

A Guide to Business Analyst & Best Practices

**FINDING & LANDING
THE PERFECT JOB IN
2025**





EMBARK ON YOUR **BUSINESS ANALYST** JOURNEY

Copyright ©Recruit Riders Technologies

All rights reserved

Unauthorized reproduction or distribution of this guide in any form, including photocopying or electronic means, is strictly prohibited without prior written consent from the publisher. For inquiries, please reach out to inquiry@recruitriders.com





What is Business Analysis ?

Business analysis involves identifying and analyzing business problems, needs, and opportunities to support an organization's strategic goals and vision. It requires active participation throughout the Software Development Life Cycle (SDLC) to ensure that proposed solutions align with business objectives and effectively address the organization's challenges and opportunities.

What Does a Business Analyst Do ?

A Business Analyst is a professional responsible for conducting business analysis in IT system development projects. Their duties involve evaluating business needs, supporting the collection of user requirements, organizing and prioritizing those requirements, validating critical project outputs, identifying potential improvements in business processes, and facilitating clear communication between business and IT stakeholders.

Understanding the Need for a Business Analyst Role

(a) In IT system development, communication gaps frequently arise between IT staff and business users due to differences in knowledge, skills, backgrounds, and perspectives. Business users may struggle with IT terminology and technical solutions, while IT staff might find it challenging to understand business terms, functions, processes, and the organizational context. These gaps can hinder the identification of genuine business



needs, the understanding of requirements, and the design of effective systems. The challenges are further amplified in outsourced IT projects, where external contractors might lack familiarity with the government environment and specific business processes. To address these issues, the role of a Business Analyst (BA) is essential to facilitate collaboration between users and IT teams throughout the SDLC.

(b) During the project initiation phase, before a project team is established, a BA can identify opportunities for improvement in the current state by developing well-founded business cases. These cases help justify IT investments and establish a clear scope and estimates for the project. The BA role is particularly crucial for scoping and planning large-scale projects at this early stage.

(c) When resources and demand warrant, it is advisable to establish a permanent BA role. This ensures ongoing system maintenance, continuous support, and enhancements to meet evolving business needs effectively.

Advantages of Having a Dedicated **Business Analyst**

Defines a clear project scope from a business perspective.

Identifies real business needs & manages user expectations.

Develops strong business cases with realistic resource estimates.



Designs IT systems that meet user needs and deliver benefits.

Produces clear requirements for accurate tender specifications.

Ensures system quality before user review and acceptance.



Primary Role of a Business Analyst in Projects

The primary role of a Business Analyst (BA) is to act as a bridge between the end-users and the IT team, assisting in identifying and analyzing business challenges and requirements. They collaborate closely with the Systems Analyst (SA) throughout the IT system development process to ensure alignment with business objectives.

Assist in Creating a Comprehensive Business Case.

Facilitate the Elicitation & Analysis of Requirements

Evaluate System Feasibility & Implementation Readiness

Plan and Monitor the Business Analysis Activities

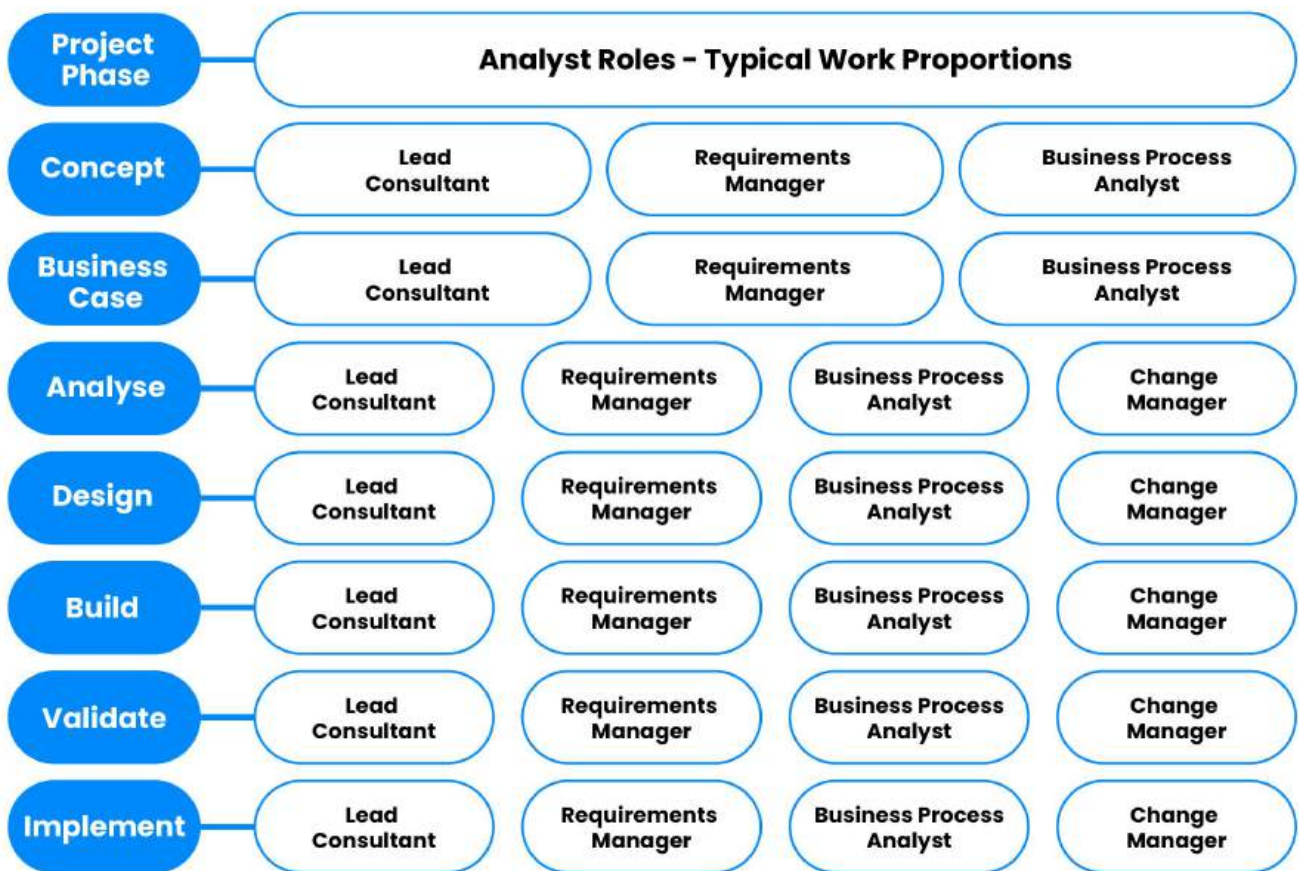
Ensure Continuous Improvement & Post-Implementation Support





Business Analyst's Role in System Development

(a) Business analysis activities are integrated into multiple phases of the SDLC. The key responsibilities of a BA in each phase of the SDLC are summarized in the figure below.

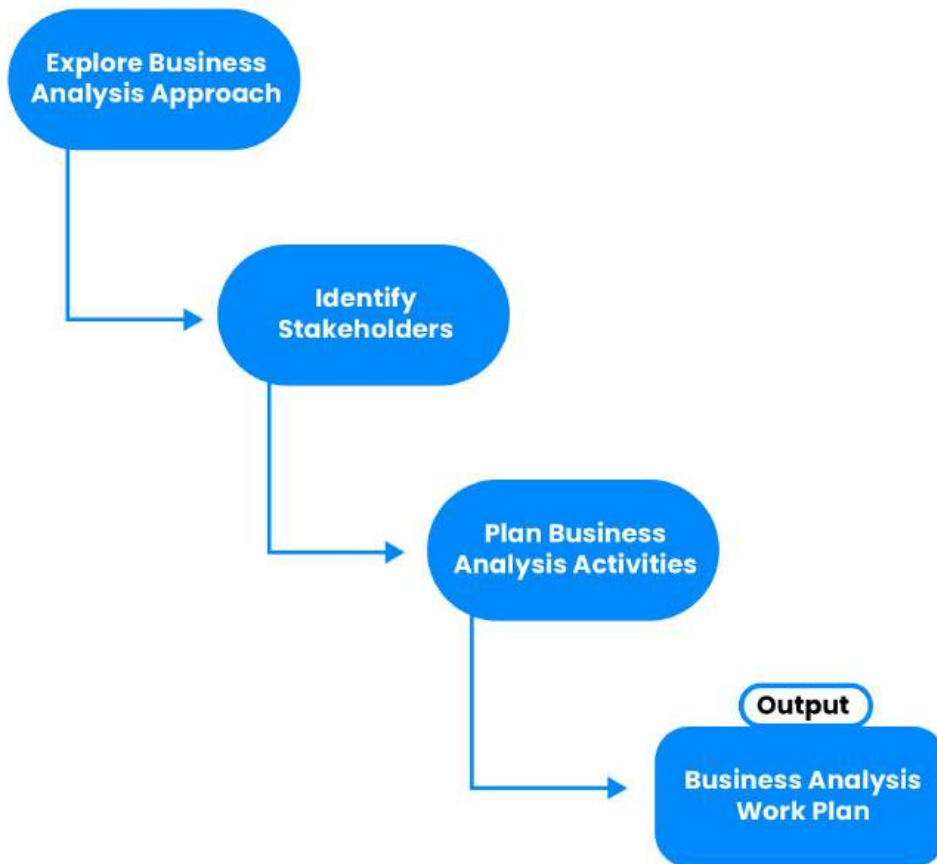


(b) More details of the key BA activities as mentioned above are explained in the following sections of the Document



