

Introduction to Business Analysis

Who is a Business Analyst?

Business Analysts have emerged to have a key role in recent business scenarios. Some people think that the role of a Business Analyst is to make money for the organization, which may not be true in direct context. But indirectly, the action and decision taken by Business Analysts do leave an impact on the financial prospects of the organization.

What does a Business Analyst Do?

A primary job responsibility of Business Analyst is to communicate with all stakeholders & to elicit, analyze and validate the requirements for changes to business processes, information systems, and policies.

A professional business analyst plays a big role in moving an organization toward efficiency, productivity, and profitability.

Before we jump into the tutorial, we will see some basic perspective of a Business Analyst to help the organization succeed. The foremost priority for any business analyst will be to try understanding following things

- Understand what business does and how it does
- Determine how to improve existing business processes
- Identify the steps or tasks to support the implementation of new features
- Design the new features to implement
- Analyze the impact of implementing new features
- Implement the new features

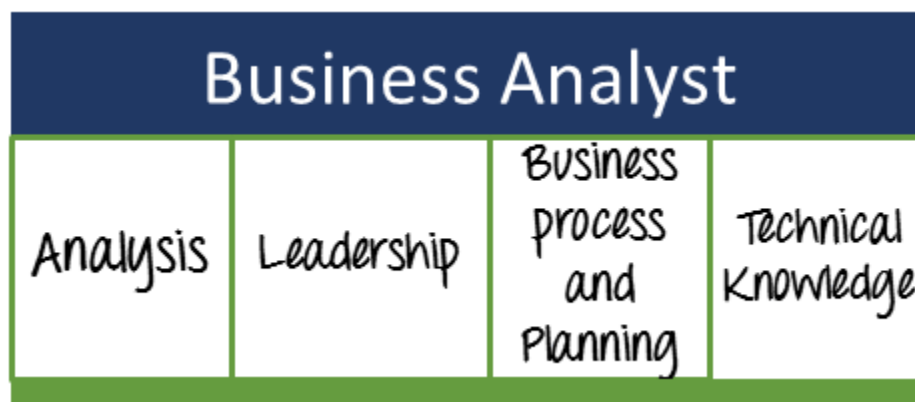
Different Business Analyst Role

Business Analyst can be from any sector, and the role differs based on the sector. Business Analyst are classified into various categories like

- Business Analyst
- Business Process Analyst
- IT Business Analyst
- Business System Analyst
- System Analyst
- Data Analyst
- Functional Architect
- Usability or UX Analyst

Characteristics of a good Business Analyst

Basically, a good business analyst is judged on these four attributes



Typical Qualities of a Good Business Analyst:

- **Analytical skills**- An outstanding analytical skill will separate out a good business analyst. A good part of BA role includes analyzing data, workflow, user or stakeholders' inputs, documents, etc.
- **Leadership skills**- directing team members, forecasting budget, helping team members with the problem, etc.
- **Business process and planning**- Planning the project scope, understanding and implementing requirement of project, identifying resources required for the project and so on
- **Technical skill**- If a business analyst is in the IT sector, few technical aspects are expected to know like operating systems, hardware capabilities, database concepts, networking, SDLC methodology, etc.

Tools of the Trade

To make their work easier, the business analyst often depends on tools like

- Top Team Analyst: This tool helps in providing a complete solution for requirements gathering and management
- Smart Draw: It is a graphic diagramming tool that uses stencils for organizational charts, swim lanes, data flow diagram, etc.
- Blueprint: This tool produces a blue print of the project artifacts like development models, test scenarios, use cases, flow charts, etc. to ensure that everything is falling in line and as per expectation
- Survey Monkey: It allows you to send survey to stakeholders, capture their feedback, rank and prioritize their view and turn them into requirements

There are many other tools like [iServer](#), [Meetingsense](#), [Ravenflow](#), [AnalystPro](#), which could be used by Business Analyst during the project.

Certifications

As per the International Institute of Business Analysis (IIBA), CBAP (Certified Business Analysis Professional) certification is a recognized certificate for a professional Business Analyst. They provide two types of certifications. The certification exam is computer based and consists of multiple-choice questions.

- Certification of Competency in Business Analysis: Pre-requisite for this certification is at least 3750 hours of work experience
- Certified Business Analysis Profession (senior level): Pre-requisite for this certification is at least 7500 hours of work experience

Jobs

Job prospectus for Business Analyst rises every year, especially for the IT sector. The average salary of business analyst is estimated around \$80,000 - \$130,000, even at entry level.

International Institute of Business Analysis (IIBA) is growing exponentially indicating increasing demand of Business Analyst. Business Analyst always remain

an organization priority since they have to work in a close proximity to top executives, clients, and stakeholders.

According to U.S Bureau of Labor Statistics, the BA job are predicted to increase by 19% between 2012 and 2022.

Conclusion:

The business analyst role is promising and has to deal with different layers of an organization. Business analyst are classified into various categories like Business Process Analyst, IT Business Analyst and so on.

- A good business analyst should encompass skills like
 - Analytical skills
 - Leadership skills
 - Business process and Planning
 - Technical Skills
- Various tools that can help Business Analyst are Top Team Analyst, Smart Draw, Blueprint, etc.
- Online certification course for BA available by recognized institute IIBA
- According to U.S Bureau of Labor Statistics, the BA job are predicted to increase by 19% between 2012 and 2022.

Stakeholder Analysis & Mapping with Template Example

Stakeholder Analysis and Mapping is done to map the interest of your stakeholders. It is a process of systematically analyzing and gathering qualitative information to determine whose interest should be taken into account.

Stakeholder Analysis and Mapping is important because it helps project leaders and managers to access a stakeholder's interest, positions, alliances and knowledge related to the project.

When Stakeholder Analysis need to be done?

Stakeholder analysis should always be done at the beginning of a project. Such analysis is helpful in the drafting of a log frame. Log frame is nothing but a general approach to project planning, monitoring, and evaluation in the form of a 'log frame matrix'. Whenever log frames are reconsidered during the life cycle of a project, a stakeholder analysis will be useful. Which means whenever mid-term reviews or annual monitoring is handled, stakeholder analysis should be the part of it.

Stakeholders Categorization

Stakeholders are categorized into two categories

Internal stakeholders	External stakeholders
Within the organization: Employees and Management	Outside the organization: Government & Trade Association

Process for Stakeholder Analysis

Following are the primary aspect needs to be considered for stakeholder analysis

Step 1) Identify your stakeholders: Your boss, your team, senior executives, prospective customers, your family, etc.

Step 2) Assess how those stakeholders could be impacted or have an effect on the organization

Step 3) Prioritize your Stakeholders-

Stakeholder Type	Action
<ul style="list-style-type: none">• High power, interested people	- Manage closely
<ul style="list-style-type: none">• High power, less interested people	- Keep satisfied
<ul style="list-style-type: none">• Low power, interested people	- Keep informed
<ul style="list-style-type: none">• Low power, less interested people	- Monitor with minimum effort

Step 4) Identify areas of conflicts (organization vs. stakeholder, stakeholder vs. stakeholder)

Step 5) Prioritize, reconcile and balance stakeholders

Step 6) Align significant stakeholder needs with organizations strategies and actions

Things to take care while dealing with stakeholders

- Could you eliminate processes, which do not add stakeholder value?
- How would you communicate with stakeholders?
- Do your communications encourage stakeholder exchange?
- Do you communicate the stakeholder the value of the deal?

Important questions to ask during Stakeholder Analysis Mapping

Different attribute check for stakeholder	Question to ask your stakeholders
<ul style="list-style-type: none">• Identification of stakeholder	<ul style="list-style-type: none">• Who is paying for the project?• Who will receive the deliverables or profits from the project?• Both from your organization and client organization who will work with you to implement the project?• Identify the expert for the project domain in the organization.
<ul style="list-style-type: none">• Interest	<ul style="list-style-type: none">• What direct benefit do stakeholders expect to get from the project?• What outcomes do stakeholders expect as a result of the project?• What changes do stakeholders need to make as a result of the project?• Are there any conflicts of interest amongst the stakeholders?
<ul style="list-style-type: none">• Influence	<ul style="list-style-type: none">• What legitimate authority do stakeholders have in the organization?• Who controls the project assets and resources?• What degree of influence or negotiation power do your identified stakeholders carry in the organization?
<ul style="list-style-type: none">• Impact	<ul style="list-style-type: none">• How much impact stakeholder could have on the project and does this going to affect the success of the project

Also, you need to figure out when stakeholders will become involved in the following-

- Project Vision
- Project Scope Definition
- Business Process Analysis
- Needs Elicitation
- Requirement Validation
- Design reviews

- User Acceptance Testing

You can create a "**Participation Matrix Table**" for the stakeholders as given below

Participation Type	Inform	Consult	Partner	Control
Needs Assessment				
Planning				
Implement				
Monitoring & Evaluation				

Tips to manage your Stakeholders

- Do not complain. Accept stakeholders as they are
- For guaranteed success, get the key leadership involved.
- Make sure, you involve your stakeholders early in the business analysis process
- In case of a sensitive issue, ensure full confidentiality to all stakeholders to win their trust.
- To avoid conflicts, help all stakeholders in realizing their personal gains from the project.
- Stakeholders mapping and analysis always helps.

