynch's definition, I am ensuring my code generates islands that act as functional navigation tools rather than just background noise.

2.1.2 Christopher Totten: Architectural Level Design

While Lynch focuses on city planning, Christopher Totten (2014) applies these ideas specifically to game levels in An Architectural Approach to Level Design. Totten bridges the gap between pure architecture and game mechanics. He argues that in games, "form follows fiction." This means that the shape of a level shouldn't just look realistic; it should visually communicate the gameplay possibilities to the player.

Totten (2014) suggests that architecture creates the environment for the core mechanics. In the context of my project, the core mechanic is survival (managing the three bars: Food, Health, and Fuel). Therefore, the "architecture" of my islands must visually explain which mechanic it satisfies. If an island provides "safety" (Health), its form should look like a shelter. If it provides "Industrial Fuel," its form should look manufactured or resource-heavy. I am applying Totten's theory by ensuring my procedural scripts follow strict rules. I am not just randomising the terrain for the sake of variety. Instead, I am attempting to encode the "fiction" of the resource into the "form" of the mesh. If the player cannot look at a generated shape and guess what resource it holds, then the form has failed to communicate the fiction.

2.2 Theories of Perception and Psychology

2.2.1 James J. Gibson: Affordance Theory

The concept of "knowing what an object does just by looking at it" comes from James J. Gibson's The Ecological Approach to Visual Perception. Gibson (1979) introduced the term Affordance. He defines the affordances of an environment as what it offers the animal, what it provides or furnishes, either for good or ill.

The key part of Gibson's theory is the idea of "Direct Perception." He argues that we don't need to intellectually process geometry to calculate what an object is. Instead, we directly perceive the possibilities for action. For example, when we see a flat surface, we instinctively know it allows for walking. In my project, I am attempting to create "Procedural Affordances." When the player sees a Green/Lush island, I want them to directly perceive "edibility" (Food) without needing a UI text box to tell them. Gibson notes that the "layout of surfaces" is what defines these affordances. This is why my tool focuses heavily on texture and object density. If I can get the procedural generation right, the player should instinctively steer towards the Green island when they are hungry, proving that the visual cues successfully communicated the affordance without words.

2.3 Existing Products related to this project

2.3.1 Subnautica (Unknown Worlds Entertainment, 2014)

Subnautica is a prime example of a survival game that uses the environment to communicate resources. The game is divided into distinct biomes (e.g. the Safe Shallows versus the Kelp Forest). Players quickly learn that if they need specific resources, they must look for the specific colours and shapes associated with that biome. The visual design of the world is the map.

However, Subnautica uses a static, hand-crafted map. The designers placed those biomes manually to ensure they made sense. My project differs because I am attempting to replicate this logic using procedural generation. I want to see if a computer algorithm can create that same level of "biome logic" on the fly, without a designer manually placing every tree. If successful, this could save huge amounts of development time for future open-world games.

2.3.2 Ghost of Tsushima (Sucker Punch Productions, 2020)

Ghost of Tsushima is famous for having almost no on-screen UI during exploration. Instead of a minimap arrow, it uses the "Guiding Wind" a particle system that blows in the direction of the objective. It also uses animals, like gold birds and foxes, to lead players to secrets.

While this is effective, it is a "Diegetic" solution meaning the cue exists inside the game world but it is still an arbitrary "pointer." A gold bird is basically just a floating arrow disguised as an animal. My project is trying to do something more subtle. I don't want to use a "guide" object like a bird or wind. I want the geometry itself (the shape of the terrain) to be the guide. I am testing if the destination can be the marker, rather than needing a separate particle effect to point the way.

2.3.3 Elden ring

Elden ring is a great example of this in action because it almost ditches the UI entirely and relies on landmarks to guide the player around its open world. The Erdtree is visible from almost anywhere in the game and it acts as a constant orientation point exactly like Lynch describes a landmark should. You don't need a marker telling you where to go because the tree is so visually dominant that you can gather your general position and heading.

It also uses Gibson's affordance at a smaller scale. The way areas are built and lit gives you a rough idea of what you're walking into before you actually get there. Which is the kind of direct perception Gibson talks about. Players pick this up on their own without the game directly telling them.

The big difference is that Elden ring was sculpted and fitted by hand. Every landmark and point of interest is placed by a designer. My project is trying to do the same kind of communication but procedurally, without a human placing every detail.

3 PRODUCT DESIGN

3.1 Interaction

To solve this problem, I will be developing an open-sea survival simulation. The experiment consists of two participant groups. The Control Group will receive minimal visual information; they will be able to distinguish that one island is different from another only by a random colour. For example, fuel islands will be coloured red and food islands blue for these participants. The island themselves will be replaced by simple spheres, this is so that no geometric information is accidentally communicated to the player.

In contrast, the Experimental Group will have full visual information available to them. For them, the fuel islands will be rocky with metal barrels dotted around, and the food islands will be lush green with trees scattered on top. This is to attempt to communicate to the player resource affordance and also perceived difficulty.

Using data collection within the Unity scene, I will compare which group survives longer, how confidently they navigate, and which islands generated worked best.

Finally, I will combine this telemetry with post-play data collected using the GUESS-18 questionnaire and specific project-related questions. By analysing this data through the lens of my background research, I aim to prove the validity of my research question.

