Project Architecture

Submitted by:

Mogammel Hossen
Student ID: 210231

Sojun Chandra Shill
Student ID: 210215

Submitted to:
Dr. Amit Kumar Mondal
Associate Professor
Computer Science & Engineering Discipline,
Khulna University,
Khulna.

Course Name: Software Development Project

Course No.: CSE 3106

Project Title: Snake Game

Project Architecture: Layered Architecture

Applying the Layered Architecture Pattern to a Snake Game can help organize the different aspects of the game into separate layers, making the code more modular, maintainable, and extensible. Here's a high-level overview of a typical architecture:

1. **Presentation Layer:**
   a. Responsible for rendering the game graphics and handling user input
   b. Includes components like the game window, graphics rending, and user input handling.
   c. Interacts directly with the application Layer to convey user action and receive updates about the game.
2. **Application Layer:**
   a. Contains the core logic of the game, including the game mechanics and rules.
   b. Orchestrates the game flow, updating the game states based on user input and controlling game events.
   c. Communicate with the Presentation Layer to receive user input and provide updates on the game state.
3. **Persistence or Database Access Layer:**
   a. Acts as an abstraction layer between the application's business logic and the underlying data storage mechanisms.
   b. Contains components or modules responsible for data access, persistence, and retrieval.
   c. Allows for easier maintenance and scalability, as changes to the data storage technology can be localized within this layer
4. **Database Layer:**
   a. Represents the actual storage mechanism where the application's data resides, such as a relational database management system (RDBMS).
   b. Optimizes database performance through indexing, query optimization, and database tuning techniques.
   c. Provides mechanisms for querying and modifying data stored in the database.

A simplified diagram illustrates the Layered Architecture Pattern for a Snake Game.

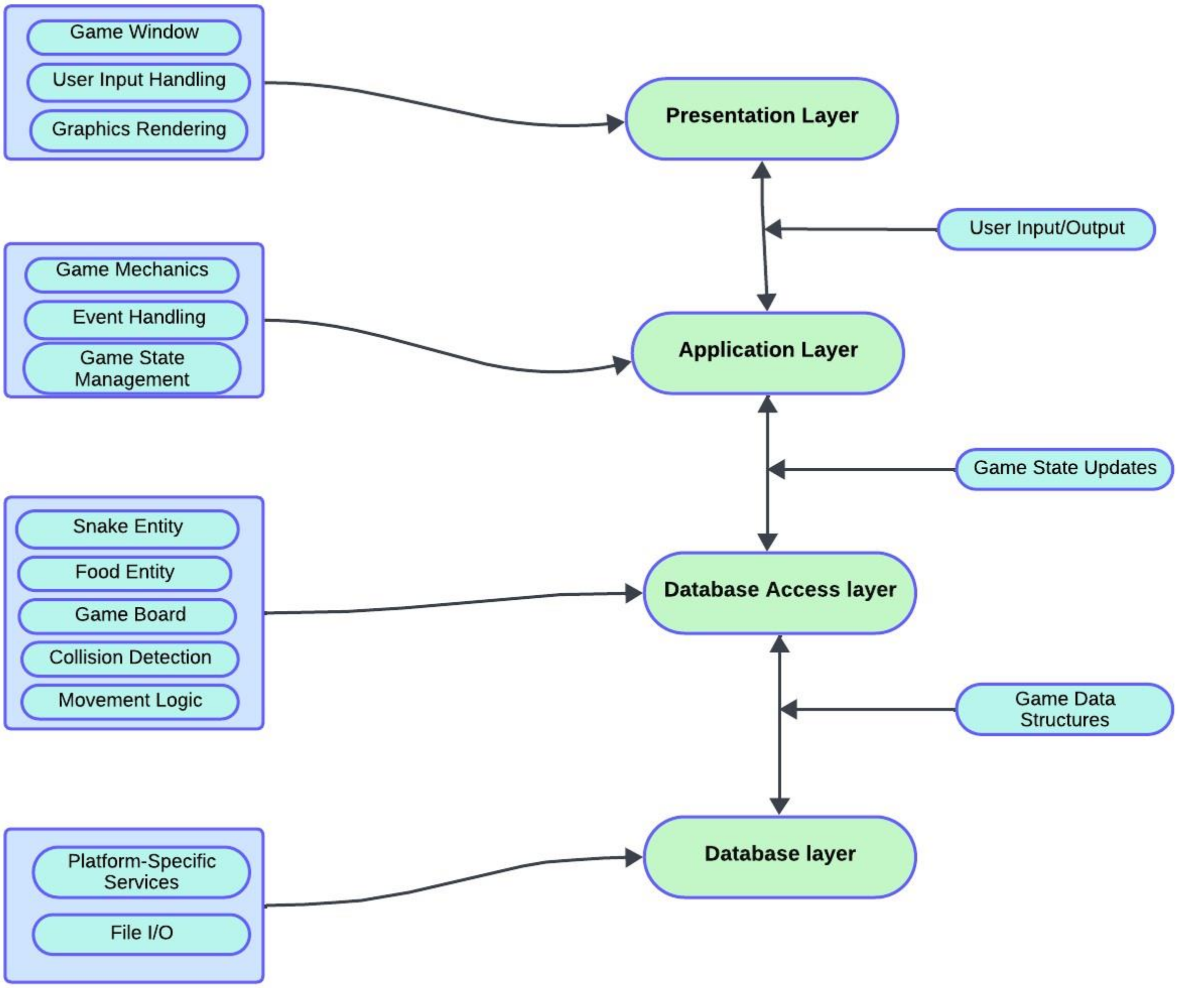

Figure: Layered Architecture Diagram

Summary:
Organizing the Snake Game into these layers makes it easier to maintain and extend the game in the future. For example, we could modify the game's graphics rendering without affecting the underlying game mechanics, or swap out the input handling system without altering the core gameplay logic.