Requirement Life Cycle: Validation, Documentation & Management

Requirement lifecycle involves a number of phases and at times it can be a complicated process. The nature of the process depends on the methodology you choose for your software development like Agile, Waterfall, Incremental, etc. Each phase may involve a lot of paperwork and approval procedure. It also deals with the project documents like a project proposal, project management plan, project scope, and the business case. Let see some of the common requirement lifecycle required to know for a Business Analyst.



Requirement Definition

It is one of the primary phases of the requirement gathering process commonly known as Requirement extraction.

Once the requirement is gathered, it can be organized in folders logically as per product release or sprint.

These requirements are analyzed further to prepare facts and figures for a business analyst to track possible result based on analysis. This procedure is referred as **Impact Assessment**.

Requirement Validation

The requirement validation phase includes analyzing the needs or conditions required to meet a new or altered product considering needs of the various stakeholders.

For the success of any project, validating requirements is very important. Requirement validation includes checking the specification, wireframe, High Fidelity Simulation, Traceability Analysis, etc.

There are requirement validation tools that do validation with very less human intervention.

Requirement Documentation

Requirement documents should cover following things

- Project stakeholder's requirement
- Business analysis plan
- Current state analysis
- Scope statement specification

Requirement Management

Requirement Management process includes planning, monitoring, analyzing, communicating and managing of those requirements. If the requirement is not managed well, the end product will get affected adversely. There is requirement management tool available online which help you to manage the requirement with minimum hurdles.

Software Requirements Analysis with Example

Software requirement is a functional or non-functional need to be implemented in the system. Functional means providing particular service to the user.

For example, in context to banking application the functional requirement will be when customer selects "View Balance" they must be able to look at their latest account balance.

Software requirement can also be a non-functional, it can be a performance requirement. For example, a non-functional requirement is where every page of the system should be visible to the users within 5 seconds.

So, basically **software requirement is a**

- Functional or
- Non-functional

need that has to be implemented into the system. **Software requirement are usually expressed as a statement.**

In this tutorial we will learn,

- Types of Requirements
- Other Sources of Requirements
- How to Analyze Requirements
- Atomic
- Uniquely Identified
- Complete
- Consistent and Unambiguous
- Traceable
- Prioritized
- Testable
- Conclusion

Types of Requirements

1. **Business requirements:** They are high-level requirements that are taken from the business case from the projects.

For example, a mobile banking service system provides banking services to Southeast Asia. The business requirement that is decided for India is account summary and fund transfer while for China account summary and bill payment is decided as a business requirement

Country	Company providing Banking Functionalities or services
India	Account Summary and Fund Transfer
China	Account Summary and Bill Payment

2. **Architectural and Design requirements:** These requirements are more detailed than business requirements. It determines the overall design required to implement the business requirement.

For our educational organization the architectural and design use cases would be login, course detail, etc. The requirement would be as shown below.

Banking use case	Requirement
Bill Payment	<p>This use case describes how a customer can login into net banking and use the Bill Payment Facility.</p> <p>The customer will can see a dashboard of outstanding bills of registered billers. He can add, modify, and delete a biller detail. The customer can configure SMS, email alerts for different billing actions. He can see history of past paid bills.</p>

The actors starting this use case are bank customers or support personnel.

3. **System and Integration requirements:** At the lowest level, we have system and integration requirements. It is detailed description of each and every requirement. It can be in form of user stories which is really describing everyday business language. The requirements are in abundant details so that developers can begin coding.

Here in example of Bill Payment module where requirement will be mentioned for adding a Biller

Bill Payment	Requirements
Add Billers	<ul style="list-style-type: none">• Utility Provider Name• Relationship/Customer Number• Auto Payments – Yes/No• Pay Entire Bill – Yes/No• Auto Payment Limit – Do not pay if Bill is over specified amount

Sometimes for some project you might not receive any requirements or documents to work with. But still there are other sources of requirements that you can consider for the requirement or information, so that you can base your software or test design on these requirements. So the other sources for requirement you can rely on are

Other Sources of Requirements

- Knowledge transfer from colleagues or employees already working on that project
- Talk about project to business analyst, product manager, project lead and developers
- Analyze previous system version that is already implemented into the system
- Analyze the older requirement document of the project

- Look into the past Bug reports, some of the bug reports are turned into enhancement request which may be implemented into current version
- Look into installation guide if it is available to see what are the installation required
- Analyze the domain or industry knowledge that team is trying to implement

Whatever source of requirement you get make sure to document them in some form, get them reviewed from other experienced and knowledgeable team members.

How to Analyze Requirements

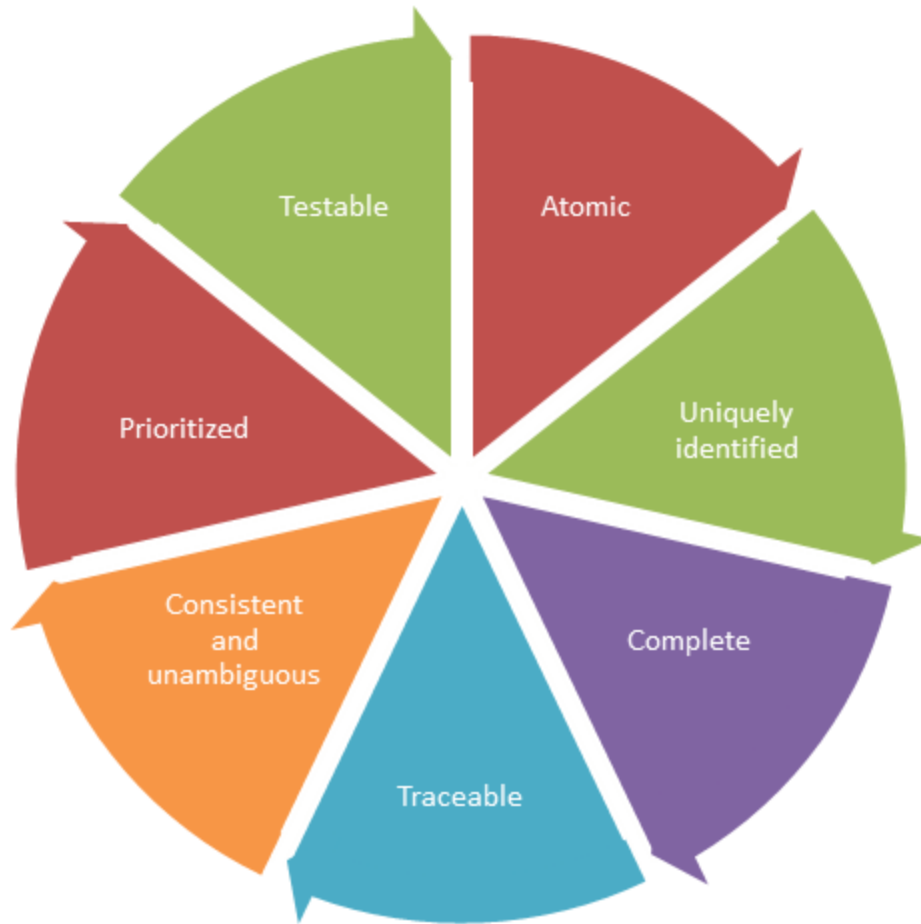
Consider example of an educational software system where a student can register for different courses.

Let's study how to analyze the requirements. The requirements must maintain a standard quality of its requirement, different types of requirement quality include

- Atomic
- Uniquely identified
- Complete
- Consistent and unambiguous
- Traceable
- Prioritized
- Testable

	Example of bad requirement	Example of good requirement
Requirement Quality		
Atomic	<ul style="list-style-type: none"> • Students will be able to enroll to undergraduate and post graduate courses 	<ul style="list-style-type: none"> • Students will be able to enroll to undergraduate courses • Students will be able to enroll to post-graduate courses

Uniquely identified	1- Students will be able to enroll to undergraduate courses 1- Students will be able to enroll to post-graduate courses	1. Course Enrolment 2. Students will be able to enroll to undergraduate courses 3. Students will be able to enroll to post-graduate courses
Complete	A professor user will log into the system by providing his username, password, and other relevant information	A professor user will log into the system by providing his username, password and department code
Consistent and unambiguous	A student will have either undergraduate courses or post-graduate courses but not both. Some courses will be open to both under-graduate and post-graduate	A student will have either under-graduate or post graduates but not both
Traceable	Maintain student information-mapped to BRD req.ID?	Maintain student information-Mapped to BRD req ID 4.1
Prioritized	Registered student-Priority 1 Maintain User Information-Priority 1 Enroll courses-Priority 1 View Report Card-Priority 1	Register Student-Priority 1 Maintain User Information-Priority 2 Enroll courses-Priority 1 View Report Card-Priority 3
Testable	Each page of the system will load in an acceptable time-frame	Register student and enrol courses pages of the system will load within 5 seconds



Let understand this with an example, there are three columns in the table shown here,

1. The first column indicates- "requirement quality"
2. The second column indicates- "bad requirement with some problem"
3. The third column is same as second column but – "converted into a good requirement".

Now let's understand each of these requirements in details starting with Atomic.

Atomic

Atomic	<ul style="list-style-type: none">• Students will be able to enroll to undergraduate and post graduate courses	<ul style="list-style-type: none">• Students will be able to enroll to undergraduate courses• Students will be able to enroll to post-graduate courses
--------	------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

So, each and every requirement you have should be atomic, which means it should be at very low level of details it should not be possible to separated out into components. Here we will see the two examples for requirements, at Atomic and uniquely identified requirements levels.

