

## Assignment 03 (Individual) – GAMES DEVELOPMENT

SE4031 – Games Development

Title – Immersive VR Exploration Experience: THE LOST SANCTUARY

Assignment Created By: Mr. Aruna Ishara Gamage, Mr. Nushkan Nismi

**End Digit 6**

Assignment Weight: **30%**

## OBJECTIVE

### *The Lost Sanctuary*

**Design and develop an immersive Virtual Reality (VR) exploration experience application using Unity, where players uncover a hidden ancient sanctuary.**

Players must navigate sacred ruins, interact with ancient artifacts, invoke rituals using voice commands, and complete a ritual-based environmental trial to progress.

## ASSIGNMENT STRUCTURE (2 Parts)

### • **Part A (60%) – Based on Tutorials + Lab Sheets**

Students can score Part A using concepts and techniques covered in tutorials and lab sessions.

### • **Part B (40%) – Self-Learning + Advanced Work (Required for High Grades)**

Part B requires independent research and creativity, focusing on voice-based ritual invocation and a unique sanctuary challenge.

## CORE REQUIREMENTS

### Part A Requirements (60%)

#### 1. VR Movement & Locomotion

- Use teleportation or joystick-based movement.
- Allow comfortable exploration across multiple sanctuary areas (halls, chambers, ruins)

#### 2. Environmental Interactable

- Implement at least **three (3)** of the following:
  - Ancient artifacts or relics: Pick up or examine items
  - Rotating mechanisms or pressure plates: Trigger environmental changes
  - Lore objects (tablets, murals, scrolls): Reveal story or clues
  - Doors or hidden passages: Unlock or access new areas

#### 3. VR UI Elements

- Include a HUD or VR-friendly UI displaying:
  - Current ritual state
  - Player objectives or guidance
  - Ritual energy / progress meter

#### 4. Basic Environmental Feedback

- Ambient sounds (chants, wind, echoes)
- Subtle lighting changes or symbol activation
- Feedback triggered by player movement or interaction

## Part B Requirements (40%) – Self Learning

### 5. Ritual Invocation System (Voice Input Only)

- The player must activate at least three (3) rituals using voice commands only (**no gestures or button presses**).

**Example Commands (DO NOT USE):**

- “Awaken” – Activates dormant structures
- “Illuminate” – Reveals hidden symbols or paths
- “Bind” – Locks or stabilizes moving elements

#### **⚠️ IMPORTANT NOTE**

- The ritual names listed below are **EXAMPLES ONLY**
- Students **MUST** create their **own original ritual names**.
- Using the example words exactly as written will result in **0 marks** for this component.

**The system must:**

- Accurately detect commands using voice input tools (e.g., Windows Speech API (wit.ai)).
- Provide visual/audio feedback upon successful commands recognition.
- Show a response or warning for unrecognized commands.
- The Windows Speech API is supported only on the Windows desktop platform.
- When the application is built and executed directly on a VR headset, the Windows Speech API will not function. However, the same voice command system will work correctly when the application is built as a Windows Desktop (.exe) and run on a PC with a VR headset connected

### 6. Voice Command Integration

- **Integrate a real-time** speech recognition system.
- Provide **visual and/or audio** feedback for:
  - **Successful** ritual invocation
  - **Incorrect** or **unrecognized** ritual words
- Voice system must remain responsive during gameplay

### 7. Environmental Reaction System

- Rituals must trigger clear environmental reactions, such as:
  - **Lighting shifts** or **glowing symbols**
  - **Moving structures** or **opening passages**
  - **Soundscape changes** or **environmental motion**

### 8. Sanctuary Trial Area (Mandatory)

- Include a **dedicated** sanctuary **trial zone** where the player must:
  - Use **voice-invoked** rituals to solve a challenge
  - Combine **artifact interaction**
  - Successfully complete a ritual-based puzzle, escape, or progression trial

#### **Part B Creative Challenge Note**

**Part B** includes a creative challenge, and it must be unique (**not copied from other students**) to score marks.

## **Folder Structure & Code**

- Unity project must follow the folder structure.
- Scripts and assets must be well-named and organized.

## **Game Documentation**

- Submit a PDF with:
  - Title, student name, and IT number
  - Game summary and spell list
  - Screenshots of gameplay
  - Control guide (movement, voice usage)
  - Credits for any assets/tools used

## **Submission Requirements**

- **Windows .exe Build**  
Include .exe and Data folder, playable with a VR headset.
- **Zipped Unity Project Folder**  
Must follow the folder structure.  
Upload to a shared Google Drive folder in Courseweb.
- **Gameplay Demo Video**
  - 5 minutes showcasing:
    - Voice-activated spellcasting
    - Movement
    - Item collection and target interaction
    - Spellbook interaction
    - Challenge zone gameplay

## **PLAGIARISM / ORIGINALITY VERIFICATION (VIVA)**

- A mandatory one-to-one viva will be conducted
  - Students must clearly explain:
    - Voice command logic
    - System interaction behavior
    - Emergency challenge design
- Failure to justify originality may result in mark deductions or zero marks

### Assessment Rubric (Part A + Part B)

#### Part A (60 Marks)

Criteria	Excellent	Good	Satisfactory	Poor	Marks
<b>VR Movement &amp; Locomotion</b>	Smooth, immersive, and comfortable navigation implemented across multiple areas	Navigation implemented with minor comfort or control issues	Basic navigation implemented with limitations	No or broken navigation	<b>12</b>
<b>Environmental Interactables</b>	All required interactables are fully implemented with clear visual/audio feedback and logical consequences	Most interactables working with minor issues	Limited interactables or inconsistent behavior	No meaningful interactables	<b>12</b>
<b>VR UI Elements</b>	Clear, minimal, VR-friendly UI supporting objectives and survival awareness	UI is mostly clear with minor usability issues	Basic UI with limited usefulness	No VR UI elements	<b>10</b>
<b>Folder Structure &amp; Code Quality</b>	Fully structured Unity project with clean, well-named scripts and assets	Mostly structured with few misplacements	Some structures are present but inconsistent	Disorganized or missing structure	<b>8</b>
<b>Game Documentation</b>	Complete and clear PDF covering all required sections	Mostly complete with minor missing details	Basic documentation with limited explanation	No documentation submitted	<b>8</b>
<b>Part A Total</b>					<b>60</b>

### **Part B (40 Marks) – Self Learning**

Criteria	Excellent (Full Marks)	Good	Satisfactory	Poor	Marks
<b>Voice-Based Psychological Commands (Voice Only)</b>	Three (3) unique psychological survival commands accurately triggered via voice with clear visual/audio feedback	Two commands working, one partially functional	Only one unstable or partially working command	Commands missing or not voice-controlled	<b>20</b>
<b>Voice Command Integration</b>	Voice system is stable, responsive, and displays clear captions and feedback	Minor delays or recognition issues	Frequently unresponsive or inconsistent	Not implemented or unusable	<b>12</b>
<b>Environmental Reaction + Survival Challenge Area</b>	Strong psychological immersion with dynamic infected-zone reactions and a fully playable survival challenge	Good immersion with a working challenge and minor issues	Basic reactions with a weak or non-threatening challenge	Poor or missing immersion and challenge	<b>8</b>
<b>Part B Total</b>					<b>40</b>

### **Plagiarism / Originality Verification (Viva)**