EcoVR

A Fun and Interactive Virtual Ecosystem Simulator!

https://youtu.be/C2f6LHMTQps

01

Implementation

Tech Stack

Unity

C#

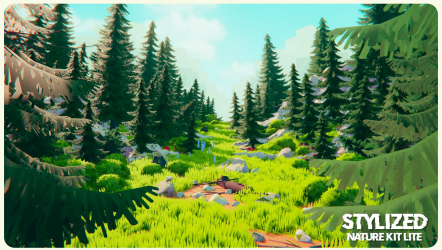
Meta Quest 3

XR Device Simulator

Git

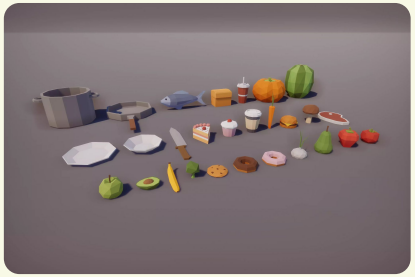
Assets

Sci-Fi Styled Modular Pack

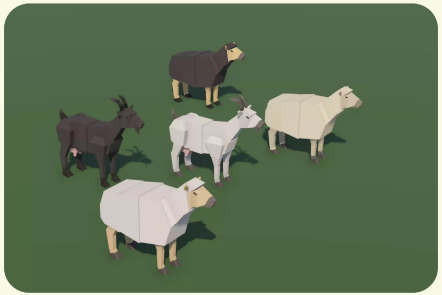


Stylized Nature Kit Lite

TENKOKU Dynamic Sky



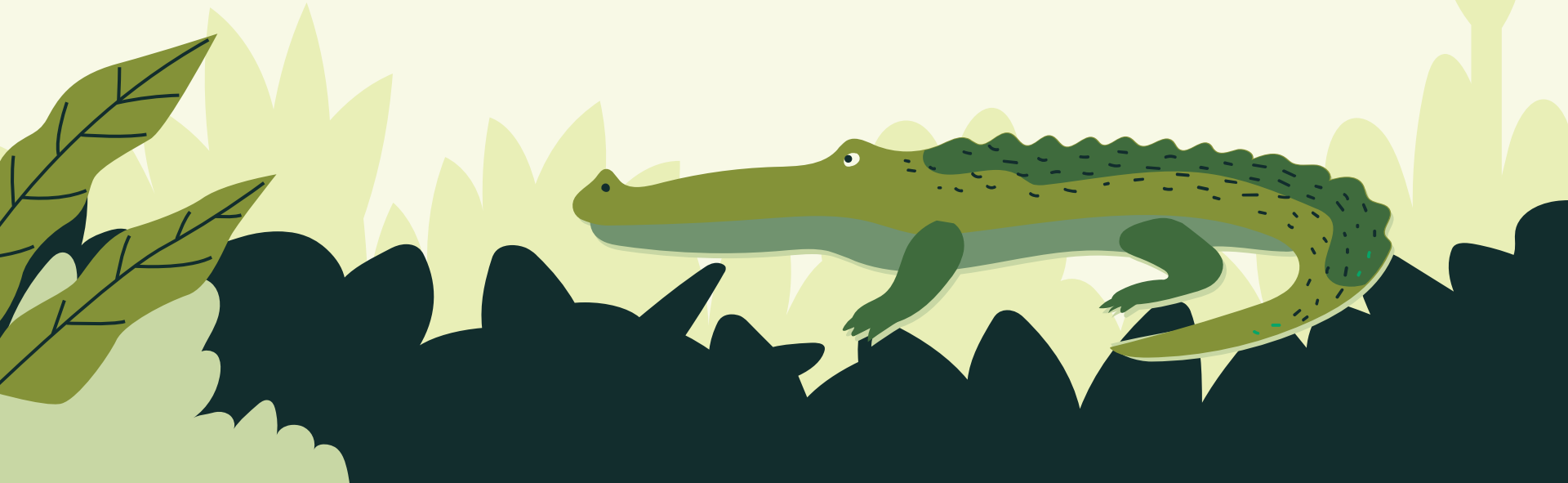
Low Poly Food Lite



Animals

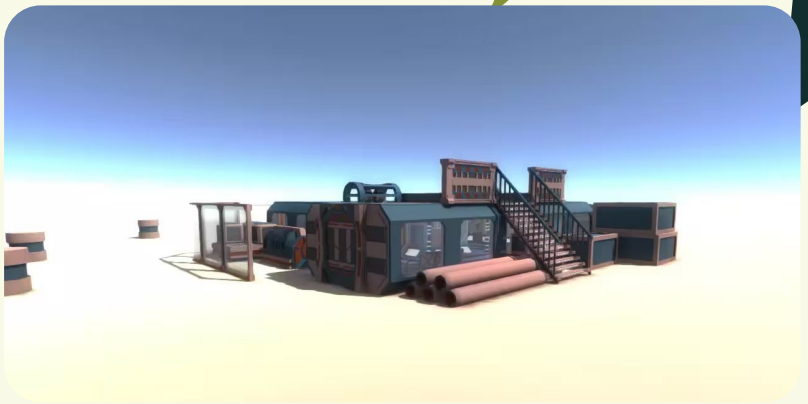
Dreamscape Mountains

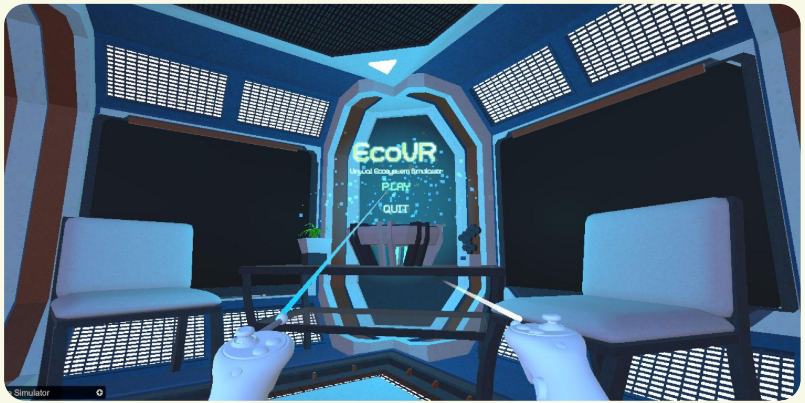
Game Features 02



Main Menu Scene

- Immersive high-tech pod set in a stylized desert
- Visual transition from barren desert to lush forest enhances discovery
- Main menu displayed via holographic projection
- Buttons use proximity-based prompts to guide interaction





Game Scene

bird-eye view

first-person view

🐦 Bird's-Eye Mode

