BUSINESS ANALYSIS

Creating by

AMRAPALI FULE

CRIATING BY

AMRAPALI FULE

Contents

INTRODUCTIO………………………………………………………………………………………………………………………………..3

[Who is business analyst? 5](#_Toc188717279)

[What does a business analyst do? 5](#_Toc188717280)

[What's the difference between a business analyst and a data analyst? 5](#_Toc188717281)

[Software Development Life cycle (SDLC) 6](#_Toc188717282)

[Post SDLC Process 7](#_Toc188717283)

[Role of Business Analyst during SDLC Process 7](#_Toc188717284)

[What is the Agile methodology? 8](#_Toc188717285)

[What is Kanban? 12](#_Toc188717286)

[Scrum vs. Kanban 13](#_Toc188717287)

[What is a product backlog? 13](#_Toc188717288)

[What are agile user stories? 14](#_Toc188717289)

[JIRA 17](#_Toc188717290)

[Conclusion 18](#_Toc188717291)

[References 18](#_Toc188717292)

# Introduction

## What is Business Analysis

Business analysis is a combination of gaining insight from data using specific techniques, and performing tasks to identify the needs of a business—then, recommending changes and providing solutions that produce value for the stakeholders. Many of the solutions potentially have software and digital data-based components, but can also incorporate organizational changes, like improving processes, developing new policies, and engaging in strategic planning.

**Objectives of Business Analysis**

* Improve business efficiency and effectiveness.
* Identify and address business challenges.
* Align business strategies with technological solutions.
* Enhance stakeholder satisfaction through value-driven solutions.

**Techniques in Business Analysis**

Business analysts use a variety of techniques to ensure a thorough understanding of business needs and the development of effective solutions. Here are some commonly used techniques:

1. **SWOT Analysis**

* Identifies the organization's Strengths, Weaknesses, Opportunities, and Threats.
* Helps in strategic planning and identifying areas for improvement.

1. **Stakeholder Analysis**

* Identifies stakeholders, their interests, and influence levels.
* Ensures effective communication and prioritization of stakeholder needs.

1. **Business Process Modelling**

* Visual representation of workflows and processes using tools like flowcharts or BPMN (Business Process Model and Notation).
* Highlights inefficiencies and areas for optimization.

1. **Use Case Modelling**

* Defines how users interact with a system to achieve specific goals.
* Clarifies system requirements and user expectations.

1. **Gap Analysis**

* Compares current performance with desired outcomes.
* Identifies gaps and suggests ways to bridge them.

1. **Root Cause Analysis (RCA)**

* Identifies the underlying causes of problems.
* Ensures solutions address the real issues.

1. **Requirements Traceability Matrix (RTM)**

* Links requirements to design, development, and testing processes.
* Ensures all requirements are addressed and accounted for.

1. **MoSCoW Prioritization**

* Categorizes requirements into Must-have, Should-have, Could-have, and Won’t-have.
* Helps in prioritizing features or tasks.

1. **Prototyping**

* Develops a preliminary version of a product or system.
* Helps stakeholders visualize and refine solutions.

1. **Benchmarking**

* Compares business performance with industry standards or competitors.
* Identifies best practices and performance gaps.

## Who is business analyst?

A business analyst is a professional who plays a crucial role in identifying business needs, analysing processes, and recommending solutions to improve efficiency, productivity, and profitability within an organization. They bridge the gap between business stakeholders and IT teams by gathering and translating requirements into actionable plans, often using data analysis and modelling techniques. Business analysts help businesses make informed decisions and implement changes that align with their strategic goals. A business analyst is a professional who works within an organization to identify and implement improvements

## What does a business analyst do?

* **Identify needs**: Business analysts analyse data to identify business needs and problems
* **Recommend solutions**: Business analysts recommend solutions that maximize value for stakeholders
* **Implement solutions**: Business analysts implement solutions that can include software development, process improvements, or organizational changes
* **Communicate**: Business analysts communicate with stakeholders to provide information about the business and how it plans to achieve its goals

## What's the difference between a business analyst and a data analyst?

Both data analysts and business analysts support data-driven decisions in their companies. Business analysts tend to focus more on recommending solutions for business needs, while data analysts work more closely with the data.