Business analysis Planning

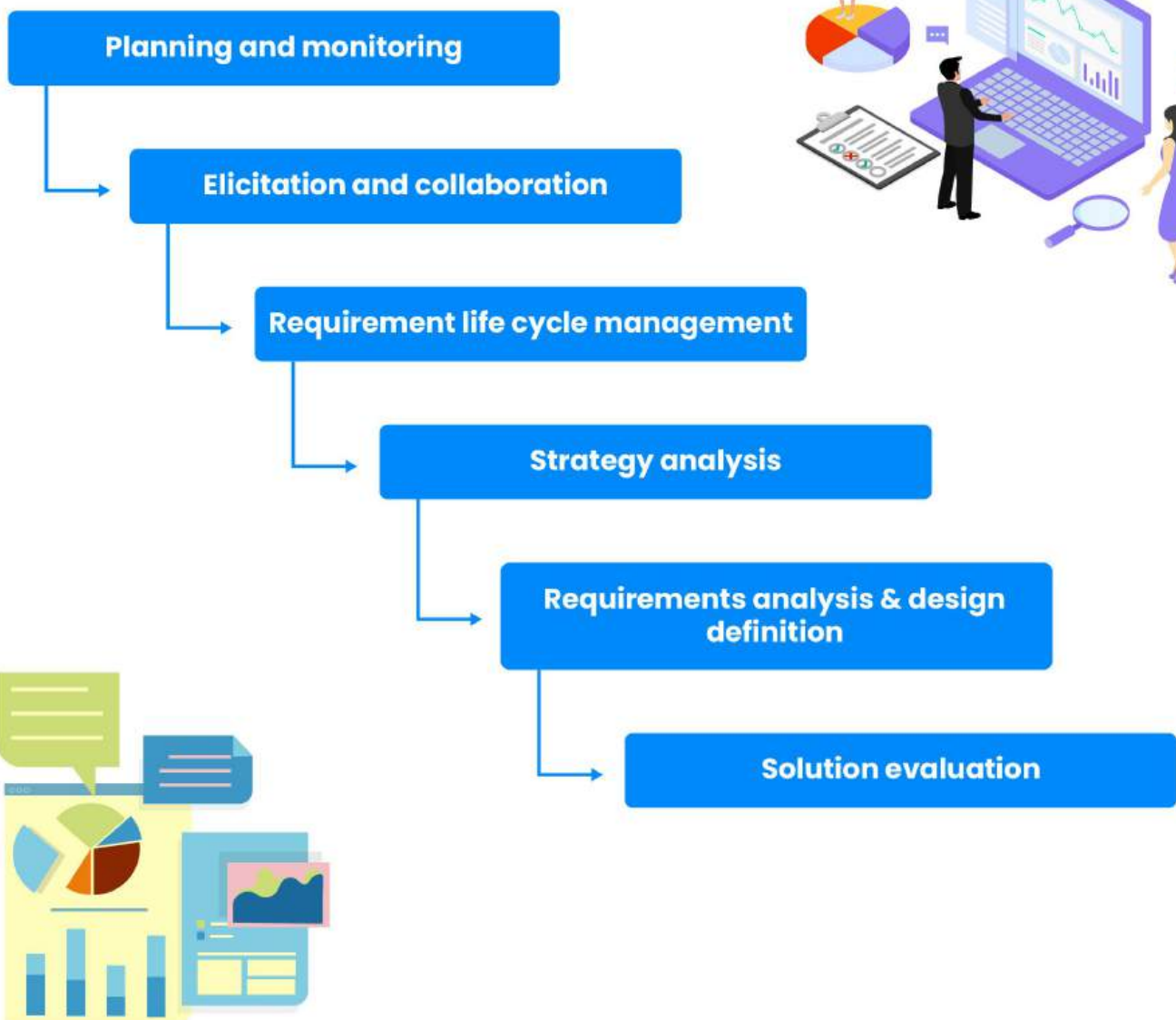
During the SDLC's Project Planning Phase, B/D should outline a high-level plan for the business analysis tasks required for the IT system development, as illustrated in the figure.





Steps of Business Analysis

There are common stages of business analysis that companies usually do. These stages are also called knowledge areas. Go Wombat also adheres to this practice, so it is high time to discover more details about these areas.





Business Analysis Process Overview

1-Planning and Monitoring

This phase involves organizing and coordinating efforts between stakeholders and business analysts. Analysts create a business analysis plan, select methodologies, and outline interaction procedures to enhance communication. It ensures informed decision-making and provides all necessary information to keep the project on track.

2-Elicitation and Collaboration

Analysts gather and validate information from stakeholders through interviews, research, and local practices. This stage ensures accurate requirements elicitation and collaboration between stakeholders and analysts.

3-Discovery Phase

A detailed analysis of the product, including its characteristics and technical requirements, is conducted. This phase spans all stages of business analysis and focuses on defining and refining product requirements for a comprehensive understanding.

4-Requirement Life Cycle Management

This stage involves managing requirements and design information throughout the project. Analysts ensure consistency and alignment between business needs, stakeholder requirements, and design implementation. It includes controlling how requirements are delivered and ensuring long-term accessibility for clients.

5-Strategy Analysis

Analysts collaborate with stakeholders to identify strategic requirements and create efficient solutions to achieve business goals. This phase helps



define long-term strategies, market positioning, and ways to demonstrate product value.

6-Solution Evaluation

The final phase evaluates solutions for effectiveness and value. Prototypes are tested with focus groups to gather feedback and make improvements, addressing issues early to minimize risks and maximize product success.





Business analysis techniques

All steps in business analysis go along with the techniques analysts use to achieve the necessary result. There are more than 50 available techniques, but we will focus on the important techniques.



Interviews



Workshops



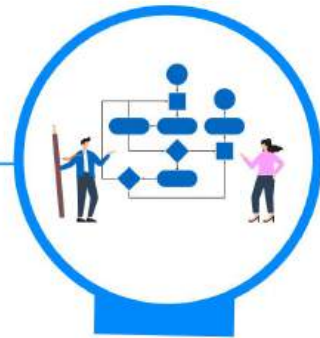
**Document
Analysis**



Brainstorming



**Mind
Mapping**



**Process
Modelling**



**Business Rules
Analysis**



**Risk Analysis &
Management**



Business analysis techniques

1. Interviews: Engage stakeholders to gather crucial information, develop strategies, and assess the efficiency of business analysis.
2. Workshops: Facilitate collaborative meetings to build analysis plans, identify solutions, manage changes, set priorities, and uncover challenges while fostering stakeholder engagement.
3. Document Analysis: Review existing documentation to extract valuable data, support strategy planning, and enhance collaboration.
4. Brainstorming: Generate ideas and gather diverse input from stakeholders to define requirements and develop innovative solutions.
5. Mind Mapping: Use visual diagrams to organize tasks, ideas, and requirements for a deeper understanding of the project and its needs.
6. Process Modeling: Create graphical models of business processes to analyze workflows and define an approach to business analysis.
7. Business Rules Analysis: Identify and refine business rules (definitional and behavioral) to guide decision-making and align with organizational goals.
8. Risk Analysis and Management: Identify potential risks in software development and develop mitigation strategies to ensure project success.

These techniques are adaptable and applied based on project needs and characteristics.

