**Assignment Questions**

**Unit 1: Layered Technologies, Waterfall model, Spira, Agile**

1. Explain the concept of **Layered Technologies** and discuss how it applies to modern software development.
2. Compare and contrast the **Waterfall model** with other software development methodologies. What are the key advantages and disadvantages of the Waterfall model?
3. What is **Spira**, and how does it support project management and software development in an agile environment?
4. Discuss the **Agile methodology**. What are the core principles of Agile, and how do they contribute to the success of software development projects?

**Unit 2: W5HH Principle, Risk Management, Effort Estimation**

1. Explain the **W5HH principle** in software engineering. How does it guide the requirements gathering and system analysis process?
2. Discuss the concept of **Risk Management** in software development. What are the different types of risks that can affect a project, and how can they be mitigated?
3. Describe the different methods of **Effort Estimation** in software engineering. How can effort estimation help in planning and scheduling a software project?

**Unit 3: Requirements Engineering, Use Cases and Functional Specification**

1. What is **Requirements Engineering**? Describe its role in the software development lifecycle and the processes involved in gathering and documenting requirements.
2. Explain the significance of **Use Cases** in software development. How do use cases contribute to functional specification?
3. How do **Functional Specifications** help in transforming requirements into a concrete software design? Provide an example of how they are structured.

**Unit 4: Structured System Design, Software Architecture, E-R Diagram, Dataflow Model, Control Flow Model, Data Dictionary, Object-Oriented Design**

1. Discuss the concept of **Structured System Design** and its importance in creating efficient software systems.
2. What is **Software Architecture**, and why is it crucial in the design phase of software development?
3. Define **E-R Diagram** (Entity-Relationship Diagram) and explain its role in system design. Provide an example of its use.
4. Describe the **Dataflow Model**. How does it help in representing system processes and data exchanges?
5. What is a **Control Flow Model**, and how does it differ from a Dataflow Model in terms of system design representation?
6. Explain the purpose and structure of a **Data Dictionary** in software design. How does it contribute to a better understanding of the system's data?
7. What are the key principles of **Object-Oriented Design** (OOD)? How does OOD improve modularity and reusability in software development?