3.2 Planning - Flow chart

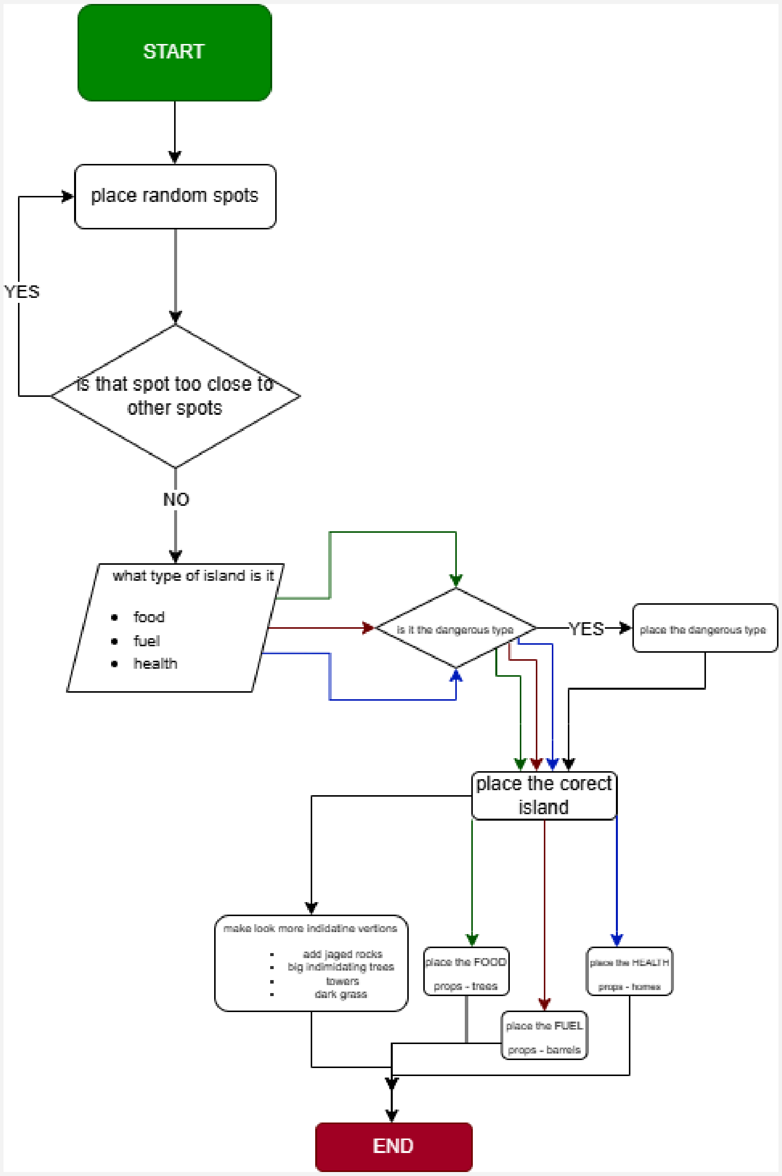


Figure 1

To construct the actual tool, as shown in Figure 1, I will create a script that randomly places points around the Unity scene to serve as potential island locations. The system will then compare the distances between each point to ensure they are not too close together. Once a valid position is confirmed, the script instantiates an island prefab of a specific type. A secondary script attached to the island will then define its identity, colour it appropriately, and finally procedurally scatter relevant props across valid spots on the terrain (e.g., oil barrels for Fuel islands, trees for Food islands).

3.3 MoSCoW table - Proposed Product Features

Feature	Priority
Boat / player movement	MUST
Island Placement generation tool	MUST
Distinct island variants to visit	MUST
In game data collection	MUST
Resource and survival mechanic	SHOULD
GUESS 18 Questionnaire	SHOULD
Island Prop placement generation tool	SHOULD
Two group set up – (group A is normal but group B is the base line so I need to set that up and make it un-biased)	SHOULD
My own additional questions for post participation	COULD
Start menu	COULD
Generating randomized meshes for islands using Perlin noise and that takes parameters based on the island type	COULD
Custom props and textures	WON'T
UI (other than the possible main menu and back button)	WON'T
Particle effects and game feel	WON'T

Table 1

Table 1 outlines the full scope of my development using the MoSCoW prioritisation method. This is important for keeping my project realistic. It forces me to draw a line between the features I absolutely need to run the experiment and the ones that are just nice to have.

the Must Haves are the non-negotiable features. If any of these are missing, the project fails because I wouldn't be able to answer my research question. For example, the Procedural Spawning Director and the Data Logging script are vital. Without the spawner, there is no environment to test, and without the logger, I have no data to analyse. I also need the Survival

Mechanics (Food, Fuel, Health) to actually give the player a reason to move; without that pressure, they are just floating aimlessly.

The Should Haves are features that are important for a good user experience but aren't strictly critical. For instance, my own additional questions and a proper Main Menu. I really should include these, but if I ran out of time, the experiment would technically still function without them.

The Could Haves are my stretch goals. I will only touch these if I am ahead of schedule. A Generating randomized meshes would be interesting to test if Kevin Lynch's "legibility" theory holds up, but it adds a lot of technical complexity.

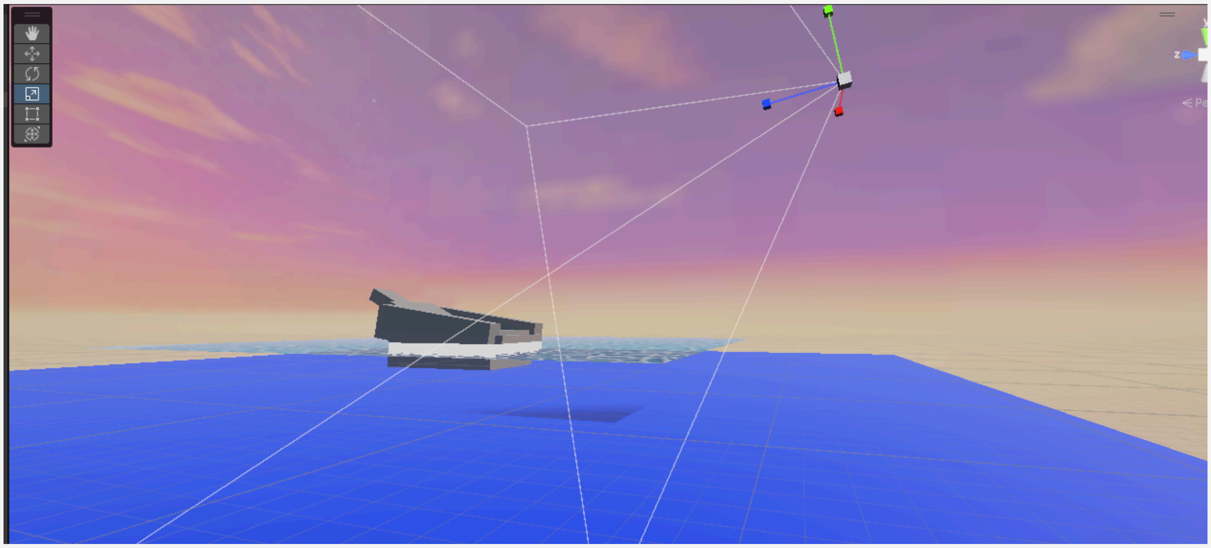
Finally, the Won't Haves are features I have explicitly decided not to build. Things like Multiplayer or Combat are completely out of scope. They would take huge amounts of effort to code and would distract the player from the navigation task, making the data noisy and unreliable. The payoff isn't worth the inconvenience, so I am cutting them now to focus purely on the single-player exploration loop.

4 PRODUCT DEVELOPMENT

4.1 Scene Configuration and Player Setup

The initial phase of development involved configuring the base Unity environment. I constructed the ocean using primitive planes and applied free-to-use assets for the water texture and the skybox to establish a clear horizon line. To prototype the player avatar, I assembled simple box meshes to act as the boat and implemented a basic movement controller.

Crucially, I adjusted the main camera to a third-person perspective and intentionally increased the Field of View (FOV). This was not just an aesthetic choice. A wider FOV gives the player a broader field of vision, ensuring that multiple procedural islands fit on the screen at once. This allows the player to actively compare their navigational options and maintain the situational awareness required for the experiment.



4.2 Player Controller and Movement

Next, I needed to add movement to the player's boat. I wanted the controls to feel heavy and deliberate so that the player is forced to take their time choosing where to go. If they could just speed around from island to island, the data I collect wouldn't really reflect careful decision-making. I wanted the movement to enforce a specific behaviour: looking around, making an informed decision, and then setting off.

To do this, I made a custom C# movement script that is entirely based on momentum. In a way, speed becomes a fourth resource. You turn your limited fuel into forward momentum, so you don't want to waste it. Speeding up takes time, and slowing down takes even longer. Turning is also slow and acts as a secondary braking force. This encourages the player to think ahead rather than making erratic decisions on the fly.

4.3 Procedural Island Generation

The next phase of development focused on creating the points of interest: the islands. These islands are the core component of the project; they serve as the visual stimuli for the resource affordances and are the primary mechanism for answering the research question. The player's ability to identify and navigate towards them provides the measurable data required for the experiment.

My initial concept was to procedurally generate the island meshes at runtime, using variables to influence their overall topology. However, upon reviewing the project requirements and timeline, I determined that building a runtime mesh generation system would be significantly out of scope and would detract from the core focus on visual affordance and data collection. Therefore, I opted for a more time-efficient and optimised solution. Rather than generating new geometry for each island during gameplay, I manually sculpted twelve distinct island base meshes and saved them as prefabs. I then developed a procedural spawning tool to handle their placement within the scene.