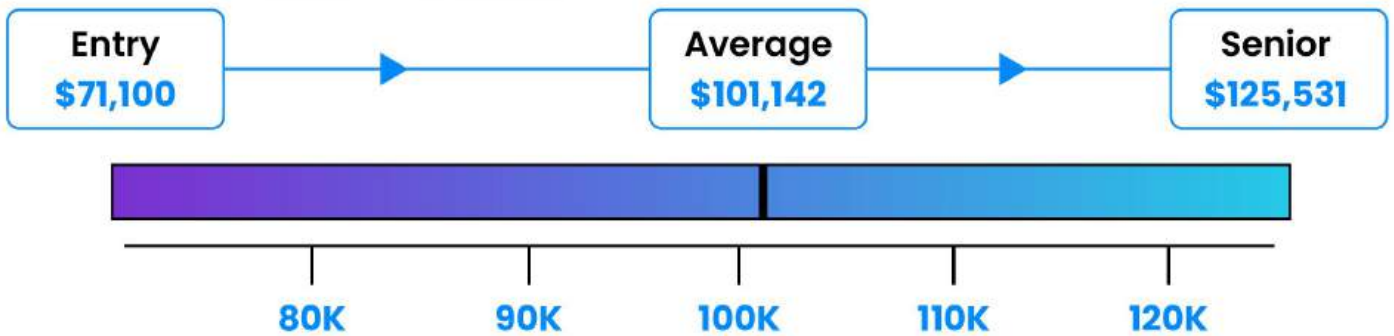


Average salary range for **Business Analyst**



Average Hourly Rate
\$48.63 / hr

Average Bonus
\$4,036/yr



The salary for a Business Analyst varies based on experience, location, and industry. In the United States, as of January 2025, the average annual salary for an entry-level Business Analyst is approximately \$70,785, with most earning between \$62,983 and \$79,055.

SOURCE - SALARY.COM

Senior Business Analysts, with more than 4 years of experience, can expect average base salaries around \$105,989, typically ranging from \$95,000 to \$120,000 or more.

SOURCE - KNOWLEDGEHUT

These figures can fluctuate based on factors such as location, industry, and individual qualifications.



What are the most commonly used tools by Business Analysts ?

Requirement Gathering & Documentation Tools



Data Analysis and Visualization Tools



Analytical and Statistical Tools





Essential soft skills required for a **Business Analyst**

Soft skills enable business analysts to communicate effectively, collaborate with stakeholders, and resolve challenges. They are essential for gathering accurate requirements, managing conflicts, and adapting to changes, ensuring smooth project execution and alignment between business needs and technical solutions.



Communication
Skills



Problem-
Solving



Stakeholder
Management

Adaptability



Time
Management



Critical
Thinking





Do's and Don'ts for a Business Analyst

There are common stages of business analysis that companies usually do. These stages are also called knowledge areas. Go Wombat also adheres to this practice, so it is high time to discover more details about these areas.

Do's	Don'ts
<ul style="list-style-type: none">• Understand the Project Scope• Engage with Stakeholders• Understand the Business Environment• Define Communication Channels• Document Everything• Ask Questions	<ul style="list-style-type: none">• Assume Knowledge• Overpromise• Neglect the Big Picture• Resist Change• Work in Isolation• Ignore Feedback





Identify Stakeholders

(a) **Identifying Stakeholders:** This step focuses on recognizing stakeholders who will either participate in or have an impact on the business analysis activities, requirements, or outcomes of the IT system project. Stakeholders can include business owners, end users, and others affected by the project.

(b) **Initial Stakeholder Assessment:** The BA and Internal PM collaborate to identify both internal and external stakeholders and perform an initial assessment. Together, they create a stakeholder register that groups stakeholders, records their locations, and notes any special considerations such as remote work, shift schedules, or long leave periods. The register and stakeholder involvement depend on the project scope, system development methodology, and the organizational structure of the department.

(c) **Using Stakeholder Analysis Techniques:** The BA can leverage techniques from the Practice Guide to Project Management (PGPM) to identify stakeholders, evaluate their roles, and understand their communication needs and expectations throughout the project.

(d) **Stakeholder Register Updates:** Once the IT project organization is established during the Project Funding Request Phase, the stakeholder register must be updated to include the confirmed appointments of PSC and PAT members.

(e) **Applying the RACI Model:** The RACI Model, outlined in the PGPM, is a valuable tool for defining roles and clarifying the level of responsibility and involvement required from various team members to complete project deliverables.



Plan Business Analysis Activities

BA is responsible for planning and scheduling business analysis activities and deliverables for the project. This involves referencing similar past projects, if applicable, to outline activities and estimate required resources. Key tasks include:

(i) Plan Deliverables: Identify and outline deliverables to be produced in each SDLC phase.

(ii) Determine Activities: Define the business analysis tasks required to create these deliverables, such as developing business cases during the Project Funding Request Phase or gathering detailed user requirements in the SA&D Phase.

(iii) Establish Communication Plan: Work with the Internal PM to define communication methods and frequency for stakeholder engagement. Estimate Resources: Determine required resources, including BAs and effort, for inclusion in the project plan.

(iii) Establish Communication Plan: Collaborate with the Internal PM to create a communication plan detailing methods (e.g., video calls, emails, meetings) and frequency (e.g., ad hoc, weekly, monthly) for engaging with stakeholders.

(iv) Estimate Resources: Calculate the resources needed, such as the number of BAs and effort (man-days), and include this in the project management plan.





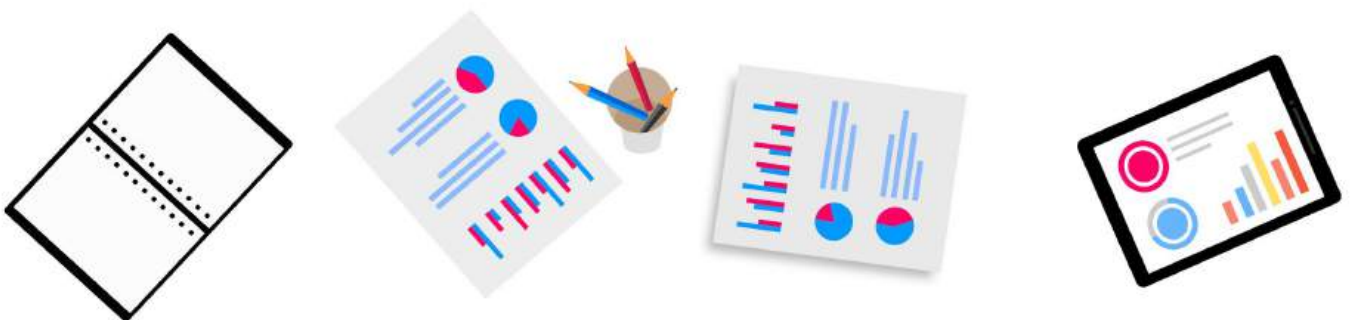
Business Analysis Work Plan

(a) After completing the initial activities, a Business Analysis Work Plan should be created. This plan outlines the approach, involved stakeholders, key activities, target deliverables (e.g., User Requirements Document), work schedule, and estimated resources.

(b) Whenever possible, the BA should present the Business Analysis Work Plan to the Project Owner or senior management for endorsement before the project begins. This ensures sufficient resources are allocated for business analysis, early support and commitment from stakeholders (including IT staff) are secured, and the planned activities and schedule are approved by the Internal PM.

(c) The Business Analysis Work Plan should be revised in later SDLC phases to address any changes in conditions or newly identified project challenges. Detailed work plans for specific activities may also be developed if needed.

(d) The Business Analysis Work Plan must align with the overall project management plan, which is prepared after project funding is approved. Relevant elements, such as the work schedule, may be incorporated into the project management plan as required.



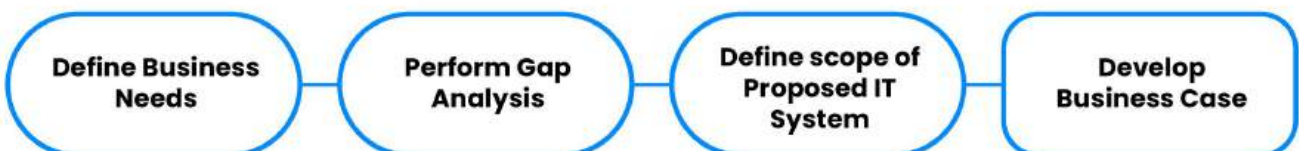


Defining the Business Case

(a) A business case serves to identify opportunities for improvement and assess the benefits of proposed investments, offering essential details for preparing a funding request. It typically includes objectives of the investment, business challenges and opportunities, policy support, the proposed IT solution, and an analysis of costs and benefits.

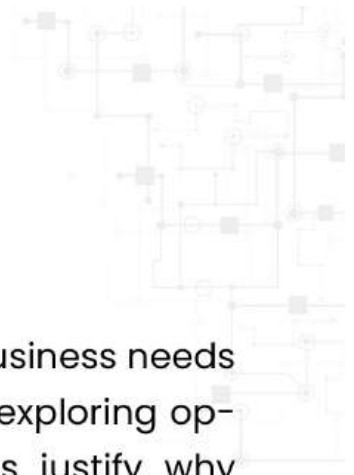
(b) The BA should work closely with relevant stakeholders to gather the necessary business information and support the IT team or other responsible parties in the department to develop the business case and propose the IT system.

(c) The key activities undertaken by the BA are outlined in the accompanying figure.



This section offers a concise overview of the process for developing a business case.