- Top-down camera for macro ecosystem observation
- Spawn Buttons to add predators/prey
- Weather Buttons to change climate conditions
- Time Speed Slider to fast-forward simulation
- Zoom Slider for focused or broad view
- Ideal for experimentation and scenario testing

First-Person Mode

- Ground-level immersion as in-world observer
- Free movement & teleportation
- Tools like binoculars & magnifiers for animal observation
- Feed/hydrate animals to influence behavior
- Personal, real-time interaction with the ecosystem

Tutorial System

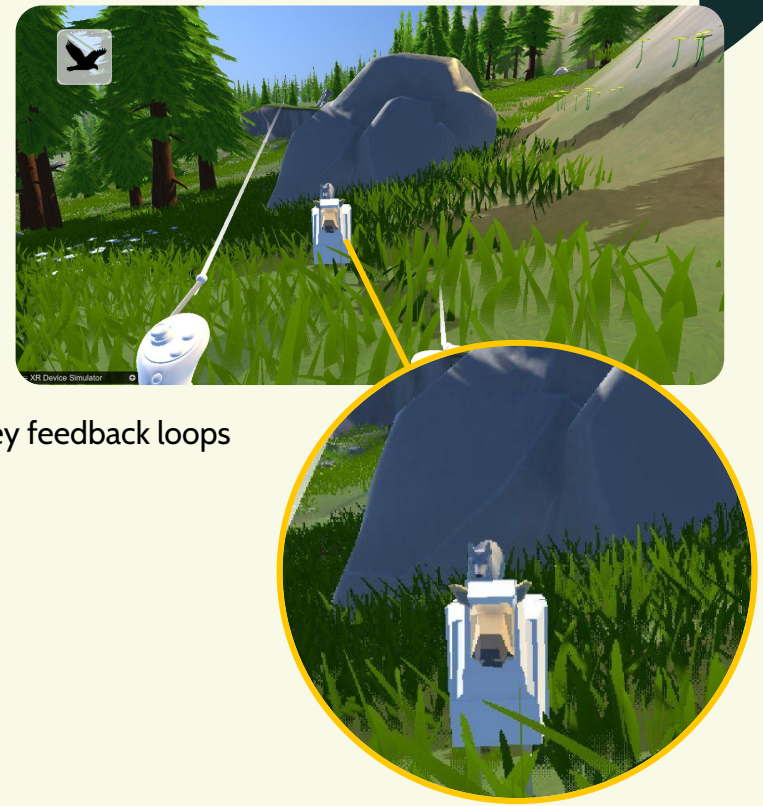
- Activated upon entering the game scene
- Guides users through key UI elements and controls:
  - Spawn animals (predators/prey)
  - Change weather conditions
  - Adjust time speed via bottom slider
  - Zoom in/out with vertical slider
- Visual highlights for each feature
- Self-paced and interactive walkthrough