4.3.0 Spawning Logic and Distribution

The procedural spawning tool is responsible for randomly distributing these prefabs around the environment. A crucial aspect of this script is its distance-checking algorithm. The tool ensures that islands spawn within a relevant radius of the player, but it strictly enforces a minimum distance between each generated island.

This minimum distance is a vital component of the experimental design. By guaranteeing a significant travel distance between choices, the system reinforces the importance of decision-making. Participants cannot rapidly hop between islands; they must commit to a route. This enforced travel time encourages players to actively study the visual characteristics of the islands on the horizon, ensuring that the resulting telemetry data reflects informed choices rather than random exploration. Additionally, the tool caps the maximum spawn distance to ensure the islands remain visible and relevant to the player's immediate navigational context.

4.3.1 The Island planning

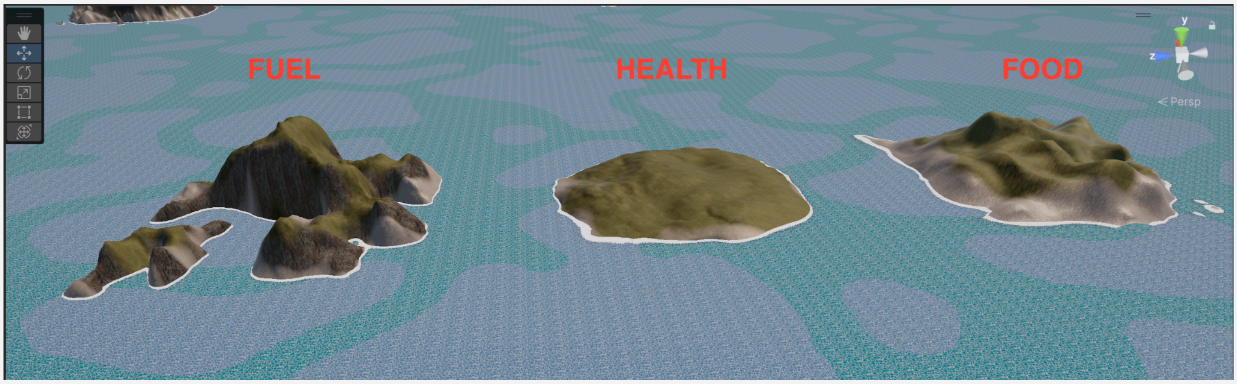
Num.	Type	Risk	size	AFFECT	ISLAND FETCHERS + LIST OF PROPS
1.	Food	LOW	S	@ +10F @ 10%: -0H	ISLAND: low, round, hollow PROPS: trees, crates, wood barrels
2.	Food	HIGH	S	@ +20F @ -10H	ISLAND: low, sharp, jagged, hollow PROPS: Trees, crates+, wood barrels+
3.	Food	LOW	L	@ +20F @ -5H @: -20F	ISLAND: low, round, full PROPS: trees+, crates, wood barrels
5.	Food	HIGH	L	@ +50F @ 40%: -50H	ISLAND: high, little jagged, full PROPS: Trees, crates+, wood barrels+
6.	Fuel	LOW	S	@ +10U @ -0H	ISLAND: low, jagged, hollow PROPS: oil barrels, oil riggs wood boxs
7.	Fuel	HIGH	S	@ +20U @ -10H	ISLAND: high, rocky, tall PROPS: oil barrels, oil riggs
8.	Fuel	LOW	L	@ +20U @ -5H	ISLAND: low, full PROPS: oil barrels, oil riggs wood boxs
9.	Fuel	HIGH	L	@ +50U @ -50H	ISLAND: tall, rocky, PROPS: oil barrels, oil riggs
10.	Health	No	L	@ +30H @ -0H	ISLAND: low, full PROPS: House
11.	Health	No	L	@ +25H @ -0H	ISLAND: tall, full, cliffs? PROPS: House
12.	Health	VERY HIGH	L	@ +20F @: +20U @: +20H @ -100H	ISLAND: tall, jagged, rocky, hollow PROPS: houses, stone towers, canons.

Table 2

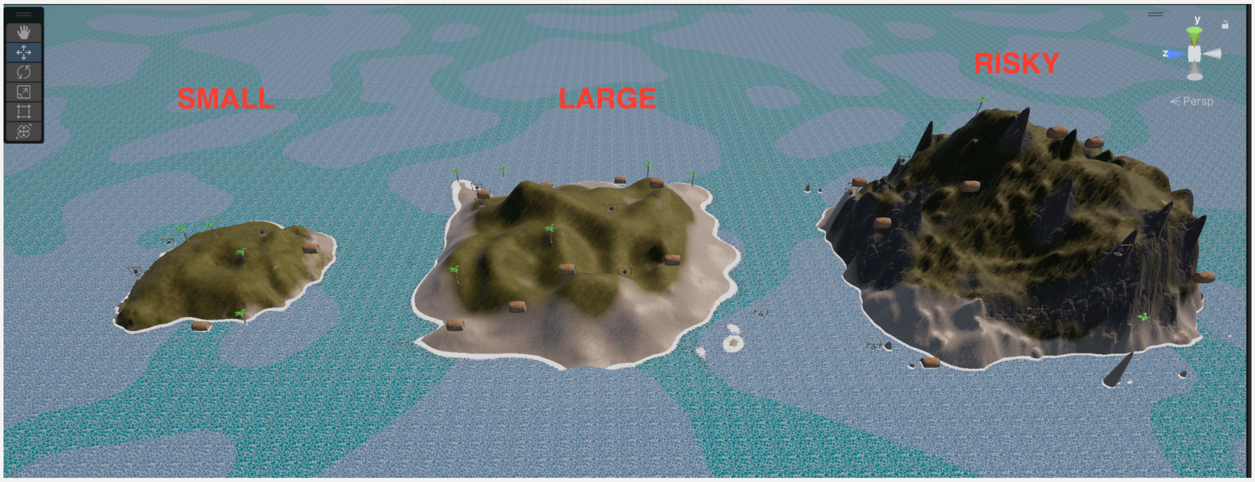
As I was now pre making 12 different islands it was imperative that I had a plan so I made the table above, table 2. This was my plan for the 12 islands that the player would travel too to collect resources. I broke each island up into 5 parts: first island type whether or not it would give the player things like food or health, then the riskiness of the island the riskier the island the more health it'll take from the player if it is collected. Next how big the island was, small or larger, each island type had 2 small islands and 2 big islands then one of each of those would be risky. Then finally it all came together in the final section which listed the characteristics the island would have based on the previous section. I would then go back to this table many times to keep track of each island while sculpting, texturing and programming in functionality.

