# Top 20 Technical Interview Questions and Answers for Junior Business Analyst (IT Projects)

# 1. Explain the Software Development Life Cycle (SDLC).

The SDLC is the step-by-step process used to design, develop, test, and deploy software. Common phases: Requirements → Design → Development → Testing → Deployment → Maintenance.  
Example: As a BA, I create clear requirement documents in the “Requirements” phase, so developers and testers understand the scope before coding begins.

## 2. What is Agile methodology? How is it different from Waterfall?

Agile delivers software in small, iterative increments with frequent stakeholder feedback. Waterfall is linear and sequential.  
CAR Example:  
C: Our team needed faster delivery of features.  
A: I introduced Agile sprints with user story breakdowns.  
R: Stakeholder feedback cycles shortened from 3 months to 2 weeks.

## 3. Describe the Scrum framework.

Scrum is an Agile framework with time-boxed sprints, roles (Product Owner, Scrum Master, Team), and ceremonies (stand-ups, sprint planning, reviews, retrospectives).  
Example: I facilitated daily stand-ups ensuring blockers were raised early and sprint goals stayed on track.

## 4. What is Kanban and when would you use it?

Kanban visualizes work on a board (“To Do → In Progress → Done”) to improve flow and limit work in progress.  
Example: Used Kanban in maintenance projects where priorities shifted daily to manage ad-hoc bug fixes efficiently.

## 5. How do you write a good User Story?

Use the format “As a [user], I want [to do something] so that [I get value].”  
CAR Example:  
C: Stakeholders provided unclear requirements.  
A: I rewrote them into INVEST-compliant user stories in Jira.  
R: Developers estimated stories accurately and reduced rework by 20%.

## 6. What are Acceptance Criteria and why are they important?

They define the conditions that must be met for a story to be considered complete and testable.  
Example: For a login feature, acceptance criteria included valid/invalid credentials, password reset, and error messages.

## 7. What is a Requirements Traceability Matrix (RTM)?

RTM maps each requirement to its test case or deliverable, ensuring coverage.  
CAR Example:  
C: QA team missed validating some business rules.  
A: I created an RTM linking all requirements to test IDs.  
R: Achieved 100% requirement coverage in UAT.

## 8. Define Definition of Ready (DoR) and Definition of Done (DoD).

DoR: Criteria a story must meet before being accepted into a sprint.  
DoD: Criteria a story must meet to be considered complete.  
Example: DoR = clear description + AC + mockups + estimates; DoD = code complete + tested + accepted by PO.

## 9. What is User Acceptance Testing (UAT)?

UAT is the final testing phase where end-users verify if the solution meets their needs.  
CAR Example:  
C: Business users needed confidence before go-live.  
A: I created UAT scripts and coordinated defect triage.  
R: 95% of UAT cases passed; system launched smoothly.

## 10. How do you create a BPMN (Business Process Model and Notation) flow?

Use tools like Lucidchart or Bizagi. Identify start → tasks → decisions → end points using standard BPMN symbols.  
Example: Modeled an “Employee Leave Request” process to identify redundant approvals and cut processing time by 30%.

## 11. What is UML and how is it used?

Unified Modeling Language visualizes system behavior and structure (use-case, sequence, class diagrams).  
Example: Created a sequence diagram showing data flow between user interface, API, and database for a new portal.

## 12. How have you used Jira and Confluence in your projects?

CAR Example:  
C: Project lacked transparent progress tracking.  
A: I managed user stories and sprint boards in Jira and documented meeting notes and requirements in Confluence.  
R: Improved collaboration—team velocity increased by 15%.

## 13. Explain how you would gather and document requirements.

CAR Example:  
C: Stakeholders had differing expectations.  
A: Conducted interviews and workshops, summarized findings into BRD and user stories.  
R: Alignment achieved early; change requests reduced by 25%.

## 14. What are functional and non-functional requirements?

Functional: What the system should do (features).  
Non-functional: How it should perform (speed, security, usability).  
Example: Login validation = functional; response within 2 seconds = non-functional.

## 15. What is your approach to data analysis and SQL as a BA?

CAR Example:  
C: Needed to validate transaction data quality.  
A: Ran SQL queries (SELECT, JOIN, COUNT) to identify duplicate records.  
R: Found 200 invalid entries; data team corrected them before release.  
Example Query:  
SELECT customer\_id, COUNT(\*) FROM orders GROUP BY customer\_id HAVING COUNT(\*) > 1;

## 16. How do you use Excel in BA work?

For data cleaning, pivot tables, charts, and simple requirement matrices.  
Example: Created a pivot chart comparing monthly sales by region to support stakeholder decision-making.

## 17. Explain your experience with Power BI (or any dashboarding tool).

CAR Example:  
C: Management needed real-time KPI visibility.  
A: Connected Power BI to SQL Server and built dashboards with slicers for region and product.  
R: Reduced manual reporting time by 4 hours per week.

## 18. How would you handle incomplete or conflicting requirements?

CAR Example:  
C: Two departments requested opposite features.  
A: Facilitated a joint workshop to discuss priorities and clarify objectives.  
R: Agreed on a hybrid solution that met both departments’ key needs.

## 19. How do you ensure requirements are clearly communicated to developers and testers?

CAR Example:  
C: Developers misinterpreted a business rule.  
A: Introduced acceptance criteria tables and walkthrough sessions using Confluence pages.  
R: Reduced post-release defects by 40%.

## 20. Give an example of a real-world BA problem you solved.

CAR Example:  
C: Customer onboarding took 10 days due to manual form handling.  
A: Mapped process in Lucidchart, identified redundant steps, proposed online form automation.  
R: Onboarding time reduced to 3 days; customer satisfaction rose by 25%.