Natural Birth & Death (Conway Game of Life)

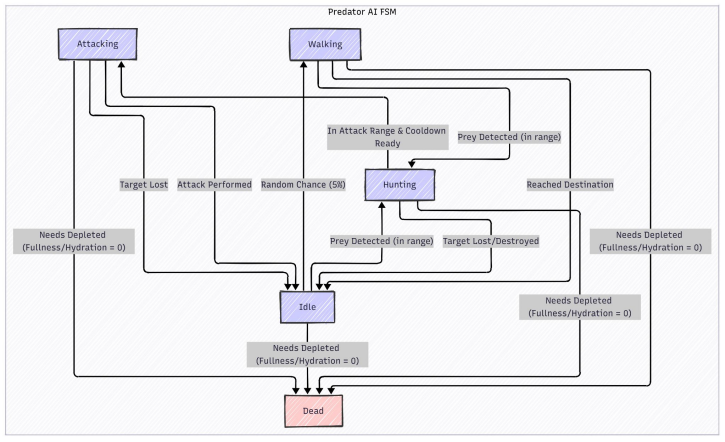
- Based on spatial rules inspired by Conway's Game of Life
- Population evaluated every 10 seconds using Unity's physics-based queries
  - Physics.OverlapSphere vs iterating all animals in the scene (O(n^2))
- Death Conditions:
  - Loneliness: dies with too few neighbors
  - Overpopulation: dies with too many nearby animals competing
- Birth Condition:
  - Proximity-based reproduction with nearby mates (50% chance)
- Species-specific tuning (e.g., predators = larger radii, prey = more tolerant)
- Emergent, unscripted population dynamics

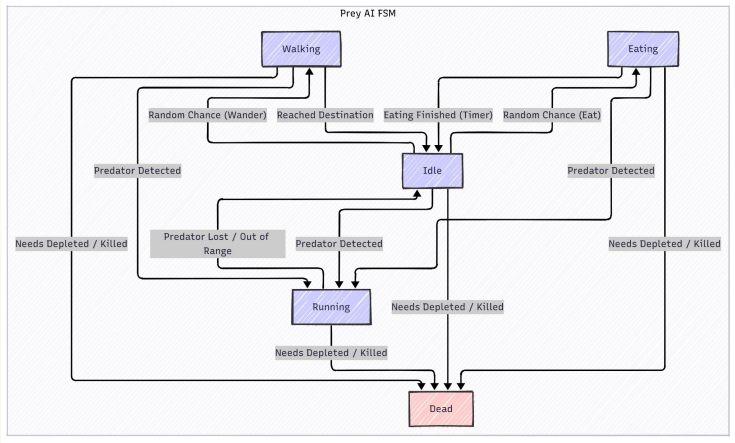
Predator Prey Dynamics (Animal Intelligence)

- Finite State Machine AI for both predators and prey
- Predators:
  - States: Idle, Walking, Hunting, Attacking, Dead
  - Uses NavMesh to chase and attack prey
  - Hunger system: die if not fed in time
- Prey:
  - States: Idle, Walking, Eating, Running, Dead
  - Roams, grazes, and flees from predators
  - Directional escape based on predator location
- Result: Self-balancing ecosystem with emergent predator-prey feedback loops



🐺 Predator Prey Dynamics (Animal Intelligence)





Hunger & Hydration Systems

- Models biological needs for realism
- Hunger:
  - Increases over time; drives behavior (e.g., hunting, grazing)
  - Starvation leads to death if not resolved
- Hydration:
  - Depletes faster under hot weather
  - Death from dehydration if not replenished
- Encourages purposeful animal behavior
- Deepens environmental interdependence

👁️ Detection Radius Visualisation

- Detection Radius Visualisation
  - Makes animal perception ranges visible in the simulation
  - Enhances understanding of predator–prey interactions
- Visual Indicators
  - 🔴 Red zones: Predator detection radius
  - 🔵 Blue zones: Prey detection radius
  - Helps visualize reaction triggers (e.g., chase or flee)
- Toggleable Display
  - UI button (top-center) enables/disabled radius overlay
  - Keeps interface clean and user-controlled
- Best Used in Bird's Eye View
  - Ideal for observing spatial dynamics and environmental effects
  - Shows impact of weather (e.g., fog) on visibility