So, let us continue with example of system build for education domain. Here, the bad requirement is "Students will be able to enroll to undergraduate and post graduate courses" . This is a bad requirement because it is not atomic because it talks about two different entities undergraduates and post-graduates' courses. So obviously it is not a good requirement but bad requirement, so correspondence good requirement would be to separate it out into two requirements. So, one talks about the enrolment to undergraduate courses while the other talks about the enrolment to the post-graduate courses.

Uniquely Identified

Uniquely identified	<ul style="list-style-type: none">1- Students will be able to enroll to undergraduate courses1- Students will be able to enroll to post-graduate courses	<ul style="list-style-type: none">1- Course Enrolment1.1- Students will be able to enroll to undergraduate courses1.2-Students will be able to enroll to post-graduate courses
---------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Similarly, the next requirement quality is to check for uniquely identified, here we have two separate requirements but they both have same ID#1. So, if we are referring our requirement with reference to ID#, but it is not clear which exact requirement we are referring to document or other part of the system as both have same ID#1. So, separating out with unique id's, so good requirement will be re-

return as section 1- course enrolments, and it has two requirements 1.1 id is enrolment to undergraduate courses while 1.2 id is enrolment to postgraduate courses.

Complete

Complete	A professor user will log into the system by providing his username, password, and other relevant information	A professor user will log into the system by providing his username, password and department code
----------	---------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------

Also, each and every requirement should be complete. For example, here the bad requirement says a "professor user will log into the system by providing his username, password and other relevant information". Here the other relevant information is not clear, so the other relevant information should be spelt out in good requirement to make the requirement complete.

Consistent and Unambiguous

Consistent and unambiguous	A student will have either undergraduate courses or post-graduate courses but not both. Some courses will be open to both under-graduate and post-graduate	A student will have either under-graduate or post graduates but not both
----------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------

Next each and every requirement should be consistent and unambiguous, so here for instance we have requirements "A student will have either undergraduate courses or post-graduate courses but not both" this is one requirement there is

some other requirement that says "Some courses will be open to both under-graduate and post-graduate students".

The problem in this requirement is that from the first requirement it seems that the courses are divided into two categories under graduate courses and post graduate courses and student can opt either of two but not both. But when you read other requirement it conflicts with the first requirement and it tells that some courses will open to both post-graduate and under-graduate.

So it is obvious to convert this bad requirement into good requirement which is "A student will have either under-graduate courses or post-graduate courses but not both". Which means that every course will be marked either being as under-graduate course or post-graduate course

Traceable

Each and every requirement should be traceable because there are already different levels of requirement, we already saw that at the top we had business requirements, and then we have an architectural and design requirement followed by system integration requirements.

Now when we convert business requirement into architectural and design requirements or we convert architectural and design requirements to system integration requirements there has to be traceability. Which means that we should be able to take each and every business requirement and map it to the corresponding one or more software architectural and design requirement. So here is an example of bad requirement that says "Maintain student information – mapped to BRD req ID?" the requirement id is not given over here.

So converting it to a good requirement it says same thing but it is mapped with the requirement id 4.1. So mapping should be there for each and every requirement. Same way we have high level and low-level mapping requirement, the mapping is also there between system and integration requirement to the code that implements that requirement and also there is a mapping between the system and integration requirement to the test case which test that particular requirement.

So this traceability is all across entire project

Prioritized

Prioritized	Registered student-Priority 1 Maintain User Information-Priority 1 Enroll courses-Priority 1 View Report Card-Priority 1	Register Student-Priority 1 Maintain User Information-Priority 2 Enroll courses-Priority 1 View Report Card-Priority3
-------------	-----------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------

Then each and every requirement must be prioritized, so the team has guideline so which requirement that able to implement first and which can be done later on. Here you can see the bad priority has register student, maintain user information and each and every requirement has given priority-1. Everything cannot be at same priority, so requirement can be prioritized. So the example of good requirement over here is the register student and enroll courses is given the highest priority 1, while maintain user information comes below at priority 2 and then we have view report card at priority-3

Testable

Testable	Each page of the system will load in an acceptable time-frame	Register student and enroll courses pages of the system will load within 5 seconds
----------	---------------------------------------------------------------	------------------------------------------------------------------------------------

Each and every requirement should be testable, here the bad requirement is "each page of the system will load in an acceptable time frame". Now there are two problems with this requirement first is that each page meaning that there can be many pages, which going to blow up the testing efforts. The other problem is that it say the page is going to load in acceptable time frame, now what is acceptable time frame? Acceptable to whom. So we have to convert the non-testable argument into a testable argument, which specifically tells about which page we are talking about "register student and enroll courses pages" and the acceptable time frame is also given which is 5 seconds.

Conclusion

So this is how we have to look at each and every requirement at appropriate level. For example, if we are going to build a software with regards to system and integration requirements. We have to look in system and integration requirements given in the software requirement specifications or user stories and apply to each and every requirement quality. Then check whether each and every requirement is atomic, uniquely identified, and complete and so on.

Requirements Analysis Techniques with Example:

As a Business Analyst, requirement analysis is the most important part of your Job. It will **help you determining the actual needs of stakeholders**. At the same time, enable you to communicate with the stakeholders in a language they understand (like charts, models, flow-charts,) instead of complex text.

A requirement analysis has a

- Specific Goal
- Specific Input
- Specific Output
- Uses resources
- Has a number of activities to be performed in some order
- May affect more than one organization unit
- Creates value of some kind for the customer

Requirement Analysis Techniques

Requirement analysis techniques are mainly used to map the business workflow so that you can analyze, understand and make required changes to that workflow or process.

There are various requirement analyzing techniques that can be used as per the software development process like

1. Business process modeling notation (BPMN)

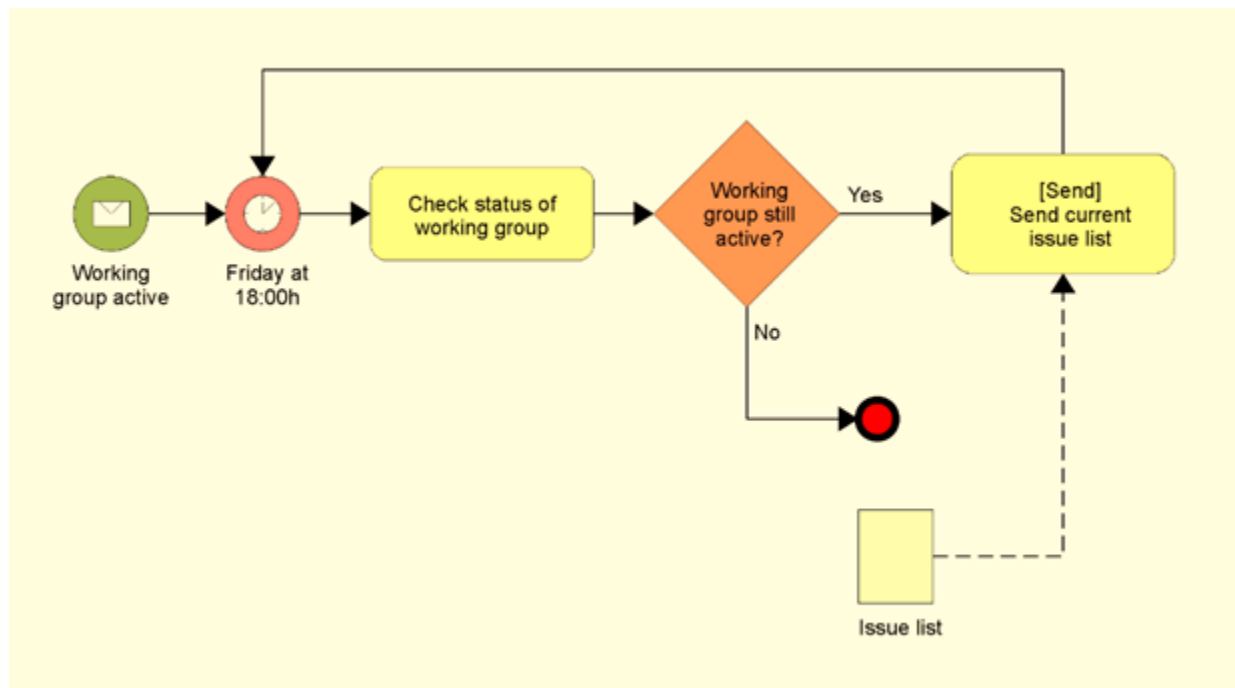
BPMN (Business Process Modeling & Notation) is a graphical representation of your business process using simple objects, which helps the organization to communicate in a standard manner. Various objects used in BPMN includes

- Flow objects
- Connecting objects
- Swim lanes
- Artifacts.

A well design BPMN model should be able to give the detail about the activities carried out during the process like,

- Who is performing these activities?
- What data elements are required for these activities?

The biggest benefit of using BPMN is that it is easier to share, and most modeling tools support BPMN.



