UTT
UNIVERSIDAD TECNOLÓGICA DE TIJUANA
GOBIERNO DE BAJA CALIFORNIA

TEMA:

Design Patterns

PRESENTADO POR:

Hernández Miranda Rafael Francisco

GRUPO:

10B

MATERIA:

Desarrollo Móvil Integral

PROFESOR:

Ray Brunett Parra Galaviz

FECHA:

06/01/2025.

Design Patterns

A design pattern is a reusable solution to a commonly occurring problem in software design. It serves as a template or blueprint that can be customized to address specific design challenges within various contexts. The concept was initially introduced by architect Christopher Alexander and has since been adapted for software engineering and other fields. Design patterns help developers communicate effectively, improve code readability, and streamline the development process by providing tested solutions that can be applied across different projects

Design patterns are generally repeatable solutions to common problems in software design. They help in creating maintainable, flexible, and efficient code by providing standard approaches to recurring design issues. The three primary categories of design patterns are:

- **Creational Patterns**: Focus on object creation mechanisms.
- **Structural Patterns**: Concerned with how objects are composed to form larger structures.
- **Behavioral Patterns**: Deal with object interaction and responsibility delegation.

Steps for Selecting a Design Pattern

1. **Identify the Core Problem**: Analyze the specific issues you are facing, such as complex object creation or managing dependencies. Understand whether your needs are primarily creational, structural, or behavioral.
2. **Define Requirements and Constraints**: Consider the flexibility, scalability, and performance needs of your solution. Determine if there are specific conditions that your design must satisfy.
3. **Explore Design Principles**: Familiarize yourself with principles like SOLID, DRY, KISS, and YAGNI. These principles can guide you in making better design decisions and avoiding common pitfalls.

4. **Review Available Patterns**: Look through existing design patterns that match your identified problems. Each pattern comes with a description, motivation, structure, and implementation guidelines that can help you compare options.

.

5. **Evaluate Trade-offs**: Assess the advantages and disadvantages of different patterns based on criteria such as readability, testability, maintainability, and compatibility with your programming environment.
6. **Choose and Implement**: Select the most suitable pattern(s) for your situation. You may also adapt or combine patterns to fit your unique requirements while documenting your choices for future reference.

Conclusion

Design patterns are invaluable tools in software development, offering proven solutions to common design challenges. By providing a standardized vocabulary and framework for addressing recurring issues, design patterns enhance communication among developers and improve code maintainability. The key to effectively utilizing design patterns lies in understanding the specific problem at hand, selecting the appropriate pattern, and adapting it to fit the unique context of the project. Ultimately, leveraging design patterns not only streamlines the development process but also fosters better software architecture, leading to more efficient and scalable applications.

References

Ponde, P., Bharambe, M., Khobragade, K., & Suryawanshi, M. (2023). Knowledge discovery for design pattern selection. Atlantis Press. Retrieved from https://www.atlantis-press.com/article/125986322.pdf

Runtime Revolution. (2024). Design patterns and sources. Retrieved from https://revs.runtime-revolution.com/design-patterns-and-sources-b602790dcebc?gi=75d8817d1b9e

GeeksforGeeks. (2024). Design patterns cheat sheet - When to use which design pattern? Retrieved from https://www.geeksforgeeks.org/design-patterns-cheat-sheet-when-to-use-which-design-pattern/

Refactoring.Guru. (n.d.). Design patterns. Retrieved from https://refactoring.guru/design-patterns

SourceMaking. (1999). Design patterns. Retrieved from https://sourcemaking.com/design_patterns