Weather System

- Built with Tenkoku Dynamic Sky for realistic weather & time transitions
- Adds environmental challenges and affects animal behavior
- Weather Types & Effects:
  - Sunny:
    - ➜ Common during daytime
    - ➜ Decreases animal hydration over time
  - Rainy:
    - ➜ Replenishes hydration
    - ➜ Reduces detection radius & movement speed
  - Foggy:
    - ➜ Greatly limits visibility and mobility
  - Snowy:
    - ➜ Boosts hydration
    - ➜ Reduces detection & speed
    - ➜ Increases hunger due to cold-induced energy use
- Influences predator–prey dynamics and survival realism
- Encourages strategic movement and adaptive behavior in animals

🔍 Binoculars & Magnifying Glasses

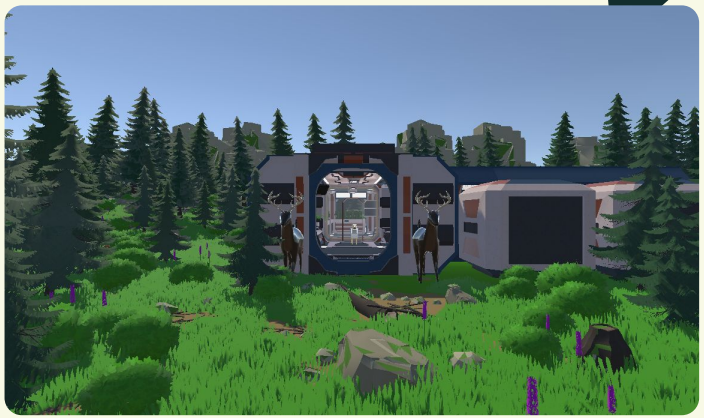
- Enables realistic zooming in VR for distant or small objects
- Designed for high fidelity & full VR compatibility
- 🛠️ Design Considerations
  - ❌ Main Camera FOV – simple but not VR-friendly
  - ❌ Shader Zoom – lightweight but low resolution
  - ✅ Render Texture + Secondary Camera – high quality, chosen method
- ⚙️ Implementation Highlights
  - Secondary Camera mounted in front of lens
  - Render Texture displays zoomed view on lens surface
  - Custom FOV for zoom (e.g., 15° for binoculars, 7° for magnifying glass)
  - Toggle On/Off with controller input (ToggleItemsController)
  - Attached to hand for natural, physical interaction

🍎 Feeding & Water Interaction

- Adds hands-on way to support animal hunger and hydration
- Enhances engagement with the ecosystem simulation
- 🎮 User Mechanics
  - Pick up food/water items and deliver to nearby animals
  - Animals auto-consume when in range
  - Affects internal hunger and hydration levels
- ⚙️ Implementation
  - Items use colliders + triggers to detect animal proximity
  - Stats updated via animal scripts upon interaction
- 🎓 Educational Value
  - Promotes empathy and active participation
  - Lets users test survival impact of care
  - Encourages reflection on ecological responsibility

