



AGILE • JIRA • GIT/GITHUB (3+ YEARS EXPERIENCE)

REAL-TIME INTERVIEW QUESTIONS & ANSWERS

AGILE METHODOLOGIES

1. Explain your Agile experience.

"I worked in a Scrum Agile environment with 4-week sprints, attending sprint planning, standups, reviews, retrospectives, and using JIRA for tracking."

2. Why 4-week sprints?

"Because our project modules were complex and required more dev + QA time with fewer disruptions."

3. Agile ceremonies you attend:

Sprint Planning, Daily Stand-up, Backlog Refinement, Sprint Review, Retrospective.

4. Task estimation method:

Planning Poker + Story Points based on effort, complexity, and risk.

5. How do you handle mid-sprint changes?

"Through change-control; new items go to backlog unless critical."

6. Challenges faced in Agile:

Scope changes, unclear requirements, QA crunch, delays. Solutions: refinement, shift-left testing, automation.

JIRA INTERVIEW QUESTIONS

1. How do you manage tasks in JIRA?

Create stories/subtasks, update status, log hours, comment progress.

2. JIRA reports you use:

Burndown, Velocity, Control Chart, CFD, Release Notes.

3. How do you log bugs?

With severity, priority, screenshots, steps to reproduce, linked story.

4. Workflow followed:

To Do → In Progress → Code Review → QA → Done.

5. JIRA dashboards:

For sprint progress, blockers, open bugs, velocity.

GIT & GITHUB INTERVIEW QUESTIONS

1. Branching strategy:

Git Flow: main, develop, feature/*, release/*, hotfix/*.

2. Pull Request process:

Develop → feature branch → commit → push → PR → review → merge.

3. Handling merge conflicts:

Fetch latest → resolve via editor → test → commit.

4. Git fetch vs Git pull:

Fetch = download; Pull = fetch + merge.

5. Revert a commit:

git revert (safe) or git reset (rewrite history).

REAL-TIME SCENARIOS

1. Sprint ending but task pending:

Inform Scrum Master, identify blockers, split/move story.

2. Build failing after merge:

Check logs → fix → re-run pipeline → notify team.

3. QA found bugs:

Reopen → fix → retest → close.

4. Team member not updating JIRA:

Raise in stand-up → remind → escalate if needed.

5. PO adds urgent requirement:

Discuss impact → re-estimate → swap scope or move to next sprint.

FUSION SOFTWARE INSTITUTE