Define Business Needs

(a) The BA plays a key role in identifying and defining the business needs of the relevant area by understanding existing issues and exploring opportunities for service enhancements. This process helps justify why changes to the current system are necessary.

(b) Input from stakeholders at various levels, including senior management, middle management, and operational staff, is essential to understand their specific business needs. The following steps can guide the process:

(i) **Establish Business Goals and Objectives:** Define the broader targets and specific outcomes that the organization aims to achieve for the business area. Goals are strategic, long-term aspirations, while objectives are detailed, actionable steps derived from those goals.

(ii) **Analyze Business Problems and Opportunities:** Examine current challenges, identify root causes, and assess system limitations. This analysis helps uncover improvement opportunities to address these challenges and align with business goals.

(iii) **Outline Desired Outcomes:** Define the benefits stakeholders expect from the proposed IT system, such as enhanced work safety, improved user satisfaction, or compliance with new regulations.

By following these steps, the BA ensures that business needs are clearly articulated and aligned with organizational objectives, creating a strong foundation for the IT system development.



Perform Gap Analysis

- (a) Once the business needs are defined, the current state—such as existing business processes, functions, system features, services, and events—should be analyzed. This involves gathering input from IT staff and stakeholders, including business owners, to understand how people, processes, technology, and architecture support the business.
- (b) A gap analysis is conducted by comparing the current state with the desired outcomes to identify any gaps that hinder the achievement of business needs.
- (c) If no gaps exist, the current state is sufficient, and an IT project may not be necessary. However, if gaps are identified, the issues preventing success must be addressed. For IT-related gaps, input and assistance from IT staff should be sought. Relevant findings, such as current processes and identified issues, may be included by the SA in the Current Environment Description of the SA&D Report during the SA&D phase.
- (d) Techniques like SWOT Analysis and document review can help identify gaps and areas for improvement.

Scope Definition for Proposed IT System

- (i) Identify In-Scope Components: Define key business-related elements, such as major system functions and features, involved divisions or teams, and business processes to be improved or redesigned.
- (ii) Highlight Major Dependencies: Identify critical business-related dependencies, such as laws, regulations, government policies, and external factors like integration with other systems or government IT infrastructure.



Defining System Requirements

(a) During the Feasibility Study (FS) and SA&D phases, developing clear and accurate requirements is essential for addressing business needs and system design. The BA's role is to support the IT team in eliciting and analyzing these requirements.

(b) Requirements are generally divided into two categories:

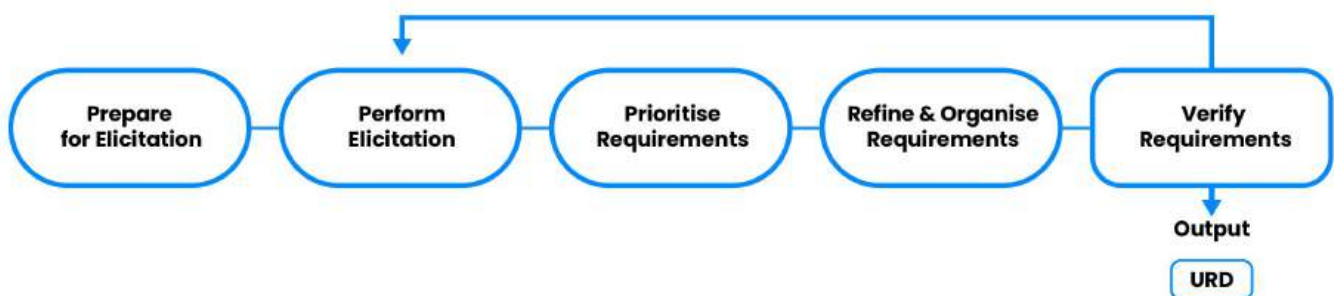
- **Functional Requirements:** These outline the system's functions or features that users utilize to support business operations (i.e., what the system should do to deliver business value).

- **Non-Functional Requirements:** These specify how the system should operate, covering aspects like audit, control, security, and service levels. Technical non-functional requirements (e.g., browser versions or IT architecture) are managed separately by the IT team during the SA&D phase.

(c) The BA provides input to ensure the system meets business needs, supports readiness, and enables effective implementation.

(d) The following sections outline the processes for requirements analysis and system design evaluation.

Elicitation and Analysis of Requirements





(a) To prepare for requirements elicitation, the BA should refer to key details from the Business Analysis Work Plan and Business Case, including:

- System scope and objectives
- Business needs
- Work schedule
- Planned activities
- Target deliverables
- Stakeholder register



(b) Preparation involves:

(i) Choosing elicitation techniques and defining the scope for each, such as brainstorming, document analysis, focus groups, interviews, observation, prototyping, or surveys.

(ii) Preparing materials for the selected techniques, including surveys, interview questions, discussion guides, and existing documentation.

(iii) Scheduling necessary resources like participants, venues, and equipment.

(iv) Creating an elicitation work plan with a schedule.

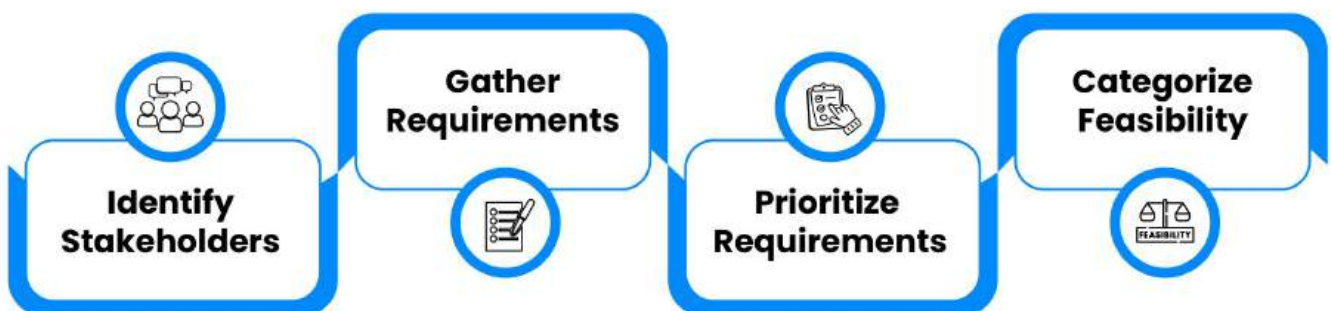
(v) Informing stakeholders and securing agreement on the elicitation work plan.

(c) The BA should confirm the scope and schedule with stakeholders and establish a process for validating and signing off on the elicited results. This ensures alignment among all parties and helps maintain clarity throughout the project.



Facilitate Elicitation Processes

- (a) After completing preparation, the BA should carry out elicitation activities as outlined in the elicitation work plan.
- (b) The process begins with reviewing all relevant business and system documentation, including policies, procedures, and existing system details.
- (c) To avoid scope creep, ensure that all requirements align with business goals and objectives, remain within scope, and address current issues and needs. Acceptance criteria for functional requirements should be defined to measure quality and meet user expectations.
- (d) During elicitation, record requirement attributes such as source, value to users, and priority to aid in managing requirements throughout the project. Documenting the time spent on elicitation can also support future planning.
- (e) All stakeholder-provided requirements must be properly documented, summarized, and maintained for project use.





Prioritize Requirements

(a) Requirements should be prioritized to determine their order for analysis and implementation. Key criteria include:

(i) **Business Value:** Focus on requirements with the highest business impact.

(ii) **Stakeholder Agreement:** Prioritize based on input from key stakeholders.

(iii) **User Impact:** Consider how requirements affect user processes or workload.

(iv) **System Impact:** Account for dependencies on other systems.

(v) **Compliance:** Address regulatory or policy requirements first.

(vi) **Dependencies:** Focus on requirements supporting higher-priority needs.

(vii) **Urgency:** Prioritize time-sensitive requirements.

(viii) **Risk Mitigation:** Address high-risk requirements early.

(ix) **Ease of Implementation:** Implement simpler requirements for quick user adoption.