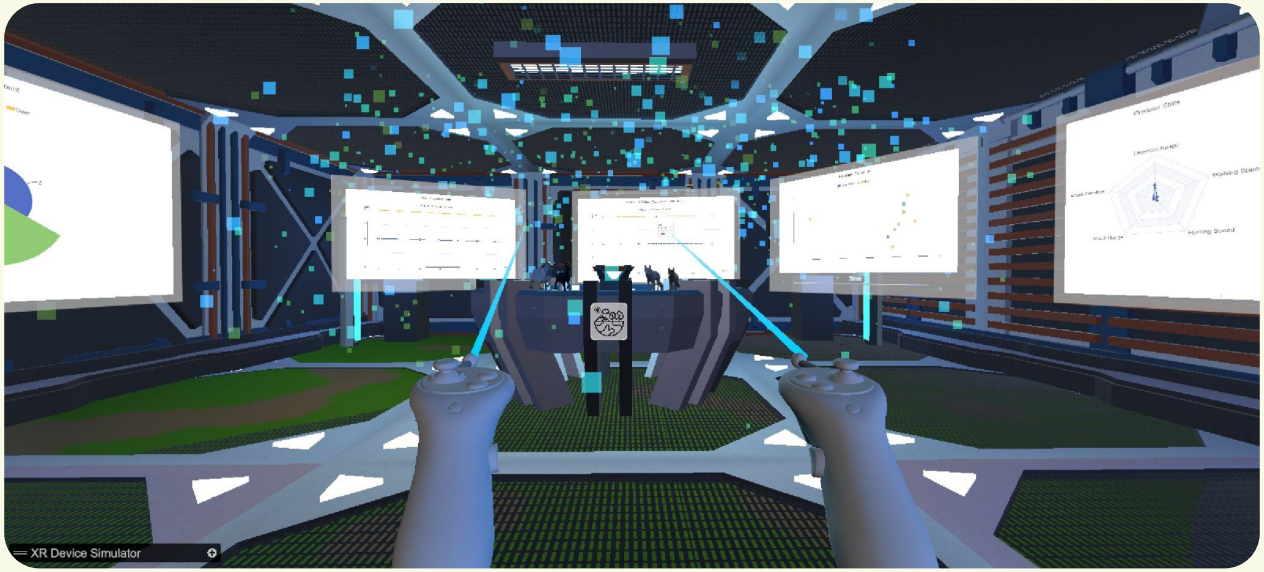
📊 Analytics Scene

- Visualizes animal behavior and population trends
- Set in a futuristic forest lab to maintain narrative immersion
- 🌲 Immersive Forest + Central Pod
  - User spawns in a forest facing a high-tech observation pod
  - Smooth transition from ecosystem exploration to data analysis
- 🐾 Animals Inside the Pod
  - Enclosed animal displays simulate a wildlife research facility
  - Reinforces subject focus and adds realism & liveliness
- 🧪 Futuristic Analytics Lab
  - Sleek pod interior with interactive data panels
- Promotes exploration-based learning through visual data interaction



📈 Ecosystem Data Analytics

- Real-time tracking of animal behavior and population (records every 5s)
- Data preprocessed for species grouping & name cleanup
- Visualizations via XCharts



📈 Ecosystem Data Analytics

Population Snapshot – Pie Chart

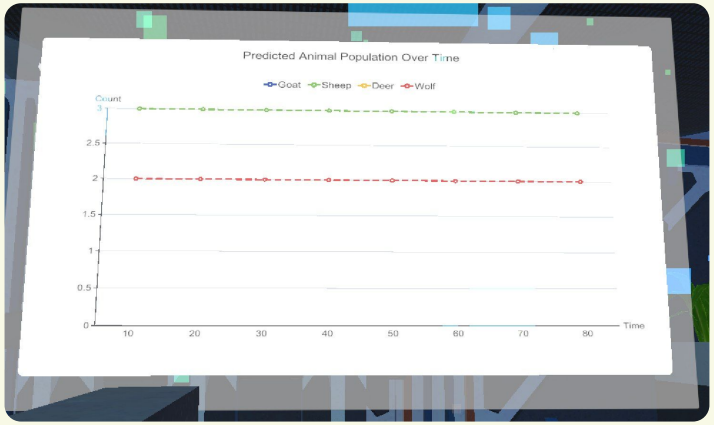
Shows proportion of prey vs predators at current moment