# Software Development Life cycle (SDLC)

Software Development Life Cycle (SDLC) is a process followed in a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. It defines a methodology for improving the quality of software and the overall development process.

SDLC is a process used by IT analysts in order to develop or redesign high quality software system, which meets both the customer and the real-world requirement.

It takes into consideration all the associated aspects of software testing, analysis and post-process maintenance.

The important phases of SDLC are depicted in the following illustration –



SDLC FLOW chart

**Requirement Analysis/ planing stage**

Every activity must start with a plan. Failing to plan is planning to fail. The degree of planning differs from one model to another, but it's very important to have a clear understanding of what we are going to build by creating the system's specifications.

Every activity must start with a plan. Failing to plan is planning to fail. The degree of planning differs from one model to another, but it's very important to have a clear understanding of what we are going to build by creating the system's specifications.

**System Design**

**S**ecound step is all analysis planing to create a design format most In system design, the design functions and operations are described in detail, including screen layouts, business rules, process diagrams and other documentation. The output of this stage will describe the new system as a collection of modules or subsystems.

**Implementation stage**

Implementation is a part of the Building Stage. In this phase, we start code generation based on the system's design using compilers, interpreters, debuggers to bring the system to life.

**Testing stage**

As different parts of the system are completed; they are put through a series of tests. it is tested against the requirements to make sure that the product is actually solving the needs addressed during the requirement phase.

**Diployment stage**

After the test phase ends, the system is released and enters the production environment. Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometime product deployment happens in stages as per the organization’s business strategy.

**Maintenance stage**

elicit, analyze, specify, and validate the maintenance-related business needs of stakeholders, be they customers or end users of applications.

## Post SDLC Process

After the product is released in the market, its maintenance is done for the existing customer base.

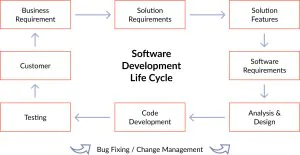
Once in the production environment, the system will suffer modifications because of undetected bugs or other unexpected events. The system is evaluated and the cycle is repeated for maintaining the system.

## 

## Role of Business Analyst during SDLC Process

As we can see the below diagram, BA is involved in driving business requirement and converting them to solution requirements.

He is involved in translating the solution features into software requirements. Then leads in analysis and designing phase, dictates in code development, then follows the testing phase during bug fixing as a change agent in the project team and ultimately fulfills the customer requirements.



# 

# What is the Agile methodology?

The Agile method is a game-changer in project management and software development. It has reshaped the landscape of business analysis. Embracing Agile principles, businesses have seen a marked increase in project success rates.

The Agile methodology has revolutionized how businesses approach project management and software development. Rooted in the Agile Manifesto, Agile emphasizes flexibility, continuous improvement, and high customer involvement.

It contrasts with traditional linear approaches like the Waterfall Model. The Waterfall Model is more rigid and sequential, however, Agile is more significant because it is adaptable and it allows teams to respond to changes. This makes Agile particularly suited to projects where requirements are likely to evolve.

Traditionally, a BA’s role involved in-depth planning. It also focused on gathering detailed requirements documentation and setting a clear-cut project scope.

Within Agile projects, the BA is pivotal in ensuring the team delivers value to the business. They are no longer scribes or conveyors of information. Instead, BAs actively engage in decision-making processes and often bridge the technical and business worlds.

The Agile BA is critical in spanning the gap between business objectives and the Agile team’s work. Using Agile, they confirm that development efforts align with business goals. This enhances the value delivered through Agile projects as they now play an active role in the project’s success.

**Key tasks, roles, and responsibilities of an Agile BA**

In the Agile framework, the role of a BA is multifaceted. It often includes adapting to the dynamic demands of Agile projects. Their responsibilities and tasks revolve around several core areas:

1. Understanding stakeholders and final signoff processes

2. Analyzing functional and non-functional requirements

3. Scope of projects and risk analysis

4. Iterative development and facilitating processes

**Role in product backlog management**

The Agile BA is crucial in managing the product backlog, a key component in Agile project management. Their responsibilities in this area, coordinated with the provided keywords, include:

1. Stocking and grooming the product backlog

* Collecting and defining requirements. Agile BAs gather requirements from various stakeholders and convert them into backlog items. They verify these items are clearly defined and align with the business needs and project objectives.
* Backlog grooming. Regularly, they refine and update the backlog, corroborating that it accurately reflects the current project constraints, priorities, and changes in the business environment.
* Example. In a project for a new mobile app, an Agile BA might continuously add and refine features in the backlog, such as user authentication, payment integration, and user feedback mechanisms.