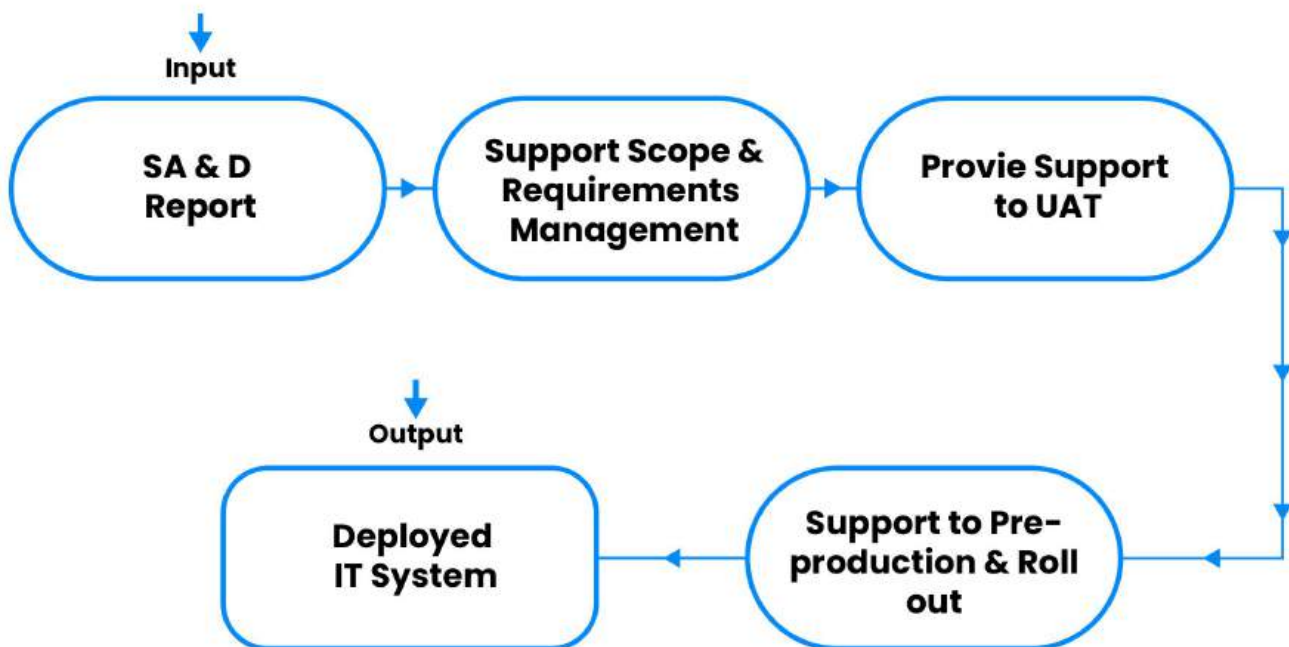
(b) Challenges may occur if stakeholders over-prioritize requirements or project teams misjudge technical complexities, but techniques like MoSCoW Analysis can aid in effective prioritization.



Implementation and Approval Process

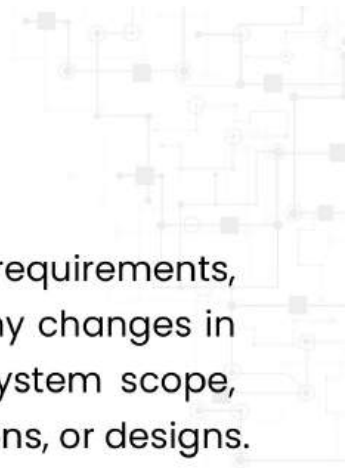
The BA supports the project team during the System Implementation Phase through three key activities:

- (i) Assisting with system scope and requirements management,
- (ii) Supporting user acceptance testing (UAT), and
- (iii) Aiding in pre-production and system roll-out.



Supporting System Scope and Requirements Management

- (a) During implementation, the BA assists stakeholders in resolving conflicts, achieving consensus on changes to scope and requirements, and securing approval for change requests.



(b) The system scope defines the boundary for managing requirements, and only requirements within this scope are addressed. Any changes in business needs or technical solutions may impact the system scope, leading to updates in approved requirements, system options, or designs.

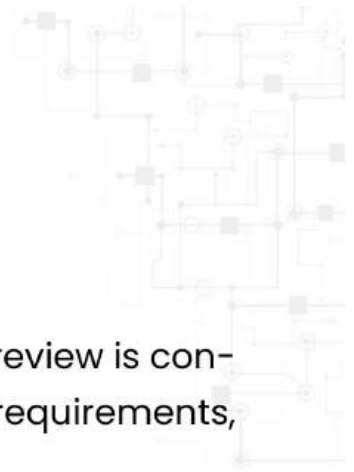
(c) Requirements from the URD, endorsed during the SA&D Phase, serve as the baseline for changes. The BA prepares change requests, submits them to the Internal PM for PSC approval via PAT, and maintains an updated list of approved requirements.

(d) The BA mediates stakeholder conflicts on requirements or technical changes, ensuring communication and resolution before seeking approval. In urgent cases, the BA and Internal PM may group change requests for batch approval during checkpoint meetings to reduce administrative burdens.

Ensuring the outsourcing contract includes a clear definition of requirements to avoid disputes.

Requesting contractors to use prototypes for early user feedback before proceeding with detailed programming.

The BA supports UAT by engaging key users early to ensure their commitment, assisting in creating a comprehensive test plan with cases, schedules, and steps, coordinating with stakeholders for interface testing, and providing hands-on training to familiarize users with the system. Additionally, the BA collaborates with users to develop a realistic testing schedule, ensures test data and expected outcomes are documented, helps conduct tests and consolidates findings, and works with the IT team to resolve and re-test any failed cases.



System Evaluation

(a) After the IT system is rolled out, a post-implementation review is conducted to ensure the system meets business needs, user requirements, and delivers the expected benefits efficiently.

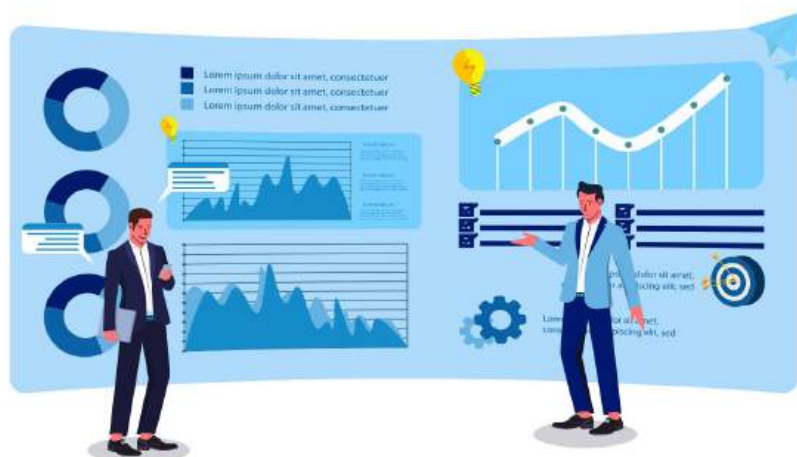
(b) The BA supports users in evaluating system performance by analyzing how it is used, assessing its business value, and identifying areas for improvement.

(c) The evaluation involves comparing planned versus actual performance using predefined metrics, such as:

- Quantitative metrics: Time saved, transactions processed, cost savings, revenue, or error rates.
- Qualitative metrics: User satisfaction, feedback, or suggestions.

If results are unsatisfactory, the BA helps identify causes and recommends improvement measures.

(d) The BA provides performance evaluation results to the Internal PM to complete the Post Implementation Departmental Return (PIDR), which must be submitted within six months of system implementation.





ESTABLISHING Business Analyst ROLES

RESOURCES PLANNING

(a) To implement a BA role in IT system development projects, organizations must identify required business analysis activities, estimate the resources needed based on project size and complexity, and recognize that resource needs vary by project.

(b) Complex projects with numerous requirements, intricate business logic, or system dependencies may require multiple BAs, while simpler projects may only need a part-time BA.

(c) Most BA resources should focus on requirements elicitation and analysis. Resource allocation may vary across SDLC phases depending on the development method used.

REQUIRED COMPETENCIES OF BA

(a) A BA's expertise in business processes, workflows, stakeholder roles, and policies, along with knowledge of data protection laws, is crucial for IT project success.

(b) Basic IT knowledge (SDLC, security, project management) and experience in system development enhance planning and communication with IT teams.

(c) Strong communication, teamwork, and adaptability are essential for bridging users and IT teams.



Documentation in **Business Analysis**

Documentation is essential for successful project management and business analysis. In today's fast-paced business and technology landscape, effectively conveying ideas, requirements, and findings is critical. This article explores documenting requirements through BRDs, user stories, and acceptance criteria, as well as communication strategies for business analysts. It highlights the importance of clear and concise documentation in presenting findings and recommendations to stakeholders.

Best Practices for Documenting Requirements

Types of Documents: BRD, FRD, and SRS

Business Requirements Document (BRD)

Purpose: The BRD provides a high-level overview of business objectives, the problems the project aims to solve, and the benefits expected. It guides the project direction without detailing implementation

Best Practice: Involve stakeholders from different departments to ensure all business needs are captured. Use techniques like workshops, interviews, and surveys for requirement gathering.

Real-Time Use Scenario: A company enhancing its website's customer portal may use a BRD to outline improvements in customer satisfaction and operational efficiency.

-A retail company could create a BRD to reduce website load time, enhance search functionality, and add personalized recommendations to improve user experience and boost sales.



How to Create a BRD: Sample Structure

Introduction: Briefly describe the purpose and scope of the project.

