UTT
UNIVERSIDAD TECNOLÓGICA DE TIJUANA
GOBIERNO DE BAJA CALIFORNIA

TEMA: Selection of Design Patterns

PRESENTADO POR: Ramírez Aispuro Juan José

GRUPO: 10-B

MATERIA: Desarrollo móvil integral

PROFESOR: Ray Brunett Parra Galaviz

Tijuana, Baja California, 01/07/2024

Recommended Design Pattern: **Singleton Pattern**

Why Choose Singleton Pattern?

1. Controlled Access to a Single Instance:

The Singleton pattern ensures that a class has only one instance throughout the application. This is ideal for scenarios where centralized control or a global point of access is required, such as managing configurations or database connections.

2. Efficient Resource Usage:

By limiting the instantiation of a class to one object, the Singleton pattern reduces memory usage and overhead, especially in resource-intensive operations.

3. Consistency Across Application:

Using a single instance ensures consistent behavior and state throughout the application. For example, logging services can use the Singleton pattern to maintain uniform logs across various modules.

4. Thread-Safety and Synchronization:

With proper implementation (e.g., using synchronized blocks or double-checked locking), Singleton ensures thread safety, making it suitable for multi-threaded environments.

This pattern is particularly useful in cases where multiple objects could lead to redundant operations or inconsistent states, such as in managing configuration settings, caching, or connection pooling.