2. Creating user stories and prioritizing features

* User story development. Agile BAs craft user stories that encapsulate the requirements in an understandable and actionable format for the development team.
* Prioritization. They prioritize these stories based upon business value, customer impact, and technical feasibility.
* Example. For an e-commerce website upgrade, an Agile BA might prioritize user stories related to enhancing the checkout process over those stories adding new product categories based on customer feedback and business impact analysis.

3. Collaboration with product owners and other stakeholders

* Facilitating collaboration. Agile BAs facilitate effective communication and collaboration between the product owner, the development team, and other stakeholders.
* Bridging gaps. They play a vital role in bridging gaps between the business side (product owner and stakeholders) and the technical side (Agile team), ensuring a mutual understanding and disposition of goals.
* Example. In a software development project, an Agile BA might organize and lead meetings between the product owner and technical team in order to discuss and resolve product vision and technical implementation discrepancies.

By managing the product backlog decisively. Agile BAs monitor that the Agile team works on the most valuable features that align with business objectives and stakeholder expectations. Their role in backlog management is integral to the Agile process, promising a smooth flow of work and maximizing the project’s success.

**Integrating Business Analysis into Agile processes**

Incorporating business analysis into Agile processes is a vital function of Agile BAs, who assert that the analytical activities seamlessly align with Agile principles and foster a flexible iterative approach.

* Alignment with Agile principles. Agile BAs integrate business analysis by lining up activities with Agile principles, safeguarding a versatile and intermittent approach to requirements.
* Continuous collaboration. They work closely with Agile teams throughout the project, maintaining ongoing collaboration and adaptation to change.
* Case. In an Agile project, BAs might integrate business analysis into daily stand-ups and sprint planning sessions to continuously match the business needs with development activities.

Through continuous teamwork and alignment with Agile principles, Agile BAs enhance the efficacy of Agile teams, ensuring that business analysis is not just an activity, but an integral part of the Agile process.

**Planning for multiple iterations of detailed requirements**

Agile BAs are pivotal in planning and adapting requirements through multiple iterations, a process central to the Agile methodology’s focus on delivering continuous value.

* Iterative requirement refinement. Agile BAs plan for and manage the evolution of requirements over multiple iterations, guaranteeing that each sprint delivers value and connects with the overall project goals.
* Backlog prioritization. They prioritize the product backlog to reflect changing business needs and stakeholder feedback.
* Case. Based on user feedback and testing results, the BA would iteratively refine user stories and acceptance criteria across sprints for a new software feature.

Agile BAs’ iterative refinement of requirements and competent backlog prioritization double check that each sprint is purposeful and closely aligned with the evolving business necessities and stakeholder feedback.

**Supporting the technology team during sprints**

In Agile teams, the role of BAs extends to providing crucial support to the technology team, facilitating an extensive knowledge of project requirements, and aiding in problem-solving.

* Facilitating understanding. Agile BAs support the technology team by clarifying requirements, answering questions, and ensuring a shared sense of the project goals.
* Problem-solving. They assist in resolving any ambiguities or issues during sprints as a bridge between technical and business perspectives.
* Case. During a sprint focused on developing a new API, the BA would work closely with developers to assure that they fully realize the business context and user needs.

By engaging with the technology team during sprints, Agile BAs play a critical role in spanning the gulf between business perspectives and technical execution, verifying project goals are met with clarity and efficiency.

The Business Analysis Plan in Agile is a dynamic blueprint that guides BAs in effectively integrating their expertise into Agile projects. It underlines the importance of iterative planning, continuous stakeholder collaboration, and proactive support for the technology team, making certain that Agile projects are well-aligned with business objectives and adapt to changing conditions.

## What is Kanban?

Kanban is a popular framework used to implement Agile and DevOps software development. It requires real-time communication of capacity and full transparency of work. Work items are represented visually on a kanban board, allowing team members to see the state of every piece of work at any time.