Population Over Time – Line & Area Chart

Prey: solid lines with circles

Predators: area chart with triangles

📈 Ecosystem Data Analytics

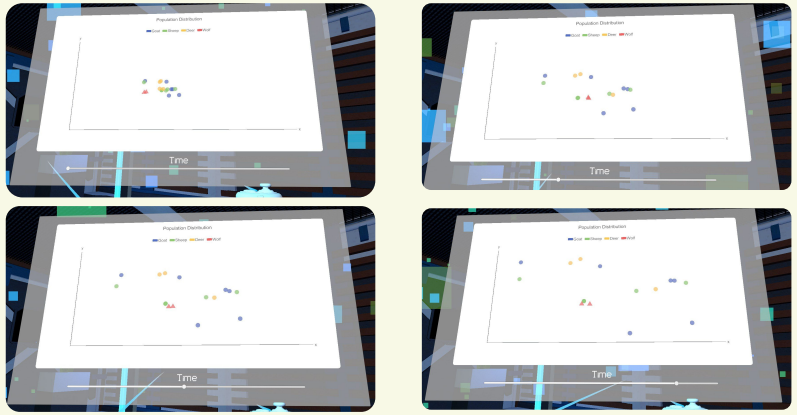


Forecast – ARIMA Prediction

Predicts future population trends with dotted line

Based on historical data

$$\hat{y}_t = \alpha + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t$$



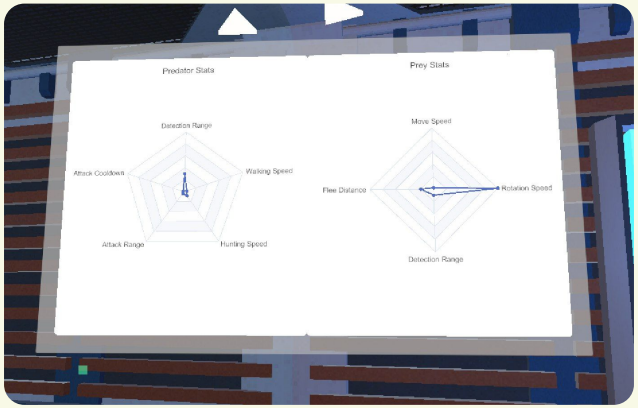
Animal Movement – Scatter Heatmap

Spatial distribution over time

Slider to view specific timestamps

Prey = circles, predators = triangles

📈 Ecosystem Data Analytics



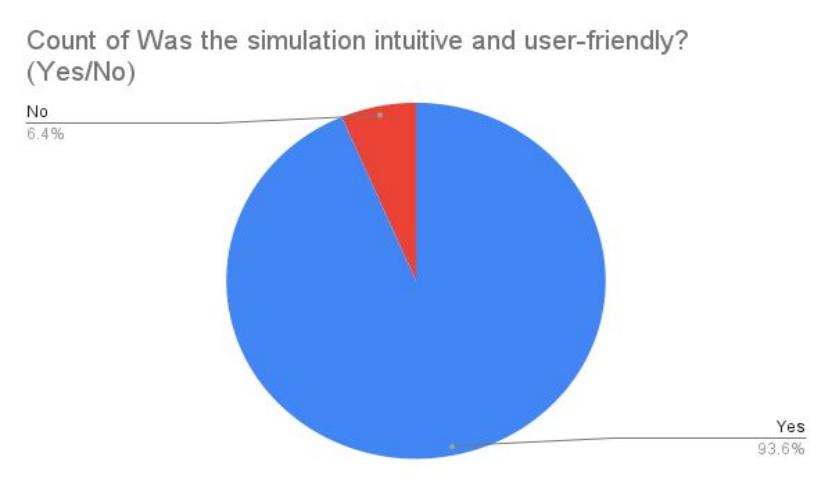
Attribute Comparison – Radar Chart

Predators: Detection, Speed, Attack traits

Prey: Movement, Rotation, Detection, Flee distance

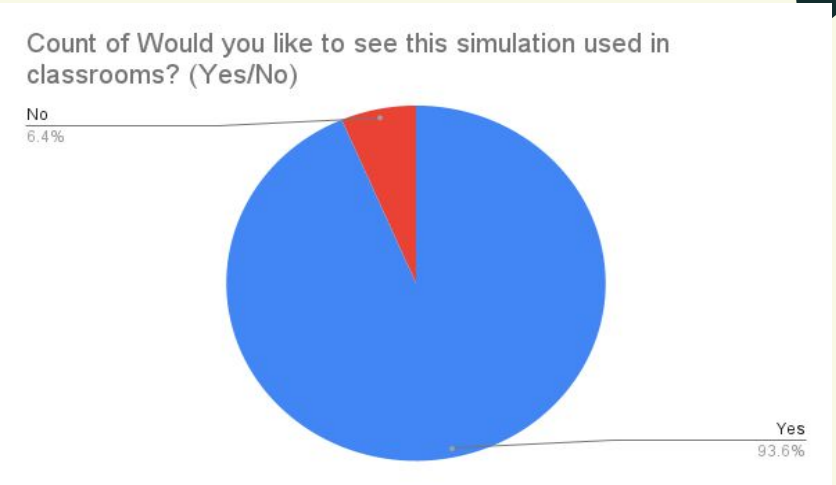
03 Survey Results

📈 Simulation Survey



Is EcoVR user friendly?

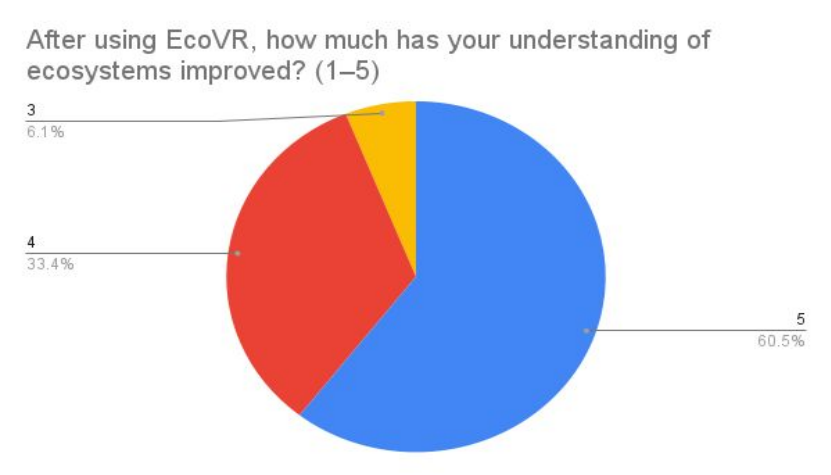
The majority find our application easy to use



Should EcoVR be used in classrooms?

