

Flawless Abbey: Test Task README

Author: Nala Mthembu

Engine: Unreal Engine 5

Design Decisions

Input & Systems Architecture

- Created Input Actions for Interact and Inventory toggling to support modular, responsive controls.
- Made use of Actor Components for clean separation of functionality:
 - BP_InteractionComponent: Handles detecting and interacting with objects in the world.
 - BP_InventoryComponent: Manages inventory state and item storage.

Data-Driven Approach

- Used Data Assets for inventory items (BP_InventoryItem) to support modular item definitions.
- Dialogue is structured through a Data Table (DT_Conversations) and Struct (ST_Conversation), enabling easy expansion and localization.

Interaction Logic

- Created a reusable Blueprint Interface (BPI_Interactable) for interactable objects.
- Developed a Pickup Base Class (BP_Pickup) inheriting the interface to enable unified interaction logic.
- Used Sphere Trace over Line Trace to allow for more forgiving interaction detection (larger interaction volume).
- Introduced a variable to store the current interaction context, cleared if the player looks away.

Event-Driven Communication

- Avoided polling UI by broadcasting events:
 - Example: OnPlayerInteract, OnInventoryOpen, etc.
- This approach keeps UI reactive and performant, without unnecessary tick logic.

UI Design & Functionality

Widget Architecture

- WBP_HUD: Root widget controlling visibility and listening for events.
- WBP_InteractionUI: Displays context-sensitive interaction prompts.
- WBP_Dialogue: Handles conversation flow and UI logic.

- W_Inventory, W_InventorySlot, W_DraggedItem: Handle inventory visuals and drag/drop operations.
- W_ItemViewport: Hosts the interactive 3D item view.

Drag and Drop Inventory

- Dragging slots creates a preview widget (W_DraggedItem) and uses a custom Drag & Drop Operation to transfer data.
- Swapping logic is handled via events that notify the inventory component to update the data model.

Item Inspection

- A dedicated BP_InventoryInspectionActor renders a mesh to a Render Target, shown in UI.
- It dynamically centers the object regardless of pivot using bounding box calculations.
- Interaction:
 - - Left Mouse Hold + Drag: Rotate object.
 - - Right Mouse Hold + Drag Up/Down: Zoom.

Dialogue Camera System

- During conversations, the camera smoothly rotates toward the character speaking.
- Once dialogue ends, control and view return to the player character using a cached camera state.

How to Test the Features

Interaction System

- 1. Press E near an interactable.
- 2. If it's a pickup, it's added to the inventory.
- 3. If it's a character, dialogue starts and camera adjusts.

Dialogue System

- 1. Click "Next" to proceed.
- 2. Final line will change button to "Close."
- 3. Camera returns to player after close.

Inventory System

- 1. Press I to open.
- 2. Drag items between slots.
- 3. Drag an item to the viewport slot:

- - 3D mesh appears and can be rotated/zoomed.
- 4. Press ESC or click close to exit inventory.