Business Objectives: List the high-level business goals in clear, measurable terms.

Background: Provide context about the current challenges or opportunities.

Scope: Define what is included and excluded in the project.

Requirements: Outline the high-level business requirements in bullet points.

Stakeholders: Identify all parties involved or affected by the project.

Benefits: Describe the expected advantages and improvements.

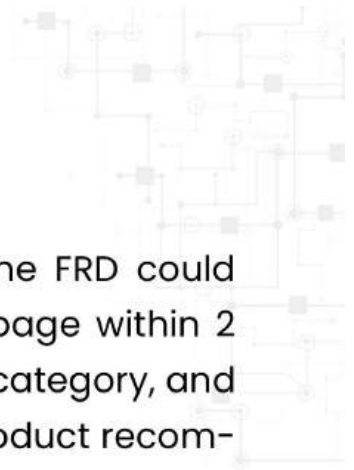
Functional Requirements Document (FRD)

Purpose: The FRD translates high-level business needs into detailed, specific requirements that the product or system must meet, focusing on functionalities.

Best Practice: Use clear, concise language and avoid technical jargon to make the FRD accessible to stakeholders from non-technical backgrounds.

Example: For the customer portal project, the FRD would specify features like user authentication, account management, and support ticket submission.

Best Practice: Use clear, non-technical language to ensure the document is accessible to all stakeholders, facilitating better understanding and alignment.



Example: For the online shopping portal enhancement, the FRD could detail requirements such as "The system must load any page within 2 seconds," "The search function should offer filters for price, category, and user ratings," and "The portal must provide personalized product recommendations based on user's browsing history"

Functional Requirements Document (FRD)

Introduction: Explain the purpose of the document.

Scope: Clarify the focus of the FRD within the project.

Detailed Requirements: List each requirement with a unique identifier. Describe functionalities, user interactions, and data handling processes in detail.

Diagrams/Models: Include any flowcharts or diagrams that can help visualize processes or interactions.

Glossary: Define any technical terms used in the document.

System Requirements Specification (SRS)

The SRS is a comprehensive description of the behavior of the system to be developed, including functional, performance, security, and interface requirements.

Purpose: The SRS provides a complete, detailed specification of the system's behavior, interfaces, performance, and security requirements. It's technical and intended for the development team.

Best Practice: Incorporate use cases and scenarios to provide context for the requirements and help developers understand how end-users will interact with the system.

Real-time Use Scenario: The SRS for the customer portal might include detailed specifications for integration with existing CRM systems,



data encryption standards for user information, and load handling capacity.

Best Practice: Incorporate realistic use cases and scenarios to give developers context.

How to Create an SRS: Sample Structure

Introduction: State the purpose, scope, and intended audience.

Overall Description: Give a high-level view of the application, including user needs and assumptions.

Functional Requirements: Describe in detail what the system should do, using use cases or scenarios.

Non-Functional Requirements: Specify performance, security, and usability requirements.

Appendices: Include any additional information, such as data models or references.

Effective User Stories





User Story Narrative

As a <Consumer of the value>
I want to <Functionality (What)>
so that <Value/benefit (Why)>

Tips

- Write stories from the end user perspective (user can be a system)
- Understand what the user wants and why (not how)
- Don't assume just one kind of user of the system
- Don't confuse user stories with tasks. Stories describe what the user wants and why. Tasks are how we will achieve the goal/benefit of the story.

Acceptance Criteria

Scenario I: <summary of test>
Given <precondition or assumption>
And <additional precondition>
When <perform an action>
And <perform additional action>
Then <this outcome will result>
And <an additional outcome>

Acceptance Criteria are essentially a measure of whether a User Story has been correctly completed. A story may have multiple scenarios to define the acceptance criteria. They also help define the scope and create a shared understanding of the story.

INVEST for Better Stories



Independent: Story can be valued, developed, and delivered on its own

Negotiable: It's a placeholder for requirements to be discussed, developed, and accepted

Valuable: Provides value to the user

Estimable: Small enough so effort to implement can be rapidly determined

Small: Small enough to fit into a single sprint but large enough to be valuable

Testable: Acceptance criteria is testable by Testers or Business Users

User Story Example and Common Problems

Example

As a customer service agent, I need to access the customer's benefit payment details to review the information and communicate it effectively to the customer.

Acceptance Criteria

Scenario 1: Calculate benefits payment - Active

Given that a customer has a benefits plan on file associated with their account

And the benefits plan is active

When the system receives a request to calculate a benefits payment

Then the benefits payment amount is correctly calculated and displayed to the customer service agent

Scenario 2: Calculate benefits payment - Closed

Given that a customer has a benefits plan on file associated with their account

And the benefits plan is closed/not active

When the system receives a request to calculate a benefits payment



Then the system will return a message stating that the customer does not have an active benefits plan

And the system will return a benefit of \$0.00

Common Problems to Avoid

•Not having a conversation

A rich conversation with the team is needed to ensure a common understanding. Use whiteboards or other medium to communicate and ideate on the story.

•Using User Stories for everything - even tasks

-Don't confuse stories (the 'Who', 'What', and 'Why') with tasks (the 'How')
-Stating the 'how' in a story leads to limiting the solution. The team may come up with a better solution if they are aligned to the goal and have autonomy to find a good solution.

•Undefined or wrong user (As a "user", As a "Developer", As a 'Product Owner', etc.)

-Example: "As a developer, I want to replaced the application widget, so that I have maintained the app widget." The Developer role is sometimes used to address issues of technical debt or spikes.

-Identifying the correct user leads to a solution that better meets the needs of that user

•No Acceptance Criteria

-Can cause the wrong definition of development tasks or the wrong estimation.

-Story can fail tests or test cases will cover different criteria due to lack of understanding.

-A good way of discovering Acceptance Criteria is asking questions such as 'What if ...?', 'Where ...?', 'When ...?', 'How will we know we're done?'.



The S.P.I.D.R. approach to splitting stories

SPIKES

Make a large story smaller by pulling out a spike, which is a research activity After which the team will know more.

Sometimes just doing a spike makes the remaining work a manageable size.

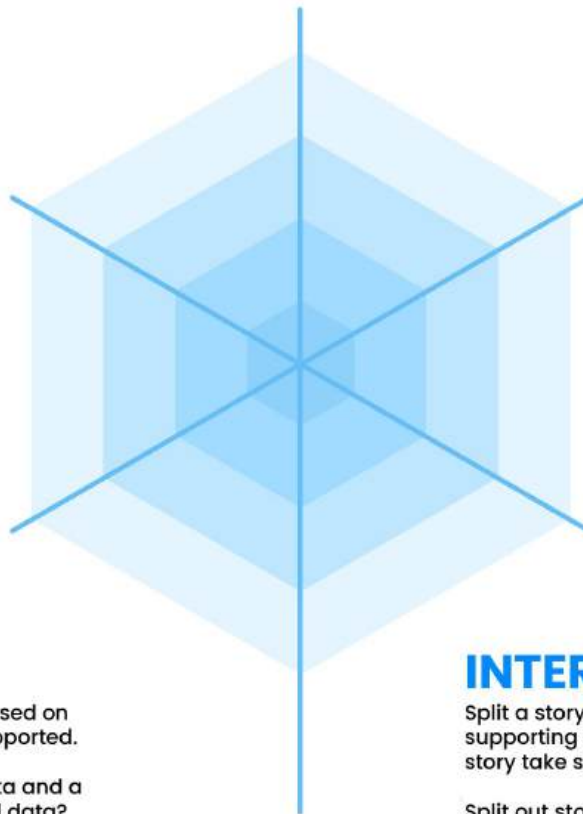
Other times, the new knowledge creating by the spike makes it easier to see ways to split the story.

RULES

Sometimes a story is large because of the business rules, technology standards, or such that must be supported.

Consider relaxing support for these rules in an initial story.

Add support for additional rules in subsequent stories.



PATHS

Consider the paths through a story and split each path into its own story.

Draw a simple flowchart of what happens in a story. Each sequence of steps can be a story.

Expand one big step of the flowchart into a story.

DATA

Look for ways to split the story based on the type of data that must be supported.

Can a first story support valid data and a later story add support for invalid data?

How about frequent types of data and less frequently seen types of data?



INTERFACES

Split a story across multiple interfaces if supporting those interfaces makes the story take significantly longer to develop.

Split out stories by browser type or version, or by different hardware.

Consider building a minimal user interface first or leave styling out of an interface initially.



What is User Acceptance Testing (UAT)?

User Acceptance Testing (UAT) is the final phase of the software testing lifecycle. It is conducted to ensure that the system meets the business requirements and is ready for deployment. During UAT, the end users or business stakeholders validate the system against real-world scenarios and workflows to confirm it works as expected.

Key Aspects of UAT

Focus: Ensures the software aligns with business needs and user expectations.