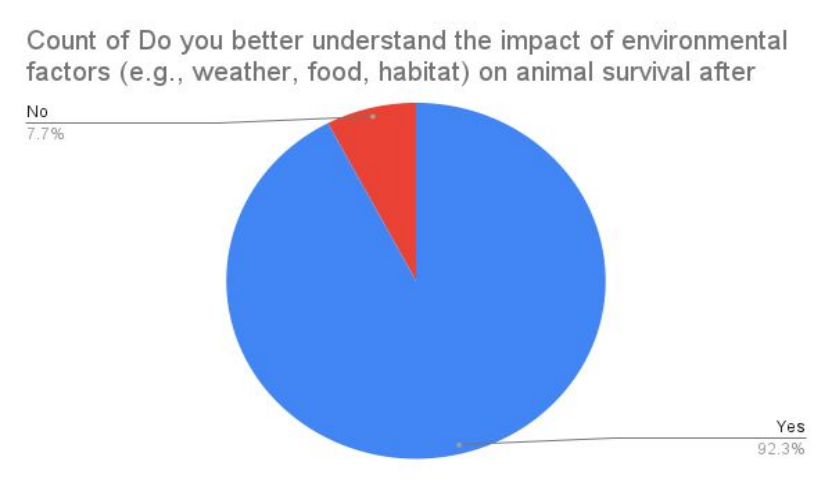
Majority support the idea of using this simulation in classrooms

📈 Simulation Survey



Can users learn more from EcoVR?

Majority of the users rated 4 to 5 out of 5



Can EcoVR help users understand ecosystem?

The majority agrees that they learn the impacts of the environment

04

Limitations & Future Work

🚧 Limitations

- ⚙️ Static Systems
  - Static Simulation Parameters:
    - ➜ Users cannot modify animal attributes or environmental parameters during runtime, reducing hands-on experimentation potential.
  - Lack of Long-Term State Saving:
    - ➜ No support for persistent sessions makes ecosystem change analysis difficult across multiple runs.
- 🎮 Gameplay & AI
  - Limited Species and AI Complexity:
    - ➜ The simulation supports only a small set of prey and predator species using rule-based behaviors. Logic is controlled via finite state machines, which limits adaptability.
  - Lack of Full Multiplayer Functionalities:
    - ➜ Multiplayer features are incomplete, reducing collaborative exploration or competitive gameplay potential.

🚗 Future Features

- 🐾 Animal Systems
  - Support for More Animals and Environments
  - Animal Evolution System
  - Advanced Animal Intelligence
  - Animal Shelter-Seeking Behavior
- 🌍 World & Gameplay
  - In-Game Modification of Attributes and Terrain
  - Procedural World Generation
  - Multiplayer Exploration and Collaboration
  - Weather Seasonal Cycles
- 🧩 Player Interaction
  - Multiplayer Exploration and Collaboration
  - Scenario-Based Challenges and Missions
- 📊 Data & Integration
  - Integration with Real Ecological Data
  - Data Export Tools