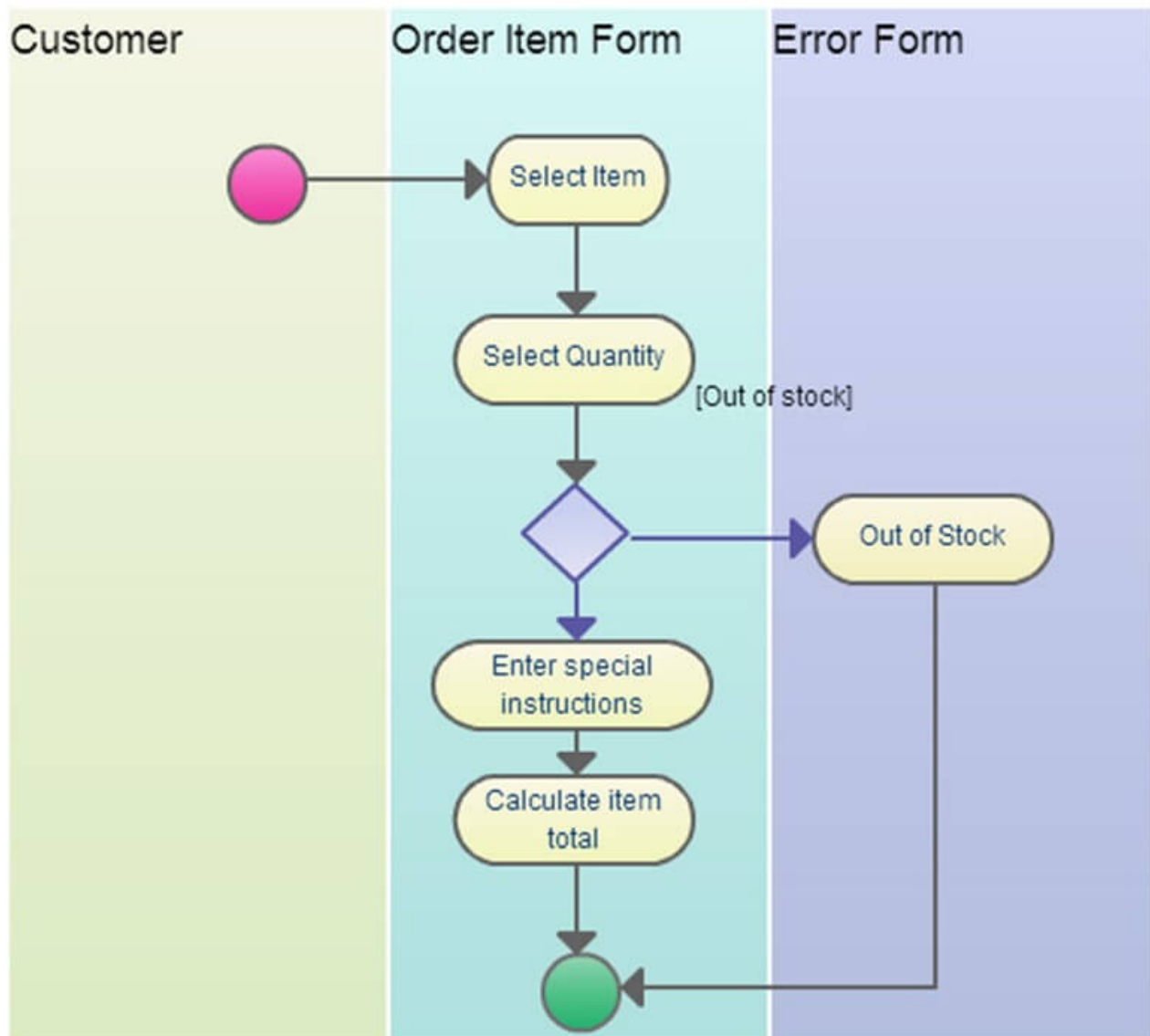
2. UML (Unified Modeling Language)

UML is a modelling standard primarily used for specification, development, visualization and documenting of software system. To capture important business process and artifacts UML provides objects like

- State
- Object
- Activity
- Class diagram

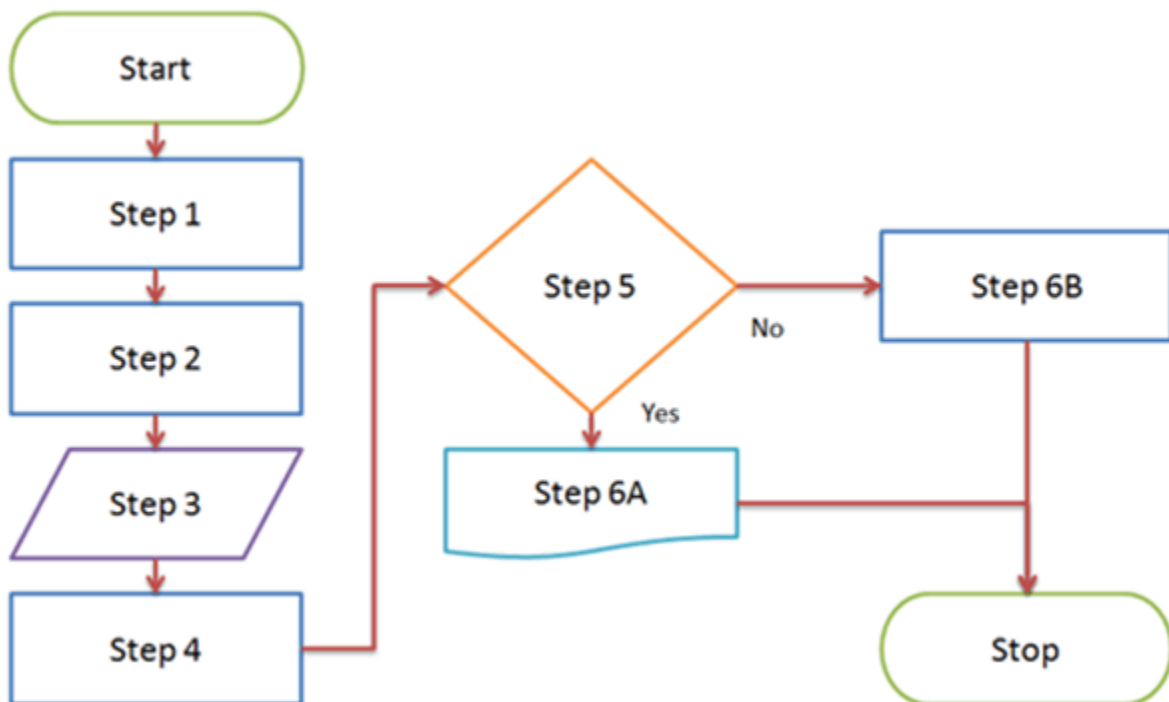
There are 14 UML diagrams that help with modelling like the use case diagram, interaction diagram, class diagram, component diagram, sequence diagram, etc.

UML models are important in the IT segment as it becomes the medium of communication between all stakeholders. A UML-based business model can be a direct input to a requirements tool. A UML diagram can be of two type's Behavioral model and Structural model. A behavioral model tries to give information about what the system do while a structural model will give what is the system consist of.



3. Flow chart technique

A flowchart is a visual representation of the sequential flow and control logic of a set of related activities or actions. There are different formats for flowcharts which include Linear, Top-down and cross-functional (swim lanes). A flow chart can be used for different activities like representing data flows, system interactions, etc. The advantage of using Flowchart is that it can be easy to read and write even for non-technical team members, and can show the parallel process by function, critical attributes of a process, etc.

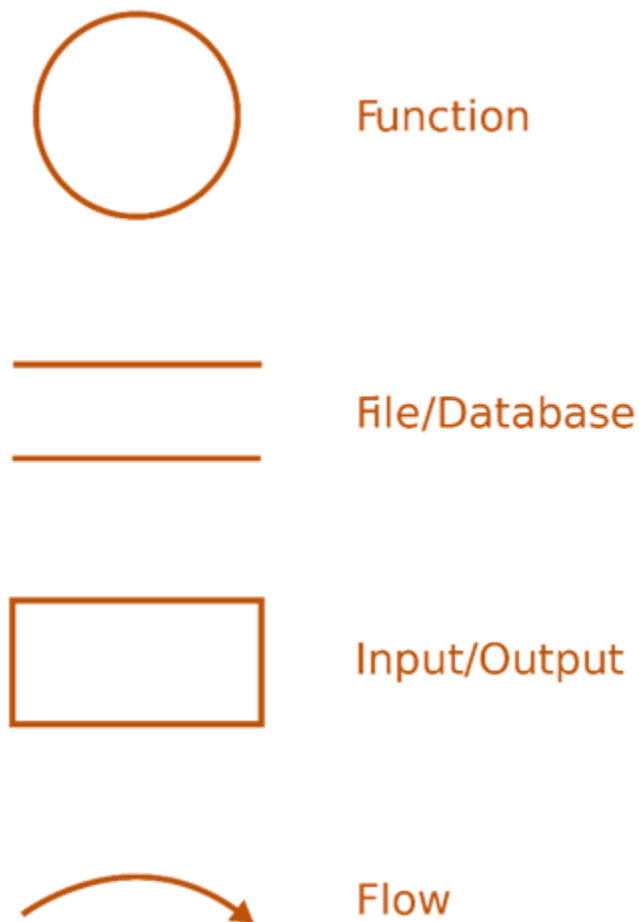


4. Data flow diagram

Data flow diagrams show how data is processed by a system in terms of inputs and outputs. Components of data flow diagram includes

- Process
- Flow
- Store
- Terminator

A logical data flow diagram shows system's activities while a physical data flow diagram shows a system's infrastructure. A data flow diagram can be designed early in the requirement elicitation process of the analysis phase within the SDLC (System Development Life Cycle) to define the project scope. For easy analyzing a data flow diagram can be drilled down into its sub-processes known as "levelled DFD".

