Summary of Chapter 5: System Modeling

1 Topics Covered

- Context models
- Interaction models
- Structural models
- Behavioral models
- Model-driven engineering

2 System Modeling

- Process of developing abstract models of a system to present different views or perspectives.
- Often uses graphical notation based on UML (Unified Modeling Language).
- Helps analysts understand system functionality and communicate with customers.

3 Existing and Planned System Models

- Models of existing systems clarify current functionalities and identify strengths and weaknesses.
- Models of new systems explain proposed requirements and document the system for implementation.
- Model-driven engineering can generate complete or partial system implementations from models.

4 System Perspectives

- **External perspective**: Models the context or environment of the system.
- **Interaction perspective**: Models interactions between the system and its environment or between components.
- **Structural perspective**: Models the organization of the system or data structure.
- **Behavioral perspective**: Models the dynamic behavior of the system in response to events.

5 Use of Graphical Models

- Facilitate discussion about existing or proposed systems.
- Document existing systems accurately, even if incomplete.
- Provide a detailed system description for implementation, requiring correctness and completeness.

6 UML Diagram Types

- **Activity diagrams**: Show activities involved in processes or data processing.
- **Use case diagrams**: Show interactions between the system and its environment.
- **Sequence diagrams**: Show interactions between actors and the system or system components.
- **Class diagrams**: Show object classes in the system and associations between them.
- **State diagrams**: Show system reactions to internal and external events.

7 Context Models

- Illustrate the operational context of a system, showing what lies outside its boundaries.
- Define system boundaries to establish what is inside and outside the system.

8 Interaction Models

- Important for identifying user requirements and understanding system communication problems.
- Use case diagrams and sequence diagrams are used for modeling interactions.

9 Use Case Modeling

- Developed to support requirements elicitation and incorporated into UML.
- Represents discrete tasks involving external interaction with a system.
- Use cases include actors (people or other systems) and are represented diagrammatically and textually.

10 Sequence Diagrams

- Part of UML, used to model interactions between actors and system objects.
- Show the sequence of interactions during a particular use case.

11 Structural Models

- Display the organization of a system in terms of its components and their relationships.
- Can be static (showing design structure) or dynamic (showing structure during execution).

12 Class Diagrams

- Used in object-oriented system modeling to show classes and their associations.
- Generalization manages complexity by categorizing entities into classes with common characteristics.

13 Behavioral Models

- Model the dynamic behavior of a system in response to stimuli.
- **Data-driven models**: Show sequence of actions in processing input data.

- **Event-driven models**: Show how a system responds to external and internal events.
- State machine models show system states as nodes and events as arcs between them.

14 Key Points

- Models provide abstract views of a system, ignoring details.
- Context models position a system within an environment.
- Use case and sequence diagrams describe interactions.
- Structural models show system organization and architecture.
- Behavioral models describe system dynamics.
- Model-driven engineering uses models to generate executable code.

15 Scenarios

15.1 Scenario 1: Hospital Management System

Context Model: Shows the hospital management system interacting with patient records and billing systems.
Use Case Diagram: Interactions between doctors, nurses, patients, and the system for scheduling appointments, updating patient records, and billing.
Sequence Diagram: Steps involved when a doctor updates a patient's record.
Class Diagram: Defines classes like Doctor, Patient, Appointment, and Bill.
State Diagram: System states when a patient is admitted, treated, and discharged.

15.2 Scenario 2: Online Shopping System

Context Model: Illustrates the online shopping system interacting with payment gateways and inventory systems.
Use Case Diagram: Shows customer interactions for browsing products, adding to cart, and making payments.
Sequence Diagram: Sequence of actions when a customer makes a purchase.
Class Diagram: Defines classes like Customer, Product, Order, and Payment.
State Diagram: System states when an order is placed, processed, and shipped.

16 Exam Questions

1. **Explain the different perspectives used in system modeling.**
   - **Answer**: The perspectives include external, interaction, structural, and behavioral perspectives, each offering a different view of the system.
2. **Describe the purpose of use case diagrams and provide an example.**
   - **Answer**: Use case diagrams show interactions between a system and its environment. Example: In a hospital management system, a use case could be a doctor updating a patient's record.
3. **What are structural models and why are they important?**
   - **Answer**: Structural models display the organization of a system, showing components and their relationships. They are important for understanding system architecture and design.
4. **How do behavioral models differ from structural models?**
   - **Answer**: Behavioral models describe the dynamic behavior of a system in response to stimuli, whereas structural models show the static organization and architecture of the system.
5. **Explain the concept of model-driven engineering.**
   - **Answer**: Model-driven engineering is an approach where a system is represented by a set of models that can be automatically transformed into executable code.