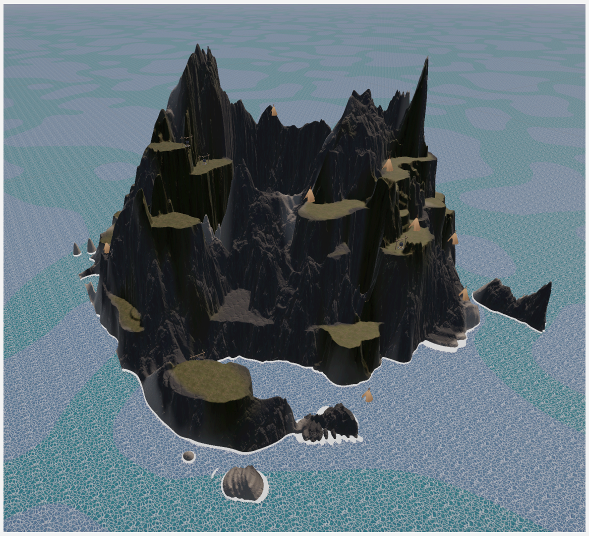
4.3.2 Making the islands

Now I had the table I needed to actively sculpt the islands. For safer islands i made them more rounded and flat to look more inviting as for rocky islands i added more rocky formations that would look more intimidating and imposing. Then I also wanted the type of island to play a part in the island layout. For food islands kept them very rounded, for health islands i wanted them to look like a camp or civilisation so i make them very flat for huts or houses to sit right on top, lastly the fuel island i kept them a little rocky to try to convey minerals and oil but also did the opposite of rounded i wanted there to be many bays and divides in it to give people a desire to park like in the island, like having many parking spots to indirectly imitate petrol stations.



For the textering of the island I wanted to make slight use of colour in the detailing to further help with resource affordance but also perceived risk. What I decided to do is make 2 grass texters one slightly darker than the other for riskier islands and in addition to this I added a dark stone texter to detail in to make some subtle darker cliffs.





As a final addition to this typology, I designed a specific outlier island engineered to act as a pure hazard, offering zero resource benefits. The purpose of this variant was to test the absolute maximum threshold of perceived risk. To achieve this, I sculpted the mesh to appear as visually imposing as possible, utilising extreme, jagged verticality and cavernous geometry. The surface was heavily textured with dark stone and sparse, desaturated grass. This design explicitly leverages the theories discussed in my literature review, using harsh architectural forms to signal an unmistakable environmental threat.

4.3.3 Procedural Prop Population

To fully establish the visual affordances of each point of interest, the islands could not remain as bare terrain. The silhouettes needed to be enhanced by dynamically generating relevant 3D props (such as lush trees for food, or metal barrels for fuel) across the surface.

To accomplish this, I implemented a secondary procedural generation script attached locally to each island prefab. The algorithmic logic mirrors the global island spawner but operates on a micro-scale. When an island is instantiated, the script calculates a predefined number of random coordinate points within the specific boundaries of that island's mesh. It then iterates through these points, mathematically verifying the distance between them. If two points are too close, the script discards one and recalculates a new position. This strict spatial checking is crucial to prevent the 3D prop models from clipping into one another or spawning in unnatural clusters. Once a valid array of coordinates is confirmed, the script selects the appropriate prop prefabs based on the island's designated resource type and instantiates them at those precise locations, effectively dressing the environment at runtime.

5. Evaluation

5.1 Methodology

To answer my research question, I ran a blind A/B experiment with 32 participants split across two groups. The Control Group (10 participants) saw the game world reduced to coloured spheres only. Each sphere was randomly colour-coded so the participants could still remember what each island offered, but no geometric or environmental information was communicated. The Experimental Group (22 participants) played the full version of the game with all the procedural islands, props, textures, and visual cues I had built.

Both groups played the exact same game underneath. The mechanics, the survival pressure, the resource drain rates, the boat controls, the spawn logic, all of it was identical. The only difference was what they could see. This was important because I wanted to isolate the visuals as the variable. If the Experimental Group survived longer, I needed to be able to confidently say it was because of the visual resource affordance of the POIs and not because of something else.

Each participant played one run until death. I logged everything the game could give me: how long they survived, how far they travelled, average speed, every island they visited, the time they visited it, what their resource bars were at the moment of collection, and the closest island to each one they visited. After they finished playing, I asked them to fill out the GUESS-18 questionnaire along with the Exploration subset from GUESS-55. 19 of the 22 Experimental participants completed the questionnaire, but only 3 of the Control participants did. This is a limitation I'll come back to.

5.2 Results

5.2.1 Survival Time

This is the headline finding. The Experimental Group survived an average of 216.0 seconds, while the Control Group survived 144.7 seconds. That's a 71.3 second gap, which means the Experimental Group lived 49% longer on average.

What's more interesting than the average is the shape of the data. 30% of Control participants died within 90 seconds. Zero Experimental participants died that fast. At the other end, 32% of Experimental participants survived past 270 seconds, compared to only 10% of Controls. The geometry didn't just make the average run longer, it created a survival floor that the colour version didn't have. Players in the Experimental Group might not have been amazing at the game, but they all stayed alive long enough to actually engage with it.

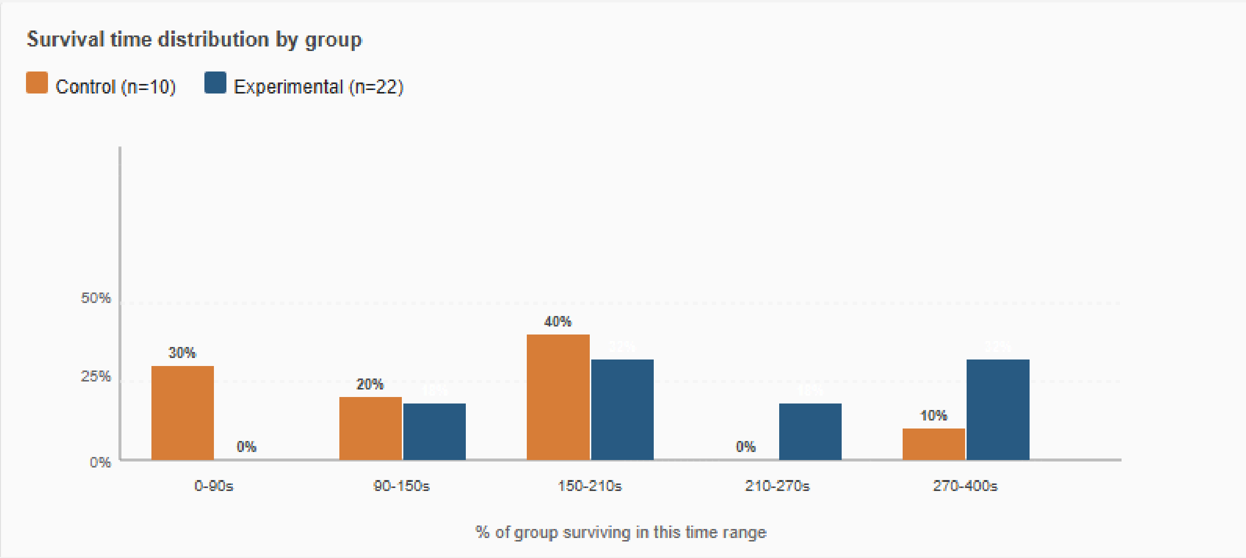


Figure2

I also looked at the variance to see how consistent each group was. The Control Group had a coefficient of variation of 53%, which is huge. The Experimental Group sat at 34%. This tells me the Control Group's outcomes were almost a coin toss; some figured out the colour system fast and survived, others fixated on one thing and died early. The Experimental Group's results clustered tighter, which I think confirms what Lynch was getting at with legibility, when an environment is readable, even weaker players are guided through it.

I ran a leave-one-out analysis on this finding to make sure no single participant was skewing it. The smallest the survival gap ever got across all iterations was 61.8 seconds, which still works out as a 43% advantage. Mean and median are within 1.3 seconds of each other (71.3 vs 72.6), so the finding isn't being dragged around by outliers.