Optimizing software development with Kanban flow

Kanban flow, a cornerstone of agile and DevOps methodologies, drives efficiency by orchestrating seamless task progression through visualized workflows. Kanban flow mirrors the streamlined inventory management of supermarkets, ensuring tasks move through development processes precisely when needed.

Visualized on Kanban boards, tasks represented as cards enable transparent progress tracking and swift identification of bottlenecks. By limiting work-in-progress (WIP), teams optimize resource allocation and maintain a steady workflow. Kanban's focus on continuous improvement is facilitated by metrics like control charts and cumulative flow diagrams, empowering teams to refine workflows iteratively.

**Kanban boards**

The work of all Kanban teams revolves around a Kanban board, a tool used to visualize and optimize the workflow across teams. While physical boards are popular among some teams, virtual boards are crucial in any agile software development tool for their traceability, collaboration, and accessibility from multiple locations.

Regardless of whether a team uses a digital or physical kanban board, it ensures that the team visualizes their work, standardizes their workflow, and immediately identifies and resolves all blockers and dependencies. A basic kanban board has a three-step workflow: To Do, In Progress, and Done. However, depending on a team's size, structure, and objectives, they can map the workflow to meet their unique processes. Because the kanban methodology relies upon full transparency of work and real-time communication, the kanban board acts as the single source of truth for the team's work.

**Kanban cards**

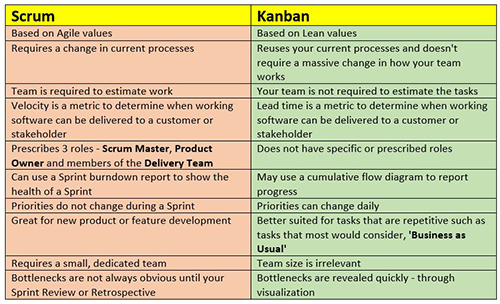
The main purpose of representing work as a card on the Kanban board is to allow team members to track progress through its workflow in a highly visual manner.

Kanban cards feature critical information about project tasks, giving teams visibility into who is responsible for which tasks and a brief description of the job, and how long tasks are estimated to take. Cards on virtual kanban boards often feature screenshots and other technical details that are valuable to the assignee.

Enabling team members to view the status of each task at any moment, alongside relevant details, promotes heightened focus, comprehensive traceability, and rapid identification of blockers and dependencies.

### Scrum vs. Kanban

Kanban and scrum share some of the same concepts but have very different approaches. They should not be confused with one another.



Some teams blend the ideals of the kanban method and scrum into "scrumban." They take fixed-length sprints and roles from Scrum and focus on work-in-progress limits and cycle time from Kanban.

For teams just starting with agile, we strongly recommend choosing one methodology and running with it for a while. If your team is ready to use the kanban methodology, use our free kanban board template today!

## What is a product backlog?

A product backlog is a prioritized list of work for the development team that is derived from the product roadmap and its requirements. The most important items are shown at the top of the product backlog so the team knows what to deliver first. The development team doesn't work through the backlog at the product owner's pace and the product owner isn't pushing work to the development team. Instead, the development team pulls work from the product backlog as there is capacity for it, either continually (kanban) or by iteration (scrum).

**How to effectively manage a product backlog**

Once the product backlog is built, it's important to regularly maintain it to keep pace with the program. Product owners should review the backlog before each iteration planning meeting to ensure prioritization is correct and feedback from the last iteration has been incorporated. Regular review of the backlog is often called "backlog grooming" in agile circles (some use the term backlog refinement).

Once the backlog gets larger, product owners need to group the backlog into near-term and long-term items. Near-term items need to be fully fleshed out before they are labeled as such. This means complete user stories have been drawn up, collaboration with design and development has been sorted out, and estimates from development have been made. Longer term items can remain a bit vague, though it's a good idea to get a rough estimate from the development team to help prioritize them. The key word here is "rough": estimates will change once the team fully understands and begins work on those longer-term items.