Testers: Typically conducted by business users, product owners, or stakeholders, with guidance from Business Analysts.

Environment: Uses a production-like environment to replicate real-world conditions.

Output: Approval or rejection based on whether the software meets the acceptance criteria.

Real-Time Use of UAT

For a Business Analyst, UAT is critical to bridge the gap between technical teams and end users. The BA ensures:

- Requirements have been properly translated into functionalities.
- End users validate that the system supports business operations.
- Any defects or gaps identified during UAT are addressed before production.



Example of UAT in Action

Scenario: Online Retail System

Business Requirement: Build an e-commerce website with features for product search, adding to cart, checkout, and payment.

UAT Steps:

Prepare UAT Plan:

Define objectives: Verify that customers can browse products, place orders, and make payments successfully.

List acceptance criteria: Example - "The system should allow users to search for products by category and display results within 2 seconds."

Create UAT Test Cases:

Test Case 1: Search Functionality

Action: User searches for "laptops."

Expected Result: System displays relevant products within the laptops category.

Test Case 2: Payment Processing

Action: User completes an order using a credit card.

Expected Result: Order confirmation is generated, and the payment is successful.

Execute UAT:

Business users log in and perform the tasks as per test cases in a UAT environment.

Example: A retail manager tests placing an order to verify if the inventory is updated post-purchase.



Log Issues:

If the search results take longer than 2 seconds or the payment gateway fails, the issues are logged and sent to the technical team for resolution.

Sign-Off:

After successful execution of all test cases, the BA collects feedback, confirms acceptance, and provides sign-off for production deployment.

Role of the Business Analyst in UAT

Facilitator: Acts as the liaison between end users and the technical team during UAT.

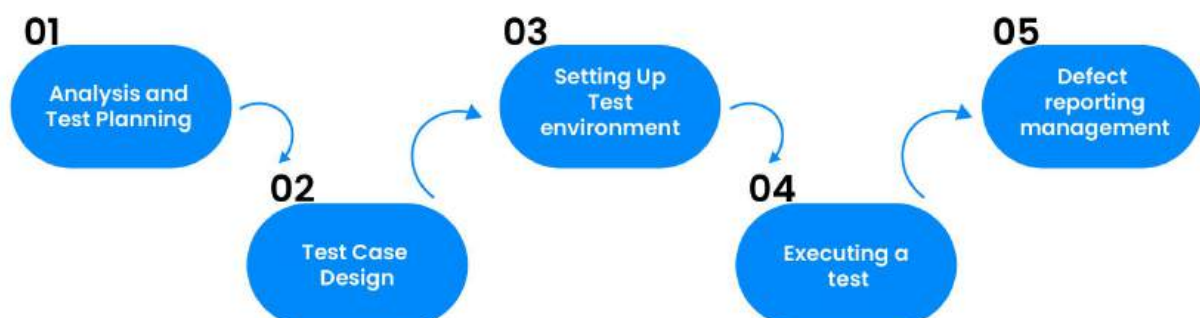
Requirement Validation: Ensures all requirements are met and documented.

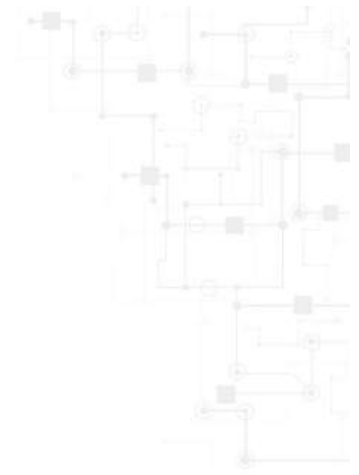
Training and Guidance: Provides UAT testers with training on how to use the system.

Issue Tracking: Logs and prioritizes defects for resolution.

Sign-Off: Secures final approval from stakeholders to proceed to production.

UAT Process





Domain Wise Project Examples

1. Banking Domain

Objective: Modernize the bank's legacy core banking system to enhance performance, reduce downtime, and integrate advanced features like real-time transaction processing.

Goals:

1. Ensure seamless data migration without data loss.
2. Enhance system scalability to support 50% more customers.
3. Improve transaction processing speed by 30%.
4. Enable regulatory compliance for cross-border transactions.

BA Role:

- Gather requirements from branch managers and customers.
- Document workflows for account creation and loan processing.
- Validate migrated data using SQL queries and oversee UAT.

2. Healthcare Domain

Objective: Implement an EHR system to digitize patient records and streamline hospital workflows.

Goals:

1. Reduce patient record retrieval time by 40%.
2. Ensure compliance with HIPAA and GDPR regulations.
3. Integrate lab reports, prescriptions, and appointment scheduling in a



4. Train healthcare staff on system usage.

BA Role:

- Conduct workshops with doctors and nurses to understand their needs.
- Document workflows for patient intake and discharge.
- Create wireframes for the user interface to ensure ease of use.

3. Finance Domain

Project Example: Financial Reporting Automation

Objective: Automate financial reporting to improve accuracy and reduce manual efforts in creating quarterly and annual reports.

Goals:

1. Reduce report preparation time by 50%.
2. Minimize errors in financial statements.
3. Enable real-time financial insights through dashboards.
4. Comply with IFRS and other regulatory standards.

BA Role:

- Elicit requirements from finance teams.
- Collaborate with data engineers to design ETL pipelines for data extraction.
- Define KPIs and create Power BI dashboards for reporting.

4. Insurance Domain

Project Example: Claims Processing System Enhancement

Objective: Enhance the existing claims processing system to improve efficiency, reduce processing time, and minimize fraudulent claims.



Goals:

1. Automate claim status tracking for customers.
2. Reduce claim settlement time by 20%.
3. Implement AI-driven fraud detection mechanisms.
4. Integrate the system with external partners like hospitals.

BA Role:

- Gather requirements from claims adjusters and underwriters.
- Create process flow diagrams for claims adjudication.
- Validate the AI model for fraud detection with real-world scenarios.

5. E-Commerce Domain

Project Example: Omnichannel Customer Experience Platform

Objective: Develop a platform to unify customer interactions across mobile, web, and in-store channels for a seamless experience.

Goals:

1. Increase customer retention by 25% through personalized offers.
2. Enable real-time inventory visibility across all channels.
3. Integrate chatbot support for customer queries.
4. Reduce cart abandonment rate by 15%.

BA Role:

- Gather requirements from marketing and customer service teams.
- Document workflows for order placement and returns.
- Define APIs for integration with inventory and payment systems.
- Oversee UAT for omnichannel features like cart synchronization.



6. Retail Domain

Project Example: Inventory Management System Implementation

Objective: Implement an inventory management system to improve stock tracking and reduce overstock and stockouts.

Goals:

1. Reduce inventory carrying costs by 20%.
2. Provide real-time stock visibility to store managers.
3. Automate inventory replenishment alerts.
4. Integrate the system with POS and supplier systems.

BA Role:

- Elicit requirements from store managers and suppliers.
- Create use cases for stock replenishment and order tracking.
- Test system integration with supplier APIs.

List Business Analyst Terminologies

Requirements Gathering

The process of collecting the needs and conditions from stakeholders to define the objectives of a project.

Example: For an eCommerce portal, this could involve gathering information on desired features such as product search, user registration, and payment options from potential users and stakeholders.



Stakeholder Analysis

Analyzing and understanding the needs and expectations of stakeholders involved in the project.

Example: In an eCommerce portal project, stakeholders could include customers, suppliers, the project team, and investors. Analyzing their needs might reveal the importance of including features like a supplier portal and investment tracking.

Business Case

A document that justifies the investment in a project, detailing the benefits, costs, risks, and opportunities.

Example: The business case for an eCommerce portal might include market analysis, competition, projected revenues, and the technological infrastructure required.

SWOT Analysis

A strategic tool for identifying strengths, weaknesses, opportunities, and threats in relation to business competition or project planning.

Example: For an eCommerce project, a SWOT analysis might highlight strengths such as a unique product offering, weaknesses like a lack of brand recognition, opportunities through market expansion, and threats from established competitors.

User Stories

Descriptions of software features from an end-user perspective.



Example: A user story for an eCommerce portal might be "As a shopper, I want to filter products by price range so that I can find items that fit my budget."

Workflow Analysis

The study of the flow of tasks and activities to accomplish a business process.

Example: Analyzing the order fulfillment process for an eCommerce portal to identify bottlenecks or inefficiencies.

Gap Analysis

Identifying the gap between the current state and the desired future state of a business or project.

Example: For an eCommerce portal, this could involve comparing current sales channels with the potential online market to identify growth opportunities.

Risk Management

Descriptions of software features from an end-user perspective

Example: In an eCommerce project, this might involve assessing risks such as data breaches or supply chain disruptions and implementing strategies to mitigate them.