5.2.2 Decision Accuracy Under Pressure

Group	Critical decisions	Correct	Accuracy
Control	24	8	33.3%
Experimental	74	35	47.3%
Difference	—	—	+14.0 pp

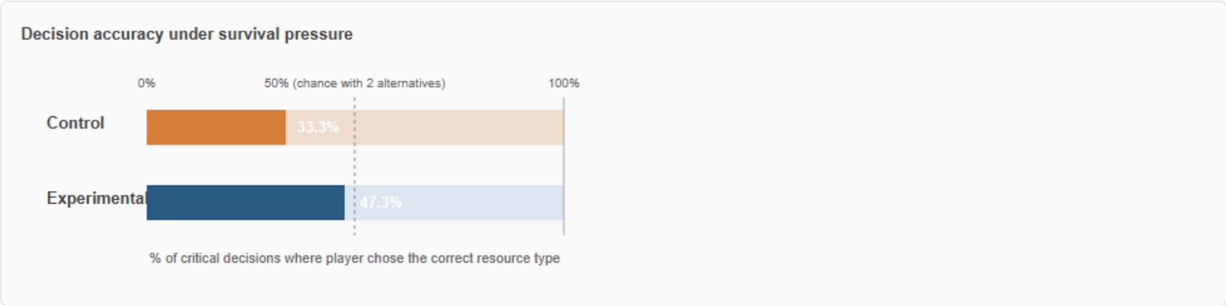


Figure 3

I wanted to know what happened when players were actually in trouble. So I looked at every island visit where one of the player's resource bars was at 35 or below at the moment they collected an island. The question was simple: did they steer to an island that addressed their most critical need?

The Control Group got it right 33.3% of the time, however this is equivalent to a coin flip as there's only 3 island types so it's 1 in 3 chance anyway which we see. The Experimental Group got it right 47.3% of the time. That's a 14 percentage point gap, and it's the cleanest evidence in the dataset that the procedural geometry is doing exactly what Gibson's affordance theory says it should. Even when players were panicking and their cognitive load was at its worst, the Experimental Group could still read the environment well enough to make the right call almost half the time. The Control Group, with the same underlying game and the same colour information, were essentially picking at random.

5.2.3 Resource Tunnel Vision

This one surprised me with how strong it came out. I checked how often each player died with at least one resource bar still healthy (above 50) while another bar had hit zero. This measures whether they were managing all three resources or fixating on just one.

70% of Control participants died with a healthy untouched bar. The Experimental Group sat at 40.9%. That's a 29 percentage point difference, which is huge. Looking at the individual cases, the Control Group had clear tunnel vision patterns. C9 (control participant 9) visited three Fuel islands in a row and died with Fuel at 87 but Food at 0. C2 died with Food at 80 and Fuel at 0. They locked onto one colour and ignored the others, sticking to what they know.

The Experimental Group showed something different. 23% of them ran all three resources to zero before dying. That's not failure, that's actually managing the full survival loop until everything genuinely ran out. Zero Control participants pulled this off. I think this is direct evidence that the procedural geometry communicates not just resource type but resource diversity. The shapes are different enough from each other that players naturally tracked which type they hadn't visited recently.

5.2.4 First-Island Choice

Each participant started in a critical state for one specific resource (noFoodState, noFuelState, or noHealthState). I wanted to know if their first island choice addressed that need, because this happens before any learning has occurred. It tells me what the player can read from the environment immediately on spawning in.

40% of the Control Group steered to a matching island first. The Experimental Group hit 59.1%. A 19 point gap on the first decision, before any in-game learning, tells me the visual affordances are readable from spawn position. The geometric language is doing real work straight away.

That said, even 59% isn't amazing. There's design room here. If the affordances were perfect, more players would steer to their needed type immediately. I think this points to a layout issue, the island matching the player's start state isn't always the most visually salient one in the opening seconds, so players sometimes get drawn to a bigger or closer island even if it doesn't match what they need.

5.2.5 Island Choice Patterns

When I looked at what kinds of islands each group preferred, an interesting pattern came out around size. The Food and Fuel islands have Big and Small variants. The Control Group went to Big islands 50% of the time and Small islands only 19% of the time. The Experimental Group basically reversed this, picking Small 46% of the time and Big 37%.

I think this is one of the most underrated findings. The Control Group went where their eye was drawn, the bigger sphere is more visually salient, so they chased it. The Experimental Group made tactical choices about size, picking what fit their current need rather than what was most obvious. A small Food island might be more efficient than a big risky one if you're not desperate, and the Experimental participants could read and make that decision. The Control did not.

For risk levels, both groups visited similar proportions of risky islands at the group level, around 50% safe, 43% risky, with very risky islands being rare for both. But within the Experimental Group, the longer-surviving players visited far fewer risky islands than the shorter-surviving ones (41% vs 59%). In the Control Group, risk-island visit rate stayed flat regardless of survival time. So in the Experimental version, strategic risk avoidance was something better players could learn. In the Control version, players couldn't really tell which spheres were risky enough for it to become a learnable skill.

5.2.6 Speed

The Experimental Group drove slower on average, 35.05 units per second compared to 42.92 in the Control. The intuitive read is that processing 3D geometry takes more cognitive effort than reading a coloured sphere, so players moved more deliberately. I'm cautious about claiming that cause directly, but the patterns within the data support it. The fastest player in the study (C9 at 60.9 u/s) had one of the shortest survival times (122s). E15 (Experimental participant 15), the slowest player by a large margin (13.4 u/s), survived 283 seconds, which is above the Experimental median. Speed without thought killed people, slow and deliberate play didn't. Supporting the fact players were looking around and making concise decisions.

5.2.7 Adjacent Island Bypass

I originally wanted to use an "adjacent island bypass" metric. This was supposed to measure horizon scanning, more bypassing means more deliberate scanning of the environment. In practice this metric didn't work well. The Control Group bypassed at 50.0%, the Experimental at 52.7%. Basically dead even. The Control Group only had 22 bypass in total, and one participant (C10, who was an unusually long survivor) contributed 7 of those. Removing C10 swings the gap from negligible to 10 points, which means the metric is too unstable to defend. The first-island choice metric does the same conceptual job (measuring deliberate visual decision-making) without these issues.

5.2.8 Questionnaire

Out of the 22 Experimental participants, 19 completed the questionnaire. Out of the 10 Control, only 4 did. This means anything I say about Control questionnaire results is more of a hint than a proper finding. I'm including the data because it points to something interesting, but I'd want a bigger sample before claiming it as proven.

The two subscales worth flagging:

Personal Gratification was the biggest gap in favour of the Experimental Group, 8.03 vs 6.67 (out of 10). This subscale captures whether players felt focused on their performance and wanted to do well. The Experimental Group felt more invested in mastering the game. The Control players, playing what is mechanically the same game, felt more like they were doing a navigation puzzle than playing a survival game.

Visual Aesthetics is the strange one. The Control Group rated it 7.0, the Experimental Group rated it 6.2. The version with plain spheres scored higher than the version with full terrain, props, and textures. All three Control respondents gave it 7 out of 10 on both visual questions. This was completely opposite to what I expected. My read on this is that minimalism reads as "clean and intentional", while busy procedural geometry can feel "noisy" if it's not communicating perfectly. Lynch's legibility works in both directions, the perfectly legible Control version is aesthetically satisfying because it's cognitively economical, even when it's functionally worse.