05

Challenges & Solutions

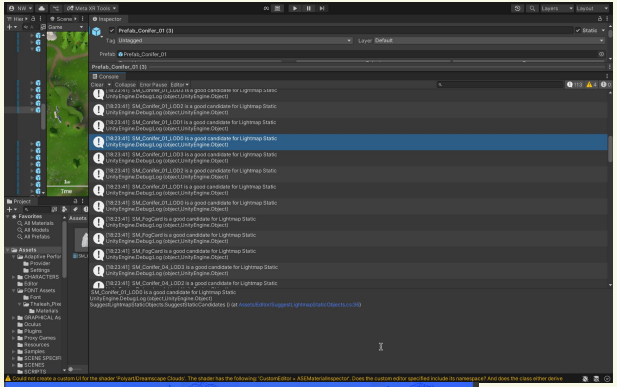
Challenges & Solutions

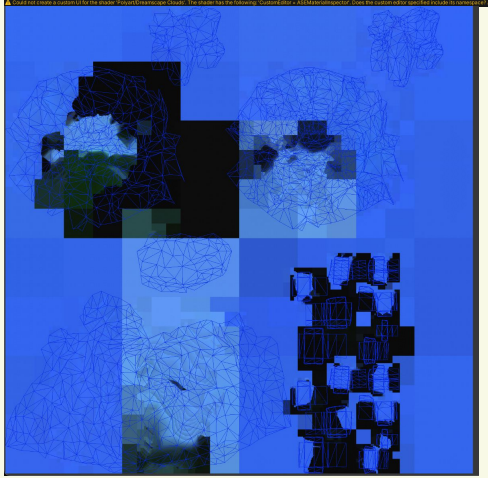
High GPU Utilisation

Excessive draw calls and overdraw from complex environments, dynamic lighting, and weather effects.

Optimisation for VR

Mixture of VR Optimisation measures such as baked lighting, modifying rendering settings and GPU instancing were leveraged





Challenges & Solutions

Lack of 3D Art Support

Our team had no experience in 3D asset modelling

Unity Asset Store

We relied on existing assets from the Unity Asset Store.

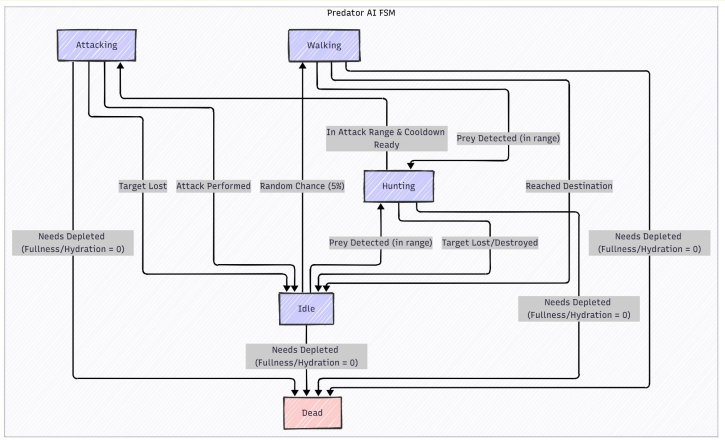
Challenges & Solutions

Balancing Realistic Animal AI with Performance Optimisation

Implementing animal behaviours individually across numerous animals at runtime introduced a performance strain

Modularised Animal AI

We modularised the AI into lightweight state machines and introduced time-staggered logic updates.



Member	Task
Noel	**ASSETS:** • Game Scene Environment Integration • Animal Asset Integration and scaling • Animal Behaviour animation controllers and scripts • Tenkoku Sky Asset Integration • VR Integration (XR Rigs, Teleportation, Button Mapping etc.) **LOGIC:** • PredatorAI, PreyAI and Predator-Prey dynamics (Animal Intelligence) • Hunger and Hydration System • Weather Events and effects on ecosystem **UI & GAMEPLAY:** • Main Menu User Interface • Game Scene User Interfaces • Time Lapse and Camera Zoom • Hunger and Hydration interfaces • Bird-Eye Mode and First-Person Mode • Tutorial System **OTHERS:** • VR Optimisation • Testing & Bug Fixes • Report + Slides

Member	Task

Jing Hua	**ASSETS:**
	• Analytics Scene Environment Integration
	• Binoculars & Magnifying Glasses Asset Integration
	• Food & Water Asset Integration
	
	LOGIC:
	• Conway Game of Life (Natural Birth and Death)
	• Binoculars & Magnifying Glasses System
	• Feeding System
	• Animal Data Tracking
	
	UI & GAMEPLAY
	• Detection Range Visualisation
	• Binoculars & Magnifying Glasses
	• Food and Water Interaction
	• Ecosystem Data Analytics
	  ○ Population Snapshot – Pie Chart
	  ○ Population Over Time – Line & Area Chart
	  ○ Animal Movement – Scatter Heatmap
	
	OTHERS:
	• Managed Git Version Control
	• Testing & Bug Fixes
	• Report + Slides

Member	Task

Miko	**ASSETS:**
	• Main Menu Scene Environment Integration
	
	LOGIC:
	• Basic Predator-Prey Dynamics (Animal Intelligence)
	
	UI & GAMEPLAY:
	• Main Menu Scene Improvement
	• Ecosystem Data Analytics
	  ○ Population Forecast – ARIMA Prediction
	  ○ Animal Attribute Comparison – Radar Chart
	OTHERS:
	• Report + Slides
	• Survey
	• Video Trailer – Film + Create + Edit

THANKS

Do you have any questions?

Team 5
Noel Wee Ee Cher
Toh Jing Hua
Leong Mininn Miko

06

Demo