Software Engineering Exam Notes

SET 1

A. Introduction to GitHub and Version Control System

What is GitHub?

GitHub is a web-based platform for version control that allows developers to manage, track, and collaborate on software projects using Git. It provides repositories for storing code and enables multiple developers to work together efficiently.

Features of GitHub:

- **Repositories:** Store project files and track changes.
- **Branching and Merging:** Develop features independently and merge them when ready.
- **Pull Requests:** Allow for code review before merging.
- **Issue Tracking:** Manage bugs and feature requests.
- **CI/CD Support:** Automate testing and deployment.

Version Control System (VCS)

A **VCS** is a tool that helps manage code changes over time. It ensures that developers can track, compare, and revert to previous versions.

Types of VCS:

1. **Local VCS**: Stores versions in a local machine (e.g., RCS).
2. **Centralized VCS (CVCS)**: Uses a central server for tracking (e.g., SVN).
3. **Distributed VCS (DVCS)**: Each developer has a full copy of the project (e.g., Git).

B. Software Architectural Diagrams

Types of Software Architecture Diagrams:

1. **Block Diagram** – High-level view of system components.
2. **Layered Diagram** – Organizes the system into layers (e.g., UI, business logic, data layer).
3. **Deployment Diagram** – Shows hardware and networking elements.

4. **Sequence Diagram** – Depicts interactions between system components.
5. **Flowchart** – Represents logical workflows.

Comparison Table:

Diagram Type	Purpose	Advantages	Disadvantages
Block Diagram	High-level representation	Simple and easy to understand	Lacks detail
Layered Diagram	Defines modular structure	Enhances scalability	Can be complex
Deployment Diagram	Shows hardware setup	Useful for system admins	Requires deep knowledge
Sequence Diagram	Defines interactions	Highlights dependencies	Can become too detailed

---

SET 2

A. Weather Modeling using Quadratic Solution

Stages of Implementation:

1. **Hard-Coding Variables** - Directly define temperature, humidity, etc.
2. **Keyboard Input** - Allow user input.
3. **Reading from a File** - Fetch data from a text file.
4. **Single Set of Input** - Process one dataset.
5. **Multiple Sets of Inputs** - Process multiple datasets dynamically.
6. **Save Versions on GitHub** - Maintain versions and debug issues.

B. Cohesion vs. Coupling

Aspect	Cohesion	Coupling
Definition	Degree of internal module connectivity	Interdependency between modules
High vs. Low	High cohesion is better	Low coupling is better

Aspect	Cohesion	Coupling
Maintainability	Easier to maintain	Harder to modify if highly coupled

---

SET 3

A. Software Requirements Specification (SRS)

Types of Requirements:

1. **Functional** – Defines what the system should do (e.g., login feature).
2. **Non-Functional** – Defines system constraints (e.g., performance, security).

B. Process Model Comparisons

Model	Description	Advantages	Disadvantages
Waterfall	Linear, phase-wise development	Simple	Rigid, late error detection
Iterative	Cyclical improvements	Adaptive	Can be costly
Spiral	Risk-based iterations	Good for complex projects	Requires expertise

---

C. Scrum Process Model

- **Roles:** Product Owner, Scrum Master, Team.
- **Events:** Sprint, Daily Standup, Review.
- **Artifacts:** Product Backlog, Sprint Backlog.

SET 4

A. JIRA Tool & Project Planning

Steps in JIRA:

1. **Create a Project** – Choose Scrum or Kanban.
2. **Define User Stories** – List tasks and features.

3. **Assign Tasks** – Distribute work among team members.
4. **Monitor Progress** – Track work using boards.

B. Comparison of RAD, RUP, and Prototyping Models

Model	Description	Advantages	Disadvantages
RAD	Rapid iterations with reusable components	Fast development	Needs skilled team
RUP	Phased iterative model	Well-documented	Complex and costly
Prototyping	Early working model	User feedback-driven	Risk of scope creep

Conclusion:

This document covers essential topics in software engineering, including GitHub, software architecture, process models, and tools like JIRA. Use this for comprehensive exam preparation!