5.3 Discussion

So what does all of this mean for the research question? The question I asked was whether generated POIs can effectively and intuitively communicate resource affordances to players. Based on the data I have, the answer is yes, and the difference is bigger than I expected.

The survival gap alone is the clearest support. Players in the Experimental Group lived 49% longer on average. But the survival gap on its own could be explained by lots of things. What makes me confident the gap is genuinely about affordances and not about something else is the supporting findings. The Experimental Group made better critical decisions under pressure (47% vs 33%), they showed less tunnel vision (40% vs 70%), they made better first-island choices before any learning had taken place (59% vs 40%), and they made tactical size choices instead of just chasing visual salience. All of these are predictions that fall out of Gibson's affordance theory directly. If procedural geometry is communicating resource type, risk, and utility to the player at a glance, you'd expect exactly these patterns. And that's what the data shows.

Where it gets interesting is the questionnaire. If we'd only had the questionnaire and not the telemetry, we'd have come to the wrong conclusion. The Control players, the ones with worse outcomes, rated the visuals higher. They felt more confident, they liked how the game looked, the simple spheres felt clean and readable. But they died. The procedural geometry was harder to subjectively appreciate but objectively worked better. I think this is actually one of the more valuable findings of the project, because it speaks to a tension in level design that doesn't get talked about much. Lynch's legibility predicts that simple, clean environments will feel better to navigate. Gibson's affordances predict that environments rich in functional cues will guide better behaviour. In my experiment these two theories made opposite predictions, and the data shows that affordances win out for behaviour while legibility wins out for subjective experience. They're both right, but they're measuring different things.

I should be honest about the limitations. The biggest one is the Control Group questionnaire sample, only 3 participants, which means any group-level claim about subjective experience is on shaky ground. The second limitation is that I only had each participant play one run. There's no way to tell from this data whether the gap would shrink with practice. The Experimental advantage I measured is the immediate, no-learning advantage, which is what the research question was about, but it'd be worth running again with multiple sessions to see if that gap holds.

There are a few things I'd change if I were doing this again. The first-island match rate of 59% in the Experimental Group tells me that even in the version with full visual information, players couldn't always immediately spot the island matching their critical start state. Players are spawning into a busy generated scene and the most visually salient island is whichever is biggest and closest, regardless of resource type. If I were redesigning, I'd want the spawn algorithm to bias toward placing the player's needed island within their initial line of sight, or I'd want to give starting needs a stronger UI representation so they're really felt as a need. The procedural affordances do excellent work once the player is engaged, but they let players down at the moment of spawn.

I'd also want to look at whether the size effect I noticed (Control chasing Big, Experimental choosing tactically) replicates in a bigger study. It's one of the more interesting findings because it suggests procedural geometry isn't just communicating "what" but also "how much" and "is this worth it for me right now." That's a richer affordance vocabulary than I was even trying to test for.

6. Conclusion

This project set out to find out if procedurally generated points of interest can intuitively communicate resource affordances to players, and based on the data I collected, they can, and they do it more effectively than I expected.

Players in the Experimental Group survived 49% longer than the Control Group. They made better critical decisions when their resources were running out. They managed multiple resources at once instead of fixating on a single colour. They picked the right islands faster from the very first decision, before any learning could have happened. And they chose island sizes tactically based on what they needed, not just what was biggest and most visible. These are all things you'd expect to see if Gibson's idea of direct perception is right, that players don't have to consciously decode a 3D environment to make sense of it, they just read it.

What I didn't expect was the questionnaire result, where the Control Group actually rated the visuals higher despite performing worse. The plain-coloured-sphere version felt cleaner, more confident, easier to look at. The procedural version felt noisier even though it was the version that kept people alive. I think this is the most academically useful finding because it shows that what looks good and what works aren't always the same thing. That tension between Lynch's legibility (making things readable and pleasant) and Gibson's affordances (making things functionally useful) is something that level designers run into all the time, and this experiment ended up showing the trade-off pretty clearly.

If I were extending this project I would want to run a longer study with multiple play sessions to see if the gap holds once players have time to learn. I'd also want to fix the spawn placement so that the player's needed island is more reliably visible from the start. And I'd want a bigger Control Group questionnaire sample to be able to actually defend the aesthetic finding properly.

Overall, I'm confident the procedural generation tool I built does what I set it out to do. Players can read the environment and act on it without UI. It's not perfect, the first-island match rate tells me there's still work to do at spawn, but the rest of the gameplay loop holds up. POIs can communicate resource affordances, and the difference is large enough to be worth the development effort.

7. References

Papers

Csikszentmihalyi, M., 1990. Flow: The Psychology of Optimal Experience. New York: Harper & Row.

Gibson, J.J., 1979. The Ecological Approach to Visual Perception. Boston: Houghton Mifflin.

Keebler, J.R., Shelstad, W.J., Smith, D.C., Chaparro, B.S. and Phan, M.H., 2020. Validation of the GUESS-18: A short version of the game user experience satisfaction scale (GUESS). Journal of Usability Studies, 16(1), pp.49-62.

Lynch, K., 1960. The Image of the City. Cambridge, MA: MIT Press.

Ludography

Phan, M.H., Keebler, J.R. and Chaparro, B.S., 2016. The development and validation of the game user experience satisfaction scale (GUESS). Human Factors, 58(8), pp.1217-1247.

Totten, C.W., 2014. An Architectural Approach to Level Design. Boca Raton, FL: CRC Press.

Software & Video Games

FromSoftware, 2022. Elden Ring. [Video game]. Tokyo: Bandai Namco Entertainment.

Rare, 2018. Sea of Thieves. [Video game]. Redmond, WA: Xbox Game Studios.

Sucker Punch Productions, 2020. Ghost of Tsushima. [Video game]. San Mateo, CA: Sony Interactive Entertainment.

Ubisoft Bordeaux, 2023. Assassin's Creed Mirage. [Video game]. Montreuil: Ubisoft.

Unity Technologies, 2022. Unity (Version 2022.3) [Software]. San Francisco, CA: Unity Technologies. (Note: Update the year/version to the exact version of Unity you used!)

Unknown Worlds Entertainment, 2018. Subnautica. [Video game]. San Francisco, CA: Unknown Worlds Entertainment.

AI Tools

Anthropic, 2026. *Claude*. [Large Language Model]. Available at: https://claude.ai [Accessed 28 Apr. 2026]. - for feedback and data formatting.

Google, 2026. *Gemini*. [Large Language Model]. Available at: https://gemini.google.com [Accessed 28 Apr. 2026]. - structure, spelling and grammar.

Unity Assets

AdobeMano, 2025. *3D Stylized Pirate Island Props Pack*. [Unity Asset]. Unity Asset Store. Available at: https://assetstore.unity.com/packages/3d/props/3d-stylized-pirate-island-props-pack-336304 [Accessed 28 Apr. 2026].

AureDevGames, 2026. *Aquis Water (Toon Shader) Orto/Perspective*. [Unity Asset]. Unity Asset Store. Available at: https://assetstore.unity.com/packages/vfx/shaders/aquis-water-toon-shader-orto-perspective-297566 [Accessed 28 Apr. 2026].

Dmitrii Kutsenko, 2019. *PBR RPG FPS Game Assets Industrial Set v1.0*. [Unity Asset]. Unity Asset Store. Available at: https://assetstore.unity.com/packages/3d/environments/industrial/pbr-rpg-fps-game-assets-industrial-set-v1-0-146519 [Accessed 28 Apr. 2026].