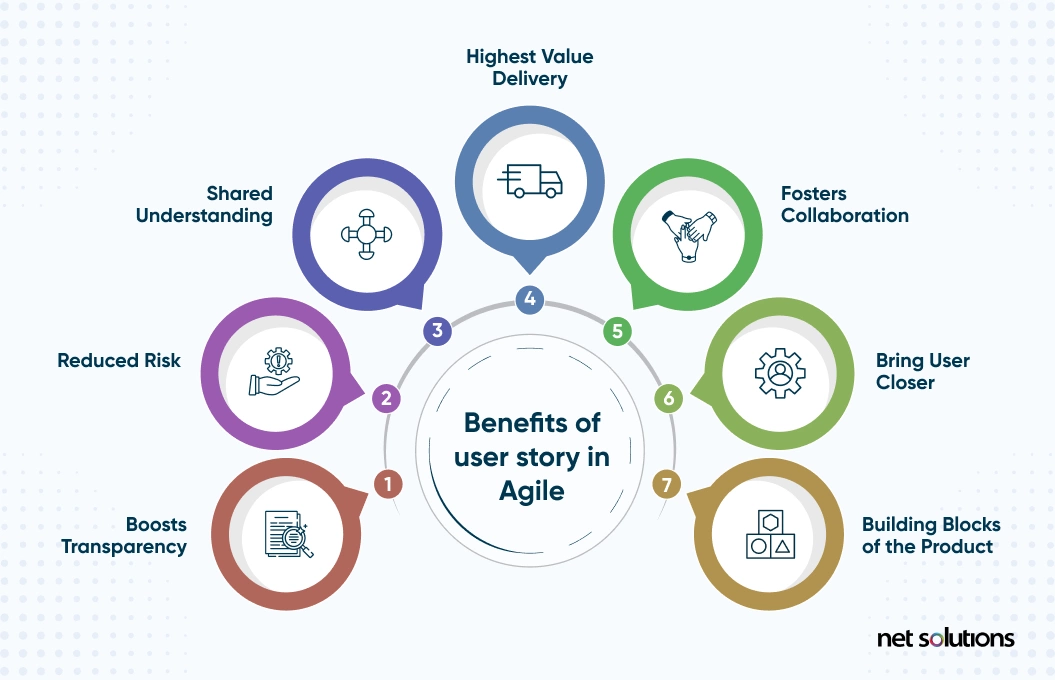
The backlog serves as the connection between the product owner and the development team. The product owner is free to re-prioritize work in the backlog at any time due to customer feedback, refining estimates, and new requirements. Once work is in progress, though, keep changes to a minimum as they disrupt the development team and affect focus, flow, and morale.

## What are agile user stories?

A user story is the smallest unit of work in an agile framework. It’s an end goal, not a feature, expressed from the software user’s perspective. It is an informal, general explanation of a software feature written from the perspective of the end user or customer.

The purpose of a user story is to articulate how a piece of work will deliver a particular value back to the customer. Note that "customers" don't have to be external end users in the traditional sense, they can also be internal customers or colleagues within your organization who depend on your team.

User stories are also the building blocks of larger agile frameworks like epics and initiatives. Epics are large work items broken down into a set of stories, and multiple epics comprise an initiative. These larger structures ensure that the day-to-day work of the development team (on stores) contributes to the organizational goals built into epics and initiatives.



**Why create user stories?**

For development teams new to agile, user stories sometimes seem like an added step. Why not just break the big project into a series of steps and get on with it? But stories give the team important context and associate tasks with the value those tasks bring.

User stories serve a number of key benefits:

* **Stories keep the focus on the user.** A to-do list keeps the team focused on tasks that need to be checked off, but a collection of stories keeps the team focused on solving problems for real users.
* **Stories enable collaboration.** With the end goal defined, the team can work together to decide how best to serve the user and meet that goal.
* **Stories drive creative solutions.** Stories encourage the team to think critically and creatively about how to best solve for an end goal.
* **Stories create momentum.** With each passing story, the development team enjoys a small challenge and a small win, driving momentum.

**Working with user stories**

Once a story has been written, it’s time to integrate it into your workflow. Generally a story is written by the product owner, product manager, or program manager and submitted for review.

During a sprint or iteration planning meeting, the team decides what stories they’ll tackle that sprint. Teams now discuss the requirements and functionality that each user story requires. This is an opportunity to get technical and creative in the team’s implementation of the story. Once agreed upon, these requirements are added to the story.

