

A PRACTICAL GUIDE

Build Your First AI Resident: A Practical Guide

Create a functional autonomous character in a 3D world with minimal code and resources

Made by the Braxis machine

braxisai.com

What this is

This guide is for creators and developers who want a small autonomous character in a 3D world.

It is a practical method, not a promise of a result. It contains no invented statistics, no fake customers, and no guarantees. The words were drafted with AI assistance.

Use the checklist at the end on a real piece of work this week.

Define the Purpose and Scope

Start by clarifying what your AI resident will do. Will it be a simple NPC that follows a path, interacts with objects, or responds to player actions? Limit its scope to one or two core behaviors to avoid complexity. For example, if you're building a shopkeeper, decide if it needs to sell items, respond to questions, or just stand in place. Write down these behaviors in plain terms—this will guide your design. Use free tools like a text editor or a simple whiteboard to outline the resident's role. Avoid over-engineering; focus on a single, clear function. This step ensures you build only what's necessary.

Choose the Right Tools and Engine

Select a lightweight 3D engine that supports AI behavior. Unity or Unreal Engine are popular, but for beginners, consider Godot or Blender with its built-in logic editor. If you're new to 3D, Blender is free and beginner-friendly. For AI logic, use a simple state machine or behavior tree. Unity's NavMesh system is great for pathfinding, while Godot's built-in AI tools are lightweight. Avoid overcomplicating the tech stack—stick to one engine and its native tools. Download the engine and install the necessary plugins or extensions for AI behavior.

Design the Resident's Movement and Behavior

Create a basic 3D model for your resident using free assets or simple shapes. For movement, set up a path or navigation mesh in your engine. In Unity, use the NavMesh Agent component; in Godot, use the NavigationServer. For behavior, define simple rules like 'move to point A when player is near' or 'rotate toward the player.' Use scripts to implement these rules. Start with hardcoded behaviors—like a resident that walks in a loop—and gradually add conditions. Test movement in isolation first to ensure it works before adding interactions.

Implement Basic Interactions and Testing

Add simple interactions like collision detection or trigger zones. For example, if your resident is a guard, set up a trigger that activates when the player enters a certain area. Use the engine's built-in physics or collision systems. Write a script to handle these interactions, such as playing an animation or displaying text. Test these interactions in the engine's editor. Use the debug tools to check for errors and refine the behavior. Start with basic interactions and expand as needed. Avoid complex logic until the fundamentals work.

Deploy and Iterate

Once your resident works in the editor, export it to a standalone build or integrate it into your project. Test it in the target environment to ensure compatibility. Gather feedback from others if possible, and refine the behavior based on their input. Document any issues or improvements needed. Iterate by tweaking movement, behavior, or interactions. Keep the project simple—focus on one improvement at a time. Share your progress in online communities for additional insights. This step ensures your resident is polished and functional in its intended environment.

Checklist

- Define the resident's purpose and limit its scope to 1-2 core behaviors.
- Choose a lightweight 3D engine (e.g., Godot, Blender) and install necessary AI tools.
- Create a simple 3D model and set up basic movement using pathfinding or navigation tools.
- Implement basic interactions (e.g., collision, triggers) and test in the engine's editor.
- Deploy the resident, gather feedback, and iterate on behavior or functionality.

Where this came from

This guide is a digital product from Braxis AI. The machine wrote it with AI assistance. It contains no invented statistics, no fake customers, and no guarantees. The PDF is delivered immediately after checkout. If something is wrong, email braxisai@gmail.com and it will be fixed.

braxisai.com