Gabro Media, 2016. *Barrels*. [Unity Asset]. Unity Asset Store. Available at: https://assetstore.unity.com/packages/3d/props/barrels-32975 [Accessed 28 Apr. 2026].

JustCreate, 2023. *Low Poly Tropical Island Lite*. [Unity Asset]. Unity Asset Store. Available at: https://assetstore.unity.com/packages/3d/environments/low-poly-tropical-island-lite-242437 [Accessed 28 Apr. 2026].

8. APPENDICES

Appendix A: Participant Ethics form

UNIVERSITY *of* GREENWICH
Faculty of Engineering & Science

PARTICIPANT CONSENT FORM

Title of Project:
How the use of points of interests can influence player decisions through an environment

Name of Researcher: Joshua Gorton

To be completed by the participant.

- I have read the information sheet about this study
- I have had an opportunity to ask questions and discuss this study
- I have received satisfactory answers to all my questions
- I have received enough information about this study
- I understand that I am / the participant is free to withdraw from this study:
  - At any time (until such date as this will no longer be possible, which I have been told)
  - Without giving a reason for withdrawing
  - (If I am / the participant is, or intends to become, a student at the University of Greenwich) without affecting my / the participant's future with the University
- I understand that my research data may be used for a further project in anonymous form, but I am able to opt out of this if I so wish, by ticking here. ☐
- I agree to take part in this study ☐
- I am over 18 years old ☐

Signed (participant):	Date 00/00/2026
Participant name in block letters:	
Signature of researcher: J.gorton	Date 00/00/2026
This project is supervised by: Peter Blenkharn - peter.blenkharn@greenwich.ac.uk	
Researcher's contact details: Joshua Gorton: jg6582v@gre.ac.uk	

Appendix B: Tables and testing results

RAW TELEMETRY — ALL 32 PARTICIPANTS										
ID	Group	Play Time (s)	Distance	Avg Speed	Start State	Final H	Final F	Final Fuel	Islands	Island Names
C1	Control	172.01	6882.8	40.015	noHealthState	0	40	0	4	Health__Risky, Food_Big__no, Food_Small_no, Health__no
C2	Control	84.33	4371.2	51.832	noFuelState	0	80	2.539	1	Food_Big__Risky
C3	Control	166.58	5166.2	31.013	noFuelState	0	0	73.684	3	Fuel_Big_Risky, Fuel_Big_no, Fuel_Big_no
C4	Control	171.28	6254.3	36.514	noHealthState	0	0	100	3	Health__Risky, Fuel_Big_Risky, Fuel_Big_Risky
C5	Control	91.03	3147.8	34.58	noFuelState	0	60	0	2	Health__Risky, Food_Big__no
C6	Control	188.06	8589.9	45.675	noFuelState	0	0	12.319	4	Fuel_Big_no, Health__Risky, Fuel_Small_no, Fuel_Small_Risky
C7	Control	59.05	2661.4	45.074	noHealthState	0	60	39.826	2	Food_Big__no, Health__VeryRisky
C8	Control	75.06	3005.1	40.035	noFuelState	0	30	0	2	Health__no, Health__Risky
C9	Control	121.67	7415.4	60.946	noFoodState	0	0	87.566	3	Fuel_Big_no, Fuel_Big_no, Fuel_Big_no
C10	Control	317.88	13818.1	43.47	noHealthState	0	0	59.17	8	Food_Big__no, Fuel_Small_Risky, Health__Risky, Food_Big__no, Food_Small_no, Food_Small_Risky, Fuel_Big_Risky, Health__VeryRisky
E1	Experimental	106.22	3308.8	31.149	noFoodState	0	0	66.038	2	Fuel_Big_Risky, Fuel_Big_no
E2	Experimental	159.52	5213.5	32.684	noFoodState	0	0	0	3	Fuel_Small_Risky, Fuel_Small_no, Food_Small_Risky
E3	Experimental	112.24	3486.6	31.063	noHealthState	0	20	81.198	2	Fuel_Big_Risky, Fuel_Big_Risky
E4	Experimental	152.11	4403.8	28.952	noFoodState	0	60	19.157	4	Fuel_Big_no, Health__no, Fuel_Big_no, Food_Big__Risky
E5	Experimental	180.82	6178.9	34.172	noFoodState	0	0	0	3	Food_Small_no, Fuel_Small_Risky, Health__Risky
E6	Experimental	119.76	3761.2	31.406	noFoodState	0	0	100	3	Fuel_Big_no, Fuel_Small_no, Fuel_Big_Risky
E7	Experimental	174.35	6974.1	40.001	noFoodState	0	30	0	2	Food_Big__Risky, Food_Small_no
E8	Experimental	200.06	9677.4	48.373	noFuelState	0	0	71.258	4	Fuel_Small_no, Fuel_Small_Risky, Fuel_Small_Risky, Fuel_Big_Risky
E9	Experimental	171.83	4899.1	28.511	noHealthState	0	20	0	3	Health__Risky, Food_Small_Risky, Food_Small_Risky
E10	Experimental	303.24	11084.2	36.553	noFuelState	0	0	53.264	6	Fuel_Big_Risky, Food_Small_Risky, Health__Risky, Food_Small_no, Fuel_Big_no, Fuel_Big_no
E11	Experimental	103.16	3242.1	31.428	noHealthState	0	50	47.326	2	Food_Small_no, Health__VeryRisky
E12	Experimental	232.74	5278.1	22.679	noHealthState	0	0	100	4	Health__Risky, Food_Small_no, Fuel_Small_Risky, Fuel_Big_Risky
E13	Experimental	300.74	11505.1	38.256	noFoodState	0	0	91.087	7	Food_Small_no, Fuel_Big_no, Health__Risky, Food_Small_no, Fuel_Big_no, Fuel_Big_no, Fuel_Big_no
E14	Experimental	241.97	6644.5	27.459	noFoodState	0	40	55.806	4	Food_Small_no, Fuel_Big_no, Food_Small_no, Food_Small_no
E15	Experimental	283.44	3789.9	13.371	noFuelState	0	0	0	3	Food_Small_no, Fuel_Small_no, Health__Risky
E16	Experimental	246.92	8754.7	35.455	noFuelState	0	0	0	5	Fuel_Small_Risky, Fuel_Big_Risky, Health__no, Food_Big__no, Health__VeryRisky
E17	Experimental	273.23	9559.7	34.988	noFoodState	0	0	4.205	5	Food_Big__no, Food_Big__Risky, Fuel_Small_Risky, Food_Big__no, Health__VeryRisky
E18	Experimental	200.73	9965.5	49.647	noFoodState	0	10	0	3	Fuel_Small_no, Food_Big__no, Food_Small_Risky
E19	Experimental	263.44	11555.5	43.864	noFuelState	0	0	12.359	4	Food_Big__Risky, Fuel_Small_Risky, Fuel_Big_no, Food_Small_no
E20	Experimental	359.28	14272.1	39.724	noFoodState	0	30	20.709	8	Food_Small_no, Food_Big__no, Fuel_Big_no, Food_Big__no, Health__Risky, Food_Small_no, Fuel_Big_no, Food_Small_no
E21	Experimental	285.28	12957.9	45.422	noHealthState	0	0	0	6	Health__Risky, Fuel_Small_Risky, Food_Small_no, Food_Small_Risky, Fuel_Small_no, Health__no
E22	Experimental	281.48	12951.1	46.012	noFoodState	0	0	37.588	6	Food_Small_Risky, Food_Small_no, Fuel_Big_no, Health__no,