Another common step in this meeting is to score the stories based on their complexity or time to completion. Teams use t-shirt sizes, the Fibonacci sequence, or planning poker to make proper estimations. A story should be sized to complete in one sprint, so as the team specs each story, they make sure to break up stories that will go over that completion horizon.

**How to write user stories**

Consider the following when writing user stories:

* **Definition of “done”**: The story is generally “done” when the user can complete the outlined task, but make sure to define what that is.
* **Outline subtasks or tasks**: Decide which specific steps need to be completed and who is responsible for each of them.
* **User personas**: For whom? If there are multiple end users, consider making multiple stories.
* **Ordered Steps**: Write a story for each step in a larger process.
* **Listen to feedback**: Talk to your users and capture the problem or need in their words. No need to guess at stories when you can source them from your customers.
* **Time**: Time is a touchy subject. Many development teams avoid discussions of time altogether, relying instead on their estimation frameworks. Since stories should be completable in one sprint, stories that might take weeks or months to complete should be broken up into smaller stories or should be considered their own epic.

Once the user stories are clearly defined, make sure they are visible for the entire team.

**Getting started with agile user stories**

User stories describe the why and the what behind the day-to-day work of development team members, often expressed as “*persona + need + purpose”.* Understanding their role as the source of truth for what your team is delivering, but also why, is key to a smooth process.

Start by evaluating the next, or most pressing, large project (e.g. an epic). Break it down into smaller user stories, and work with the development team for refinement. Once your stories are out in the wild where the whole team can see them, you’re ready to get to work.

# JIRA

JIRA, developed by Atlassian, is a popular and versatile project management tool that has gained widespread adoption across industries. Originally designed for software development, JIRA's flexibility and customizable features make it an excellent choice for Business Analysts to manage projects effectively. JIRA's robust issue tracking, customizable workflows, collaboration features, and integration capabilities make it an invaluable asset for managing projects effectively.

**Issue Tracking & Management:**

With JIRA's intuitive interface, you can easily track the status of each issue, ensuring nothing falls through the cracks.

**Customizable Workflows:**

JIRA allows you to create custom workflows that reflect your organization's processes and methodologies. As a Business Analyst, you can define stages such as 'To Do,' 'In Progress,' 'Review,' and 'Done.' These workflows streamline the project's progress and keep everyone on the same page.

**Collaboration and Communication:**

Effective communication is vital for successful project delivery. JIRA provides a centralized platform where team members, stakeholders, and clients can collaborate, share ideas, and exchange feedback. Its commenting and notification features ensure that everyone is aware of updates, changes, or any important discussions related to the project.

**Requirements Management:**

As a Business Analyst, managing project requirements is one of your core responsibilities. JIRA's 'Epics' and 'User Stories' functionalities allow you to break down complex requirements into smaller, manageable chunks.

**Dashboards and Reporting:**

Visual representation of project data is essential for quick decision-making. JIRA offers customizable dashboards that allow Business Analysts to create reports, charts, and graphs based on various metrics such as project progress, team performance, and issue status.

**Integration Capabilities:**

JIRA's compatibility with a wide array of third-party tools and plugins enhances its functionality even further. You can integrate it with tools like Confluence (for documentation), Bitbucket (for version control), and various test management and automation tools.

**Agile Project Management:**

For Business Analysts working in Agile environments, JIRA is an ideal choice. It supports Agile methodologies like Scrum and Kanban, allowing you to plan sprints, track user stories, and manage backlogs effortlessly. Its adaptability to changing requirements and continuous feedback loops align perfectly with Agile principles.

# Conclusion

Business analysis is a vital function for organizations aiming to adapt and thrive in dynamic markets. By applying a variety of techniques, business analysts can identify opportunities, solve problems, and drive value. Their work ensures that businesses not only meet their objectives but also remain competitive and resilient in the face of change.

# References

<https://www.techtarget.com/searchbusinessanalytics/definition/business-analytics-BA>

<https://www.tutorialspoint.com/business_analysis/business_analysis_software_development_life_cycle.htm>

<https://techdocs.broadcom.com/us/en/ca-enterprise-software/valueops/rally/rally-help/learning-agile/the-agile-primer.html>

<https://www.tutorialspoint.com/jira/jira_project.htm>

<https://www.simplilearn.com/agile-business-analyst-article>