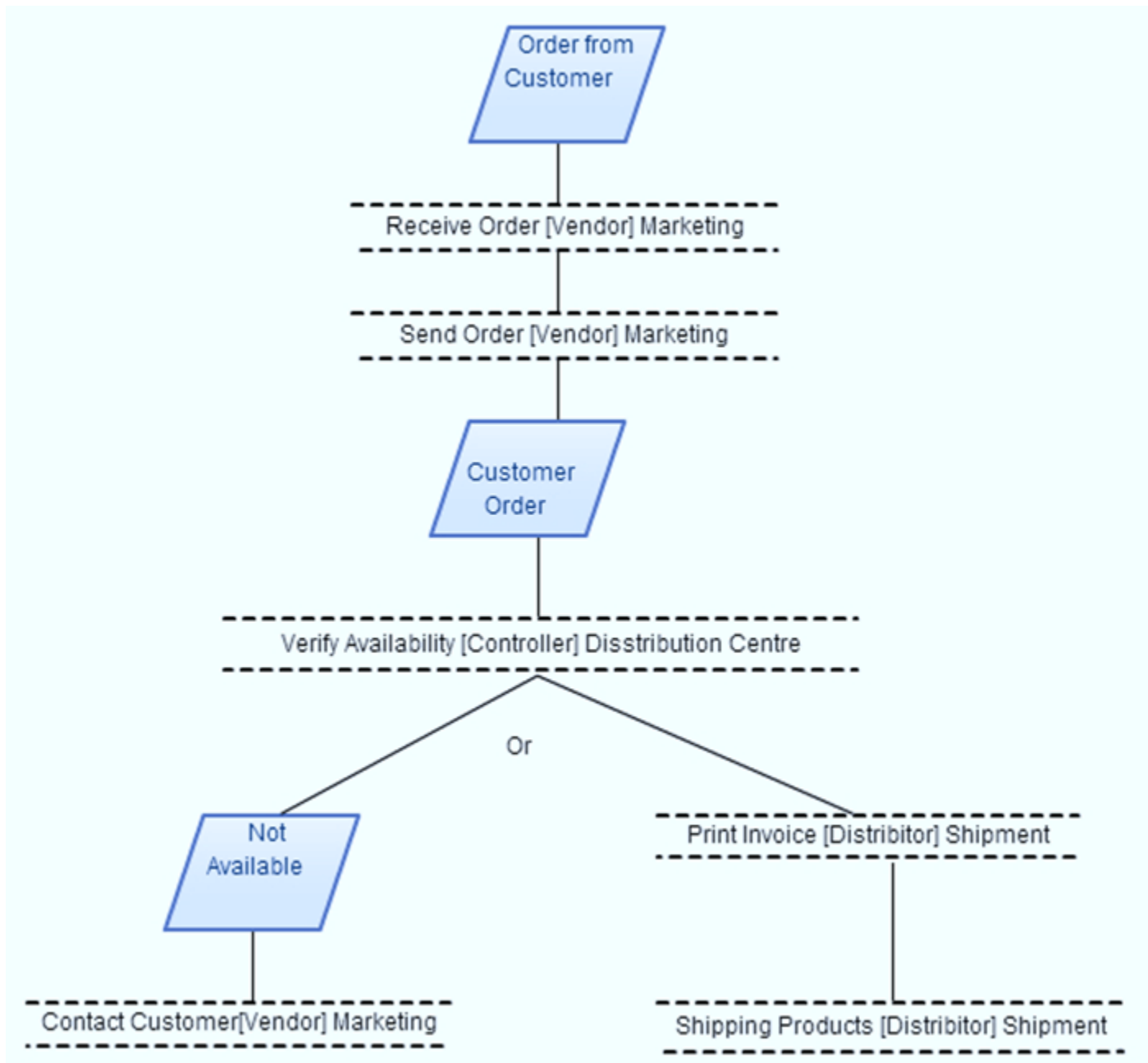


5. Role Activity Diagrams- (RAD)

Role activity diagram is similar to flowchart type notation. In Role Activity Diagram, role instances are process participants, which has start and end state. RAD requires a deep knowledge of process or organization to identify roles. The components of RAD includes

- Activities
- External events

- States



Roles group activities together into units of responsibility, according to the set of responsibility they are carrying out. An activity can be carried out in isolation with a role, or it may require coordination with activities in other roles.

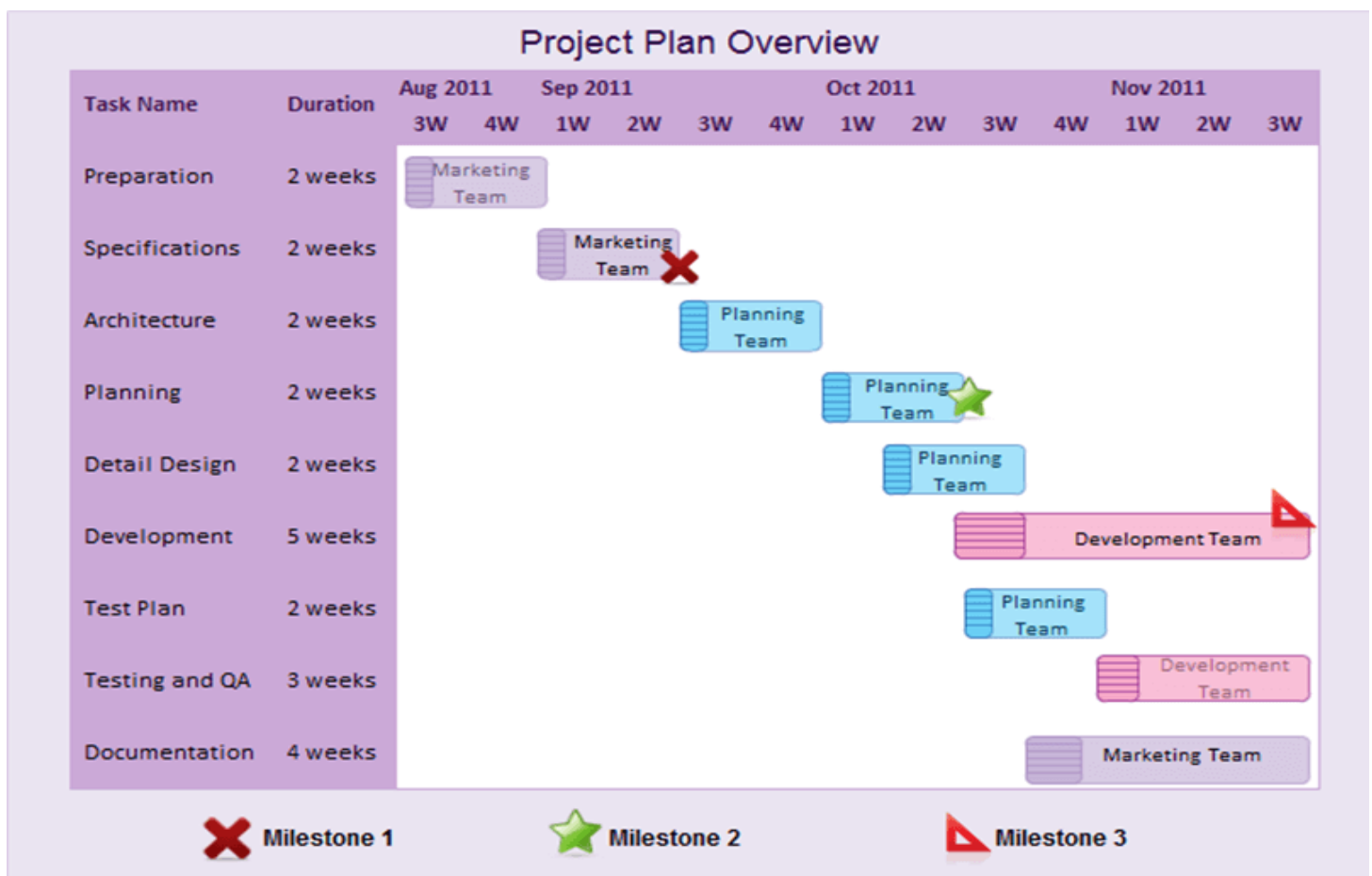
External events are the points at which state changes occur.

States are useful to map activities of a role as it progresses from state to state. When a particular state is reached, it indicates that a certain goal has been achieved.

RAD is helpful in supporting communication as it is easy to read and present a detailed view of the process and permitting activities in parallel.

