1. Programming Languages (5 points)

C#: The programming language we are using for our game project is C#. We use it in custom scripts to control game objects, such as handling character movement, camera behavior, and enemy AI. We chose C# because it is the only scripting language supported by Unity, and it integrates well with the engine's built-in tools. It also makes it easier to organize our code and manage game mechanics efficiently.

2. Platforms, APIs, Databases, and other technologies used (5 points)

Unity: We use Unity as our primary development platform to handle rendering, physics, animations, and scene management. It provides built-in tools that simplify game development and allows for quick iteration.

Visual Studio Code/VisualStudio Community: We use Visual Studio Code and Visual Studio Community as our code editors for writing and debugging C# scripts. It offers useful features like IntelliSense, extensions, and Git integration, making development more efficient.

Unity Asset Store: We use the Unity Asset Store to find and integrate premade and free assets to use in our game. This allows us to spend our time building and programming game mechanics rather than making art.

Unity's Physics Engine: Used to handle collisions, gravity, and movement mechanics, ensuring realistic interactions in the game.

Tilemaps (For Level Design): We use Unity's Tilemap system to efficiently create game environments with reusable assets.

Version Control (Git): We use Git to track changes in our project, allowing for collaboration and version history management.