



---










**TEAM PROJECT**

Word Hunter Game: Educational Adventure Game

---

# TEAM ROLES



-  **Team Leader:**  George Bolias
  -  **Programmer:**  Thomas Kastanaras
  -  **Level Designer:**  Aristotelis Davetas - Michopoulos
  -   **UI/UX & Audio Designer:**  George Bolias
-

---

# OVERVIEW – GAME IDEA



Word Hunter is a 3D educational scavenger hunt game designed and developed in Unreal Engine 5 by our team. At its heart, the game merges gameplay with learning, specifically targeting vocabulary development for young children. Inspired by traditional scavenger hunts, it offers a fully interactive environment where players are encouraged to explore, discover, and learn.



Set in vibrant and engaging levels such as forests, beaches, and cities, players take control of a fox character who must find items starting with each letter of the alphabet. The goal is to collect 26 unique objects in each level (one for every letter) while being rewarded with trophies and visual recognition.

The game's artistic direction focuses on friendliness and color to ensure the target audience, children in early education, feel welcomed and excited to play. In addition to collecting, the game encourages observation, memory, and pattern recognition, all of which are vital skills in a child's cognitive development



---

# TARGET AUDIENCE - BENEFITS



Word Hunter is designed specifically for children aged 5 to 10 years, particularly those in preschool and early primary education. The educational aspect is built to support vocabulary recognition, phonemic awareness, and association between visual objects and their corresponding letters.



In addition to being suitable for at-home play, the game is ideal for classroom settings where teachers can use it as a supplementary learning tool. Its design supports both solo play and guided learning, allowing parents or educators to join the experience and assist children in finding items, discussing object names, and reinforcing lessons outside traditional methods.



---

# EDUCATIONAL VALUE



The educational foundation of Word Hunter is deeply integrated into its mechanics. By requiring players to collect items starting with each letter, the game reinforces both letter recognition and object-word association. These skills are critical in early childhood literacy development.



The game also supports:

- Repetition-based learning: Reinforces memory by revisiting collected objects.
- Environmental learning: Encourages context-based recognition (e.g., finding a "Shell" on the Beach level).
- Auditory reinforcement: Sounds associated with correct collection help solidify learning.
- Reward-based motivation: Trophies and celebratory animations build intrinsic motivation to complete each level.

Teachers and parents can use the game as a vocabulary enrichment activity, prompting discussions about new words and encouraging kids to use those words in sentences after playing.



---

# TECHNICAL ASPECTS



Word Hunter is built using Unreal Engine 5, combining the power of Blueprint visual scripting with C++ programming to manage core logic, UI interaction, and performance.



Key technical features include:

- A collectible tracking system that monitors the player's progress across sessions.
- Interactive UI widgets that show real-time updates on collected items.
- A dynamic main menu system and level selection interface via a custom tablet mesh.
- Sound integration for both action feedback (e.g., collecting an item) and ambient enhancement (e.g., birds in the forest level).
- Optimized asset management and a packaged executable version for easy distribution and use in non-development settings



---

# INSTALLATION INSTRUCTIONS

## MINIMUM REQUIREMENTS



To run Word Hunter, the following minimum system specifications are recommended:

- Operating System: Windows 10 or later
- Processor: Intel Core i5 or AMD Ryzen 5 (or equivalent)
- RAM: 8 GB minimum
- Graphics: NVIDIA GTX 1050 or higher with DirectX 11 support
- Storage: At least 2 GB of available space



---

# INSTALLATION INSTRUCTIONS

## (PACKAGED VERSION)



For most users, especially educators and parents:

1. Download the zipped game package.
2. Extract the folder to a location of your choice.
3. Locate and double-click the WordHunter.exe file.
4. The game will launch in full-screen. Use the keyboard and mouse to interact.



---

# INSTALLATION INSTRUCTIONS

## (SOURCE VERSION)



For developers or students wanting to explore or expand the game:

1. Download and install Unreal Engine 5 (latest version preferred).
2. Extract the source files.
3. Open the .uproject file inside the Unreal Editor.
4. Compile and build the game using the development tools.
5. Press Play to enter the game world and begin testing or editing



---

# USER MANUAL



## Main Menu

- Start the game
- Adjust settings
- View credits
- Exit the game

## Controls

- W, A, S, D – Move the character (fox)
- Mouse – Look around the environment
- Left Click – Interact with objects (e.g., collect items)
- Escape – Open the pause menu to adjust settings or exit

## Main Menu

1. Start the game and enter the main room.
2. Select a level from the tablet.
3. Explore the level to find 26 items (A–Z).
4. Collect each item to unlock trophies.
5. Complete all items in a level to finish and return to the hub.



---

# ENVIRONMENTS



There are three beautifully designed levels, each offering unique themes and associated objects:

- City: Urban elements like buses, benches, balloons.
- Beach: Seashells, umbrellas, kites.
- Forest: Trees, animals, fruit.



These environments were built with optimized assets and intentional object placement to provide discovery and learning.



---

# UI ELEMENTS



The game includes several interactive interface components:

- HUD widget: Tracks collected items with icons or letters.
- Pause menu: Accessed via Escape key; includes sound control and quit option.
- Tablet menu: Used to navigate levels.
- Trophy wall: Displays achievements visually in the hub world.



---

# TEAM CONTRIBUTION



- George Boliias – Team Leader, UI/UX Design, Audio: George oversaw project management, user experience design, and audio implementation. He developed intuitive UI systems and ensured the game was accessible for its young audience.
- Thomas Kastanaras – Programmer: Thomas was responsible for core gameplay mechanics, including character movement, item interaction, and logic systems. His work in Blueprints and C++ ensured the game was stable and functional.
- Aristotelis Davetas-Michopoulos – Level Designer: Aristotelis created the 3D environments, placed collectible assets, and designed navigation flow to maintain engagement and clarity.

Weekly sync-ups ensured that all features were integrated cohesively, with frequent playtests guiding adjustments in difficulty, design, and visual clarity.



---

# DEVELOPMENT PROCESS SUMMARY



The game followed a structured development approach divided into five phases:

- **Phase 1: Planning**

Brainstormed themes, defined gameplay mechanics, and divided roles. A shared task board kept progress transparent.

- **Phase 2: Core Development**

Developed movement controls, collectible systems, and began basic level layout.

- **Phase 3: Polish**

Focused on improving UI, integrating audio feedback, and adding environmental props for immersion.

- **Phase 4: Playtesting**

Collected player feedback and fixed bugs. Improved clarity of instructions and refined the win condition system.

- **Phase 5: Packaging & Submission**

Packaged the game for distribution. Prepared presentation materials, documentation, and gameplay trailer (if applicable).

---





---

# TEAM UN-REALS



**Thank you for your time.**

**Questions?**

 George Bolias  Thomas Kastanaras  Aristotelis Davetas - Michopoulos

---