GROUP STATISTICS — BOTH GROUPINGS COMPARED							
PRIMARY GROUPING — Researcher's stated assignment | Control n=10 (C1–C10) | Experimental n=22 (E1–E22)							
Metric	Ctrl Mean	Ctrl SD	Ctrl CV%	Exp Mean	Exp SD	Exp CV%	Exp−Ctrl
Play Time (s)	144.7	76.77	53.1	216.02	73.64	34.1	71.33
Distance (units)	6131.22	3368.22	54.9	7702.9	3620.35	47	1571.68
Avg Speed (u/s)	42.92	8.73	20.3	35.05	8.67	24.7	-7.86
Islands Visited	3.2	1.93	60.4	4.05	1.7	42.1	0.85

CRITICAL NEED ACCURACY — when resource ≤ 35 at collection, did				
Group	n	Critical Visits	Accurate	Accuracy%
Control	10	24	8	33.3%
Experimental	22	74	35	47.3%

DECISION ACCURACY & ADJACENT ISLAND BYPASS				
ADJACENT ISLAND BYPASS RATE — did player skip closest island to go further out?				
Group	n	Opportunities	Bypasses	Bypass Rate%
Control	10	22	11	50.0%
Experimental	22	67	34	50.7%

ISLAND VISIT FREQUENCY ANALYSIS					
RESOURCE TYPE SPLIT (primary grouping: n=10 ctrl / n=22 exp)					
Resource	Ctrl Visits	Ctrl %	Exp Visits	Exp %	Diff (ppts)
Food	9	28.1	35	39.3	11.2
Fuel	13	40.6	39	43.8	3.2
Health	10	31.2	15	16.9	-14.4
RISK LEVEL SPLIT					
Risk Level	Ctrl Visits	Ctrl %	Exp Visits	Exp %	Diff (ppts)
Safe	16	50	48	53.9	3.9
Risky	14	43.8	38	42.7	-1.1
Very Risky	2	6.2	3	3.4	-2.9

Appendix C: Questionnaire and results

A	B	C
Item	Category	Evaluation Statement
1	Usability/Playability	I find the game's controls to be straightforward.
2	Usability/Playability	I find the game's interface easy to navigate.
3	Narratives	I am captivated by the game's story.
4	Narratives	I enjoy the fantasy or story provided by the game.
5	Play Engrossment	I feel immersed in the game world.
6	Play Engrossment	I lose track of time when playing this game.
7	Enjoyment	I think the game is fun.
8	Enjoyment	I enjoy playing this game.
9	Creative Freedom	I feel the game allows me to be imaginative.
10	Creative Freedom	I feel creative while playing the game.
11	Audio Aesthetics	The audio in the game enhances the experience.
12	Audio Aesthetics	I enjoy the game's audio (e.g., sound effects, music).
13	Personal Gratificati	I want to do as well as possible in the game.
14	Personal Gratificati	I feel successful when I overcome the obstacles in the game.
15	Social Connectivity	I enjoy playing this game with other people.
16	Social Connectivity	I like to play this game with others.
17	Visual Aesthetics	The game is visually appealing.
18	Visual Aesthetics	The visual style of the game enhances the experience.
19	Exploration & Affor	I could easily identify what resources an island held from a distance.
20	Exploration & Affor	The shape and colour of the islands accurately communicated their danger level.
21	Exploration & Affor	I felt confident in my navigational choices without the use of text UI.

GUESS-18 + EXPLORATION SUBSET — RAW SCORES (0–10 scale)

PID	Group	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	Q23
Subscale		Usability	Usability	Narratives	Narratives	Engrossment	Engrossment	Enjoyment / Fun	Enjoyment / Fun	Creative Freedom	Creative Freedom	Audio Aesthetics	Audio Aesthetics	Personal Gratification	Personal Gratification	Social Connectivity	Social Connectivity	Visual Aesthetics	Visual Aesthetics	Exploration	Exploration	Autonomy	Autonomy	Curiosity
									REVERSE →															
E1	Experimental	7	6	0	0	4	4	5	5	2	1	0	0	8	9	0	0	5	5	5	5	4	7	6
E2	Experimental	8	8	2	4	7	7	3	3	3	3	4	4	6	3	2	2	5	3	4	3	2	2	2
E3	Experimental	10	10	8	8	10	10	10	0	8	10	0	0	10	10	5	5	5	5	8	10	5	10	8
E4	Experimental	6	2	1	1	6	7	5	4	4	2	5	5	6	6	3	2	6	6	8	9	6	5	7
E5	Experimental	8	8	2	6	5	2	7	2	3	3	0	0	6	8	5	0	6	6	10	8	3	6	7
E6	Experimental	7	8	7	7	7	7	8	2	6	5	5	5	8	9	0	5	7	7	8	4	5	7	7
E7	Experimental	9	8	2	0	10	7	7	3	4	4	0	0	7	9	1	8	7	6	8	10	3	5	5
E8	Experimental	5	5	4	5	5	0	7	7	7	4	5	5	6	6	5	2	7	7	6	8	7	7	7
E9	Experimental	10	9	5	5	7	7	10	0	8	5	0	0	7	10	0	0	7	7	8	10	5	8	10
E10	Experimental	8	8	5	6	6	8	9	2	2	6	0	0	9	10	3	6	7	7	9	7	7	7	9
E11	Experimental	7	5	0	0	5	3	6	10	2	1	0	1	5	7	0	2	5	6	8	2	5	3	5
E12	Experimental	10	8	3	6	4	4	7	2	5	4	1	1	7	7	4	8	6	5	7	9	7	6	8
E13	Experimental	10	10	5	5	10	5	10	0	5	6	5	0	10	10	5	5	10	10	10	10	5	10	10
E14	Experimental	9	8	2	5	6	2	8	2	7	3	0	0	7	9	7	6	3	2	8	9	1	6	7
E15	Experimental	8	7	0	4	8	7	9	8	7	9	5	5	10	10	5	10	4	3	7	7	8	9	9
E16	Experimental	10	10	10	10	10	10	10	10	10	10	5	5	10	10	0	0	10	10	10	0	5	7	3
E17	Experimental	10	8	7	6	4	4	8	8	5	3	5	5	6	7	2	0	5	5	8	6	5	4	4
E18	Experimental	10	10	5	0	10	10	10	10	10	10	0	0	10	10	0	0	10	10	10	10	5	10	10
E20	Experimental	7	6	6	5	7	6	9	0	6	6	5	5	8	9	5	5	6	6	8	8	5	8	8
CONTROL GROUP (n=3 — small sample; treat comparisons as indicative only)																								
C3	Control	7	7	8	5	7	5	8	7	5	2	0	0	10	10	5	7	7	7	5	6	1	0	6
C4	Control	10	10	7	7	7	7	8	1	8	8	0	0	10	10	5	5	7	7	10	10	7	10	10
C10	Control	8	8	0	0	5	2	6	5	5	5	0	0	0	0	0	0	7	7	10	8	7	8	7
GROUP AVERAGES (raw — Q8 not yet reversed; see Sheet 7 for subscale analysis)																								
Experimental avg (n=19)		8.37	7.58	3.89	4.37	6.89	5.79	7.79	4.11	5.47	5.00	2.37	2.16	7.68	8.37	2.74	3.47	6.37	6.11	7.89	7.11	4.89	6.68	6.95