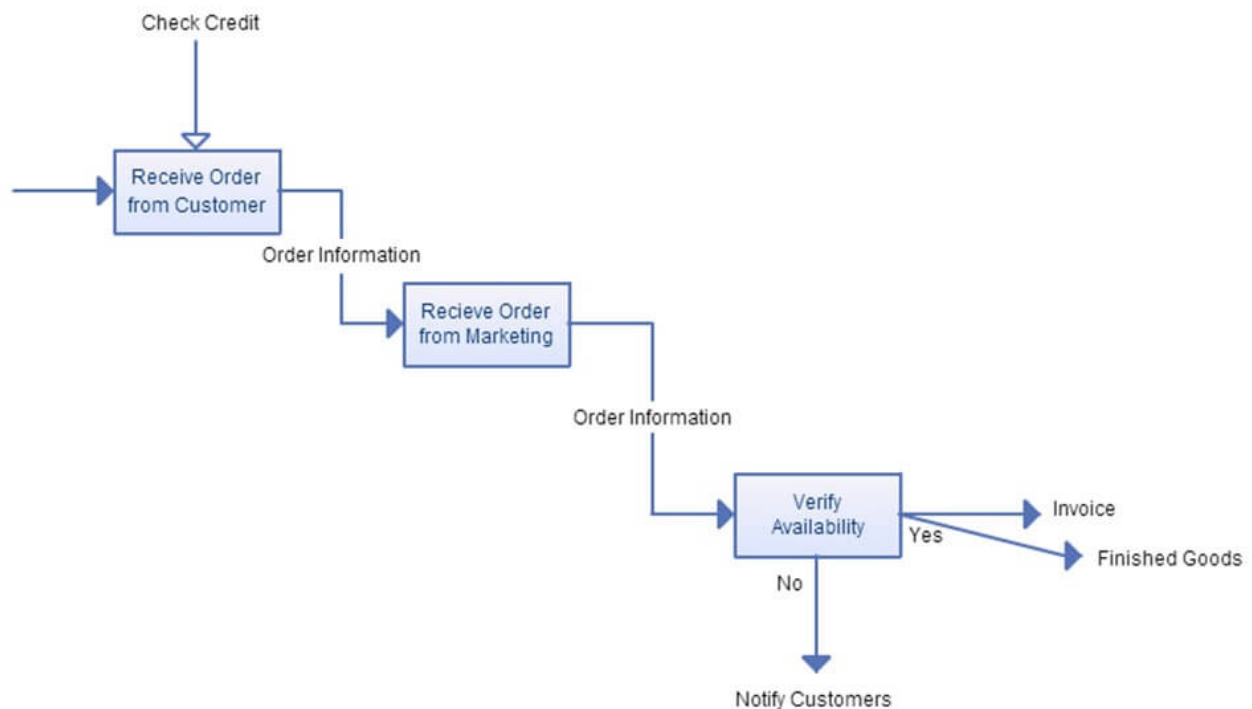
6. Gantt Charts

A Gantt chart is a graphical representation of a schedule that helps to coordinate, plan and track specific tasks in a project. It represents the total time span of the object, broken down into increments. A Gantt chart represents the list of all task to be performed on the vertical axis while, on the horizontal axis, it list the estimate activity duration or the name of the person allocated to the activity. One chart can demonstrate many activities.



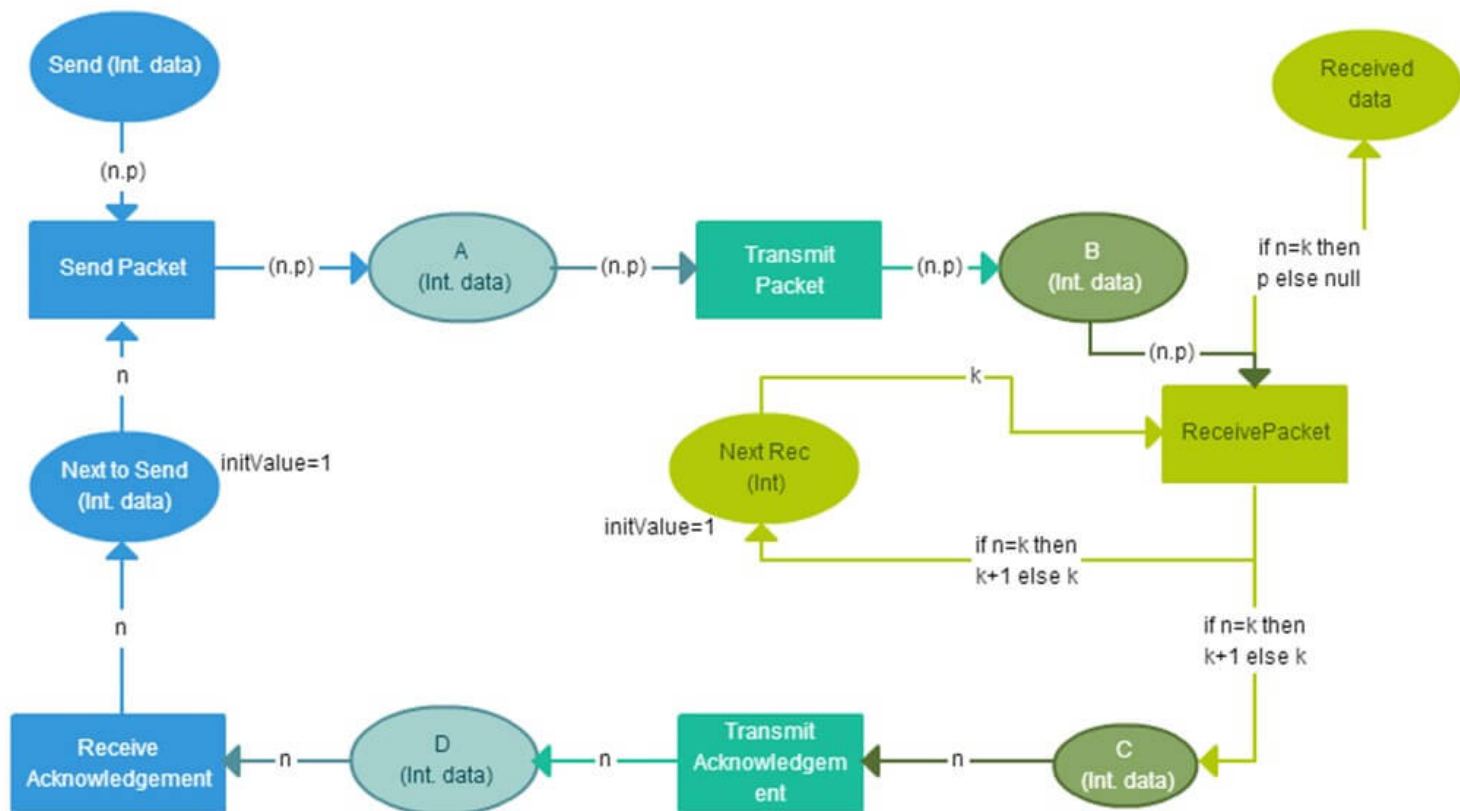
7. IDEF (Integrated Definition for Function Modeling)

IDEF or Integrated Definition for Function Modeling is a common name referred to classes of enterprise modeling languages. It is used for modeling activities necessary to support system analysis, design or integration. There are about 16 methods for IDEF, the most useful versions of IDEF are IDEF3 and IDEF0.



Petri nets objects have specific inscription like for

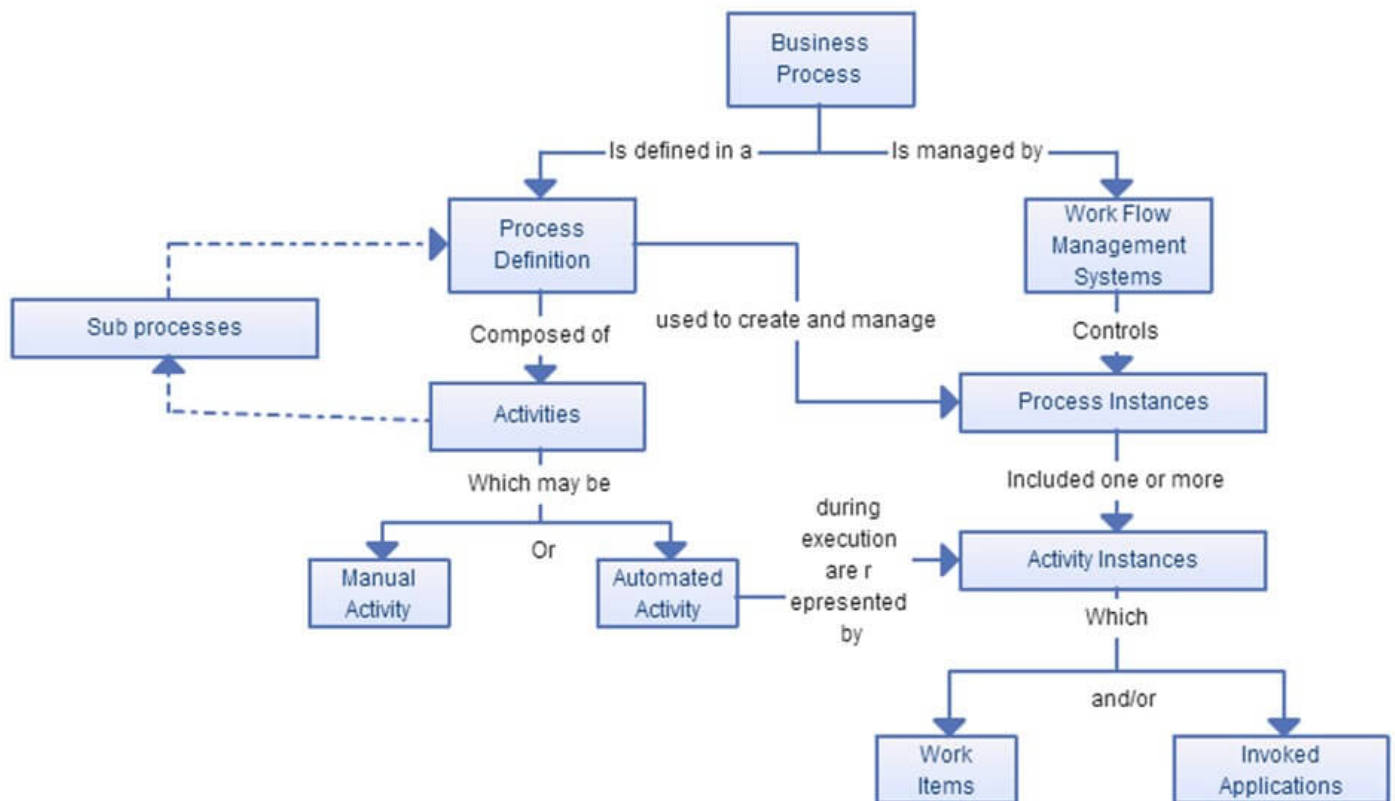
- **Places:** It has inscription like .Name, .Color Set, .Initial marking etc. While
- **Transition :** It has inscription like .Name (for identification) and .Guard (Boolean expression consist of some of the variables)
- **Arcs:** It has inscription like .Arc. When the arc expression is assessed, it yields multi-set of token colors.



