# 1. Project Proposal & Team Members Identification

### Title

**Labyrinth Navigator**

### Student Names

Name: Eyad Mostafa — ID: 320230028 — Major: AID — Section: 1

Name: Ahmed Abdelkader — ID: 320230035 — Major: AID — Section: 1

Name: Omar Momtaz — ID: 320230071 — Major: AID — Section: 2

### Game Description

Labyrinth Navigator is a simple, first-person exploration game set within a highly geometric, dimly lit 3D maze. The player's objective is to locate a hidden **Key Crystal** and then use it to unlock and reach the **Exit Portal** before a 60-second timer runs out. The design is deliberately minimalist to emphasize core graphics techniques (lighting, transformations, and projection) while ensuring high performance on integrated graphics hardware.

### Proposed Features (Mapping to Course Requirements)

| **Course Requirement** | **Feature Implementation** | **Graphics Concept Demonstrated** |
| --- | --- | --- |
| **A. 3D World** | Maze constructed from simple THREE.BoxGeometry walls, floor, and ceiling. | Translation, Scaling, Low-Poly Geometric Modeling |
| **B. Camera System** | **First-Person Camera** view, rigidly linked to the player's position. | Viewing and Projection Transformations |
| **C. Lighting** | **SpotLight** attached to the camera (Flashlight effect) that casts **Dynamic Shadows** throughout the maze. Minimal ambient light is used. | Diffuse Lighting, Shadow Mapping, Shading (Gouraud/Phong) |
| **D. Textures / Materials** | 1. Tiled **texture map** on maze walls and floor (e.g., concrete). 2. Key/Exit objects use an **emissive/shiny material** for visibility and specular highlights. | Texture Mapping, Specular Highlights, Material Properties |
| **E. User Interaction** | 1. **Keyboard Movement (WASD)**: Translates the player's position. 2. **Mouse Look**: Rotates the camera/player orientation. 3. **Activation (E Key)**: Interaction to collect the Key or activate the Exit. | User Input Handling, Transformation Updates (Translate/Rotate) |
| **F. Animation** | 1. Key Crystal uses continuous **Rotational** transformation. 2. Certain wall sections use smooth **Translational** oscillation as moving obstacles. | Real-time Transformations, Interpolation |
| **G. Game Mechanics** | **Collect N objects** (Key) and **Reach a Destination** (Exit Portal) within a **Time Limit**. Simple AABB collision detection is used for walls and collection points. | Game State Management (Win/Loss condition, Collision) |

### Tools/Technologies to be Used

* **Framework:** **Three.js (WebGL)**
* **Language:** JavaScript (ES6+)
* **Version Control:** GitHub