Change Management:

The approach to transitioning individuals, teams, and organizations to a desired future state.

Example: Introducing a new eCommerce portal might require training for staff on new processes and managing the transition for existing customers.

Data Modeling

The process of creating a data model for the data to be stored in a database.

Example: Designing a database for an eCommerce portal that includes tables for users, products, orders, and transactions.

Process Improvement

The proactive task of identifying, analyzing, and improving upon existing business processes within an organization for optimization.

Example: Streamlining the checkout process on an eCommerce portal to reduce cart abandonment rates.

Business Intelligence

The use of data analysis in strategic decision making.

Example: Using sales data from an eCommerce portal to identify best-selling products and optimize inventory.



Feasibility Study

An assessment of the practicality of a proposed plan or method.

Example: Conducting a study to determine if integrating an advanced AI recommendation engine into the eCommerce portal is technically and financially viable.

Stakeholder Engagement

The process of involving stakeholders in the decision-making process by communicating and working with them.

Example: Regularly updating and seeking feedback from investors and potential users throughout the development of the eCommerce portal to ensure the project meets their expectations.

Cost-Benefit Analysis (CBA)

A systematic approach to estimate the strengths and weaknesses of alternatives.

Example: Assessing costs of high-availability infrastructure versus benefits like uptime and customer satisfaction.

Scope Management

The process of defining and controlling what is included and what is not included in a project.

Example: Defining launch features and future updates to avoid scope creep.



Quality Assurance (QA)

Ensuring that a product or service meets specified requirements and customer expectations.

Example: Implementing a rigorous testing phase for the eCommerce portal to identify and fix bugs before launch.

Project Charter

A document that officially starts a project and outlines its objectives, scope, and participants.

Example: Creating a project charter for the eCommerce portal that details the project's goals, key stakeholders, budget, and timeline.

Business Process Modeling (BPM)

The activity of representing processes of an enterprise so that the current process may be analyzed and improved.

Example: Modeling the order processing flow of the eCommerce portal to identify inefficiencies and areas for automation.

Functional Requirements

Specific behaviors or functions of a system.

Example: The eCommerce portal must support multiple payment methods, including credit cards, PayPal, and bank transfers.



Non-functional Requirements (NFRs)

Requirements that specify criteria that can be used to judge the operation of a system, rather than specific behaviors.

Example: The eCommerce portal should be able to handle up to 10,000 simultaneous users without performance degradation.

Prototyping

Creating a preliminary model of a product to test and validate ideas and assumptions.

Example: Developing a clickable prototype of the eCommerce portal's user interface to gather user feedback on its usability.

Agile Methodology

An iterative project management approach enabling faster, efficient value delivery.

Example: Using Scrum practices to manage the development of the eCommerce portal, allowing for rapid iterations based on stakeholder feedback.

Return on Investment (ROI)

A performance measure used to evaluate the efficiency or profitability of an investment.

Example: Calculating the ROI of the eCommerce portal project by comparing the costs of development and operation against the generated revenues and cost savings.



Minimum Viable Product (MVP)

A product version designed to gather maximum customer insights with minimal effort.

Example: Launching the eCommerce portal with just enough features to satisfy early customers and provide feedback for future product development.

Key Performance Indicators (KPIs)

Quantifiable measures to assess performance against strategic and operational goals.

Example: Monitoring KPIs such as conversion rate, average order value, and customer acquisition cost to assess the performance of the eCommerce portal.

Customer Journey Mapping

The process of creating a visual story of your customers' interactions with your brand.

Example: Mapping out the steps a customer takes from discovering the eCommerce portal to making a purchase, to identify opportunities for improving the customer experience.

Agile Methodology

An iterative approach to project management for faster, streamlined value delivery.



Example: Using Scrum practices to manage the development of the eCommerce portal, allowing for rapid iterations based on stakeholder feedback.

FREQUENTLY ASKED QUESTIONS IN BUSINESS ANALYST INTERVIEWS

These questions and answers showcase both technical knowledge and real-world applications, which are critical in Business Analyst interviews. Here's a list of commonly asked technical interview questions for Business Analysts, covering SQL, JIRA, and various documentation tools, along with real-time answers

Explain the difference between WHERE and HAVING in SQL ?

WHERE is used to filter rows before grouping is performed.

HAVING is used to filter groups after grouping has been applied.

For example, in a sales dataset, if I want to find all sales where the amount is greater than 100, I'd use **WHERE**. If I want to find groups of products with total sales exceeding 500, I'd use **HAVING** after grouping by product.

What is the difference between UNION and UNION ALL in SQL?

UNION combines the result sets of two queries and removes duplicate rows.

UNION ALL combines the result sets but includes all duplicates.



In practice, I use UNION when I need distinct records and UNION ALL when performance is critical, as removing duplicates is resource-intensive.

What are stored procedures, and why are they used?

A **stored procedure** is a precompiled collection of SQL statements stored in the database. It helps in reusability, performance optimization, and security.

For example, I've used stored procedures to handle complex business logic, such as calculating monthly sales commissions, which improves maintainability and execution speed.

How do you manage requirements in JIRA?

In JIRA, I create **epics** for high-level requirements and break them down into **stories and tasks**.

I use custom fields, workflows, and dashboards to track the progress of each requirement. For instance, during a recent project, I mapped user requirements to JIRA stories, assigned them to sprints, and tracked the completion status through burndown charts.

What is a BRD (Business Requirements Document)?

A BRD outlines the high-level business needs and objectives for a project. It focuses on what the business wants to achieve without detailing technical solutions.

In my last project, the BRD included sections like project goals, scope, key stakeholders, and functional requirements. For instance, a requirement might state, "The system must allow users to reset passwords via email."



How is an FRD different from a BRD?

An FRD translates the high-level business requirements from the BRD into detailed functional specifications. It focuses on **how** the system should work.

For example, while the BRD states, “The system must generate sales reports,” the FRD specifies, “The report must include sales by region, filtered by date range, and exported in PDF format.”

What is the purpose of an SRS (Software Requirements Specification)?

An SRS provides a detailed description of the software to be developed, covering both functional and non-functional requirements.

I recently created an SRS for a CRM project. It included functional requirements (e.g., user login, data import), non-functional requirements (e.g., system uptime of 99.9%), and use case diagrams.

What is FSD (Functional Specification Document), & how do you use it?

An FSD is similar to an FRD but is more detailed, often used for guiding the development team. It includes workflows, screen designs, and detailed functionality descriptions.

I’ve used FSDs to communicate the exact behavior of a feature, such as “The payment gateway must support Visa, MasterCard, and PayPal, with error messages displayed for failed transactions.”

How do you leverage Excel for Business Analysis?

I use Excel for data analysis, leveraging tools like pivot tables and VLOOKUP to identify trends, such as customer churn.



How do you use MS Visio in your role as a Business Analyst?

I use MS Visio to create process flows, organizational charts, and system architecture diagrams. It helps visualize complex workflows and communicate effectively with stakeholders.

For example, I once mapped out a customer onboarding process, including touchpoints, system interactions, and decision points, which clarified bottlenecks for the team.

Can you explain the difference between a BRD and FRD?

A BRD outlines business goals and requirements, while an FRD details how the system will meet them. For example, a BRD may state the need for online payment integration, while an FRD specifies support for credit cards, PayPal, and wallets.

How do you create and manage tasks in JIRA?

In JIRA, I create tasks by defining a clear title, description, and assigning them to team members. I use labels, priorities, and dependencies to organize tasks. For instance, during a sprint, I created user stories for a CRM project, linked them to epics, and tracked their status using Kanban boards to ensure timely delivery.

What are the key components of an SRS ?

Introduction: Project purpose and scope.

System Features: Functional and non-functional requirements.

Use Cases: User interactions.

Assumptions and Constraints: Project limitations (e.g., banking proj-



-ect, the SRS detailed login features, transaction limits, and security protocols."

What is an FSD, and how is it different from an FRD?

An FSD (Functional Specification Document) provides a detailed, step-by-step explanation of how a system will implement the functionalities outlined in the FRD. The FRD defines 'what' the system should do, while the FSD explains 'how' it will do it. For example, in a healthcare project, the FRD mentioned patient appointment scheduling, and the FSD detailed the database schema and API workflows for this feature.

How do you write and organize user stories in JIRA?

I follow the INVEST principle (Independent, Negotiable, Valuable, Estimable, Small, Testable) while writing user stories. For example, in a logistics project, I wrote a story: 'As a warehouse manager, I want to view low-stock items in real-time so that I can reorder inventory promptly.' I included acceptance criteria and linked it to relevant tasks and epics in JIRA.

Last reminder?

Embrace curiosity and perseverance, view mistakes as lessons, and trust that consistent effort will pave the way to success.

Follow Us



Recruit Riders Technologies 



(+1) 302-240-0042

*All the best for
your future career*