9. Workflow Technique

Workflow technique is a visual diagram that represent one or more business processes to clarify understanding of the process or to make process improvement recommendations. Just like other diagrams like flowcharting, UML activity and process map, the workflow technique is the oldest and popular technique. It is even used by BA for taking notes during requirements elicitation. The process comprises of four stages

- Information Gathering
- Workflow Modeling
- Business process Modeling
- Implementation, Verification & Execution



10. Object oriented methods

Object-oriented modeling method uses object-oriented paradigm and modeling language for designing a system. It emphasis on finding and describing the object in the problem domain. The purpose of object-oriented method is

- To help characterizing the system
- To know what are the different relevant objects
- How do they relate to each other
- How to specify or model a problem to create effective design
- To analyze requirements and their implications

This method is applicable to the system which has dynamic requirements (changes frequently). It is a process of deriving use cases, activity flow, and events flow for the system. Object oriented analysis can be done through textual needs, communication with system stakeholder and vision document.

The object has a state, and state changes are represented by behavior. So, when the object receives a message, state changes through behavior.

11. Gap Analysis

Gap Analysis is the technique used to determine the difference between the proposed state and current state for any business and its functionalities. It answers questions like what is the current state of the project? Where do we want to be? etc. Various stages of Gap Analysis include

- Review System
- Development Requirements
- Comparison
- Implications
- Recommendations

How to organize Requirements as a Business Analyst

A business requirement is a formal document that addresses the need of the stakeholders for the project or product. There is no standard format to present the business requirement. However, it should cover the product or project description in enough detail to discuss, analyze, document and validate.

A business requirement can be presented in any of the following ways

- A table or a spreadsheet
- A diagram (workflow)
- A graph
- A model (entity-relationship diagram)
- A prototype or simulation
- A structured sentence or text template

How to organize and present a business requirement

Step 1) Categorize the requirements

- Place specific requirement to its relevant categories
- For technical stakeholders there should be technical requirement category, for non-technical stakeholders there should be generic requirement category
- Each organization should figure out which category suits their standards
- Categorization can also be done based on their types (functional versus business). Though this is not applicable to all cases

Step 2) Gather and arrange requirements in a logical order. So when stakeholders review the requirements, it is easy to navigate and also identify missing items.

Step 3) Prepare a list of the requirements that is meant to be reviewed by specific stakeholders.

For example, if a stakeholder is from technical background then he would like to know only the technical aspect of the product

Step 4) If tracing requirement to each other is difficult then use unique identifiers, ease in traceability.

Step 5) In certain scenarios, you might have to present same requirement in different ways for different stakeholders. For example, one stakeholder prefers a graphical format while other prefers a structured sentence format

Step 6) Prepare a table of content for all the requirements, it helps stakeholder to easily track requirements

Step 7) Use tools that help in presenting and categorizing the requirements

Step 8) In your requirement document, remove all unnecessary requirements, and organize requirement documents by process flow

Step 9) Map the requirements you have gathered to a particular step in a process flow, this will help reviewers to relate requirement to process flow

Step 10) Use a table for presenting complex requirement. Use bullet points to highlight the key aspect of requirement

Useful tips for presenting requirements

For better presentation and tracking of requirements for stakeholder, here are some tips that might be helpful to BA.

- Categorizing requirement is time-consuming and may not be feasible for every organization to create new category each time. For best practice, it is recommended that there should be a standard set of categories which can be commonly used by BAs, stakeholders, subject experts and technical teams
- Your requirement should be prepared in context to your audience. Understand who are the key players, influencers and decision makers. (Stakeholders, technical staff, developers, etc.)
- Define one requirement at a time. Each requirement should be atomic
- Avoid ambiguity by avoiding acronyms like etc., approx., and so on
- Do not refer to a requirement that is yet to be defined
- Avoid duplicate and contradictory statements
- Break complex requirement into manageable and reviewable points

- Avoid describing how the system will do something only mention what system will do

What is Change Control?

Change Control is the process that a company uses to **document, identify and authorize changes** to an IT environment. It reduces the chances of unauthorized alterations, disruption and errors in the system.

Why Change Control?

Whenever any new or different changes are requested for the system, especially by stakeholders, it is neither optional nor ignorable. It has to be implemented without affecting other components of the system. This is when the change control comes handy. It helps project teams to modify the scope of the project using specified controls and policies. Change Control is practiced whenever a project is not progressing as planned.

It is mandatory that a formal document for change request is completed and reviewed in order to keep control of change requests.

Number of question one might encounter while analyzing Change Control like

- Who will approve the change?
- Does it require to run through a change control board?
- How much time will be required to research and implement the change?
- What are the impacts of changes to other components of the system (schedules, cost, resources, etc.)?
- Is there any threshold under which the project management can approve it?

Different factors of Change Control process

There are various factors that a Change Control process should consider

Steps in Change Control Process	Action taken in Change Control
<ul style="list-style-type: none"> Change request initiation and Control 	<ul style="list-style-type: none"> Request for changes should be standardized and subject to management review Change requestor should be kept informed
<ul style="list-style-type: none"> Impact Assessment 	<ul style="list-style-type: none"> Make sure that all requests for change are assessed in a structured way for analyzing possible impacts
<ul style="list-style-type: none"> Control and Documentation of Changes 	<ul style="list-style-type: none"> A change log should be maintained that tells the date, person details who made changes and changes implemented Only authorized individual should be able to make changes A process for rolling back to the previous version should be identified
<ul style="list-style-type: none"> Documentation and Procedures 	<ul style="list-style-type: none"> Whenever system changes are implemented the procedures and associated document should update accordingly
<ul style="list-style-type: none"> Authorized Maintenance 	<ul style="list-style-type: none"> System access right should be controlled to avert unauthorized access
<ul style="list-style-type: none"> Testing and User signoff 	<ul style="list-style-type: none"> Software should be thoroughly tested
<ul style="list-style-type: none"> Version Control 	<ul style="list-style-type: none"> Control should be placed on production source code to make sure that only the latest version is updated
<ul style="list-style-type: none"> Emergency Changes 	<ul style="list-style-type: none"> A verbal authorization should be obtained, and the change should be documented as soon as possible

Process of Change Control

Before we look into what is involved in Change Control process, we will get familiarize with what documents are used in Change Control. While carrying out Change Control, there are mainly two documents involved

- **Change Log:** A change log is a document that list the details about all the Change Requests like project number, PCR (project change request) ID, priority, Owner details, Target date, status and status date, raised by, date when raised etc.

Project Acronym	Change Request Log	
Customer Name	Project ID No.: Project ID No	

Document Information

Project Name:	Project Name		
Prepared By:	Document Author	Document Version No:	Version Number
Title:	Document Author Title	Document Version Date:	Version Date
Reviewed By:		Review Date:	

Request No.	Description	Requirement Reference *	Initiator	Initiation Date	Price	Status**	Status Date	Priority (H, M, L)	Target Close Date
				TOTAL	\$	0			

Legend

* **Requirement Reference** – Identify the location of the description of requirements effected by this change, if applicable. This can be a Requirement ID in a requirement tracking log, or a section-page-paragraph reference in a requirements specification document.

**	Status
----	--------

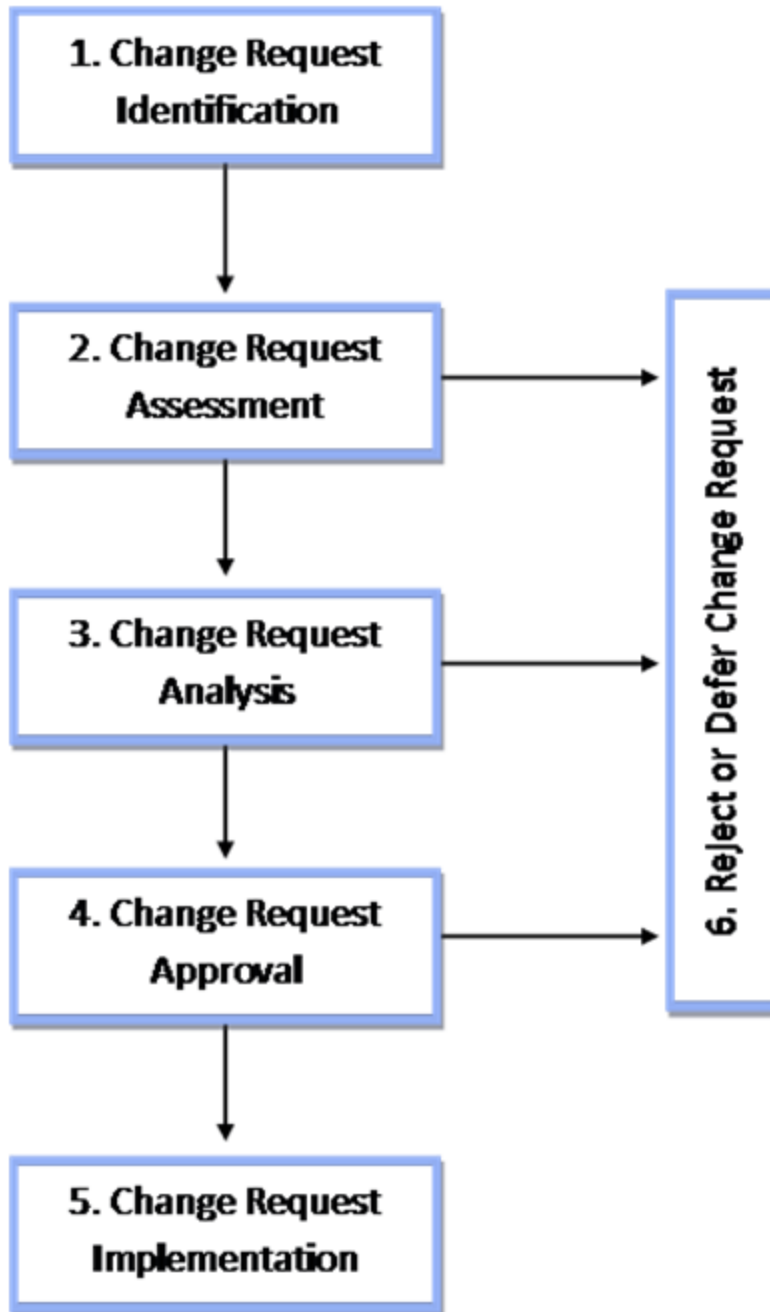
- **Change Request Form:** It is used to document details required to support the decision-making process like type of change, benefits of change, name of resource requesting the change, time and estimate cost, priority of change, authorized person detail, change request status etc.

Change Control Form

Section A		
Project		Change Number
Controlled Item		Item Version
Identification of Aspect to be Change	For Document give section number / page number For Software give Module, Screen or Report name	
Change Details Include indication of importance and urgency		
Tick if Continued Overleaf <input type="checkbox"/> ↗		
Requester of Change Print Name		Date Raised
Section B		
Investigator of Change		
Impact, give details of other items affected		
Investigation Outcome Reject / Action at No Cost / Action at Cost	Suggested Priority High / Medium / Low	Date Investigated
Section C		
Implementor		Date Scheduled
Section D		
Change Implemented	Signature	Date
Implementator		
Project Manager		

Change Process Flow-Diagram

Change Process follows a specific pattern to implement the changes in the product or system. Here through flow-diagram we explained what are the steps involved in the Change Process.



Steps for Change Control

Steps for Change Control	Action
<ul style="list-style-type: none"> Change request identification 	<ul style="list-style-type: none"> Identify the need for a change and describe it on the project change request form
<ul style="list-style-type: none"> Change request assessment 	<ul style="list-style-type: none"> If the change is not valid, it has to be deferred or rejected Determine appropriate resources required to analyze the change request Perform quick assessment of the potential impact and update the change request form At this stage, rejected change request should stopped
<ul style="list-style-type: none"> Change request analysis 	<ul style="list-style-type: none"> For analysis assign the change request to an authorized member Deferred change re-enter this analysis step At this stage, rejected change request should stopped
<ul style="list-style-type: none"> Change request approval 	<ul style="list-style-type: none"> Identify change risk and complexity level before approval Identify the impact level of the change before approval Review impact of Change Request to authorized person for approval At this stage, rejected change request should stopped
<ul style="list-style-type: none"> Change request implementation 	<ul style="list-style-type: none"> Update project procedure and management plans Inform about the changes to the team Monitor progress of change request Record the completion of change request Close change request

NOTE: The approval for Change Control may be done by **Project Manager, Lead IT or Lead Developer, Stakeholder.**

Change Management Vs Change Control

Change Management	Change Control
<ul style="list-style-type: none">It is responsible for managing and controlling change requests to effect changes to the IT infrastructure or any aspect of IT services to minimize the risk of disruption of services and promoting business benefit	<ul style="list-style-type: none">Change control includes activities like submission, recording, analyzing and approval of change to improve the overall performance of the system or product

The difference between a Requirement and a Specification

Requirements	Specifications
They outline “what” the software must do	They outline “how” the software will be created
They outline the software from the end-user , business and stakeholder perspective.	They outline the software from the technical team perspective.

There are a plethora of terms and terminology for various documents

Specification Documents like -

- SRS - System Requirement Specifications
- FRS - Functional Requirement Specifications
- BRS - Business Requirement Specification
- CRS- Compatibility Requirements Specifications
- PRS - Performance Requirements Specifications
- RRS- Reliability Requirements Specifications
- CRS-Configurations Requirements Specification

Requirement Documents like -

- BRD - Business Requirement Document
- SRD - System Requirement Document

Points To Ponder

- In many places these documents are not separate and are used interchangeably.
- Specifications and requirements roughly communicate the same information, but to two completely different audiences.

- For a given project which documents of above are created, depends on the “nature” of the project and the organizational “processes”

In this tutorial we will discuss SRS and BRS



BRS (Business Requirement Specification)	SRS (System Requirement Specification)
It describes at very high level the functional specifications of the software	It describes at a high level , the functional and technical specification of the software
It is a formal document describing about the requirement provided by client (written, verbal)	It specifies the functional and non-functional requirements of the software to be developed
Usually its created by the Business Analyst who interacts with clients	Usually its created by the System Architect who is an technical expert . Though in smaller companies the BA will create SRS as well. Some companies do not create SRS altogether. Their BRS is detailed enough to be used as SRS as well.
It is derived from client interaction and requirements	It is derived from the BRS

Business Analysis Process:

For any Business Analyst the biggest challenge after getting a project is, how to start and from where to start the project? Or what deliverables might be creating and how to complete the project successfully?

To help you with all sort of questions we have listed out some of the key techniques for a successful business process analysis.

It will give guide you step by step from the day 1 business analysis process till the end of the planning stage.

Step 1) - Gather all information about the project

It is business analyst's responsibility to gather all the details related to the project. By asking questions to people connected with the project (project manager, project sponsor, manager or business owner).

The information gathered should cover these topics

- Project scope and boundaries
- Current factors influencing the organization
- Project risk and constraints
- Broader organizational context

Stakeholders that are actively involved in it. This would be a good time to conduct Stakeholder Needs Analysis

After gathering all this information, analyze your part in the project and make a checklist that as a Business analyst you can include like

- From your previous experience what you can implement in your current project
- Documentation and planning required into the current project
- Discussing the possible outcome of the project with stakeholders
- Sort out the members that are involved in the project
- Look for any help from client - fixing a meeting with the stakeholders
- What are the expected deliverable and in what format it is required?

- What existing documentation you can review for better project idea
- Find out what methodology (Agile or Waterfall) will be appropriate for your project

Step 2) Set up a review meeting with Project manager/ Stakeholder/ Team members

An unclear agenda may lead to a failure of the project.

- Be specific in what is expected out of the project.
- Involve Project manager/ Stakeholder/ Team members in meeting and ask questions related to project
- It is very likely that you might be working on completely new project, in that case, ask project manager or contact person who has worked in that domain before

Step 3) Analyze all the project relevant documents like

- Business process documentation
- Business and system requirements documents
- Business cases
- Charts and flow diagrams
- Project plans
- Organization chart
- Strategy documents and business plans
- Policies and legislation

Uncover the information hidden in business requirement document and trace out any gaps with current systems, processes, procedures and operations. It is possible that document that is provided to you is out of date, so validate the information that you discover.

Step 4) Record all the facts and information that you discover

From your research and analysis, you may discover many useful information relevant to the project that has to be changed or implemented in the project. Record them.

- Business requirements including reporting requirements
- Business processes and supporting systems
- Functional and non-functional requirements
- Issues and risks that are currently influencing the project

Step 5) Understanding the problem domain

By now you will have a good insight of the project, now you can identify the problem domain in the project. You need to find out

- Exactly which business function will be affected
- Risks and factors affecting the business
- Policies and constraints that influence the project
- Values that determines the level of importance of the project
- System currently supports the business activities
- Document giving brief about the problem domain- e.g., Annual Report
- Issues currently blocking the business to achieve the desired outcomes
- On proposed change does it make any difference to problem domain

Step 6) Presenting your Business Requirement

Once you have gather all your business requirement and understand the problem domain. The next step is presenting your Business Requirement to Stakeholders or Project Manager. There are numerous techniques which can be used for presenting requirement like

- A table or spreadsheet
- A diagram or graph
- A prototype or simulation
- A structured text template or structured sentence

Glossary of words that will give quick overview of business analyst process

- **Purpose:** Defines the purpose of the business analysis activities required for the proposed initiative
- **Scope:** Defines the deliverables that are included and excluded
- **Root Cause:** Define the root causes of the issues identified
- **Current Condition:** Defines the problem that cause the need for change

- **Planned Activities:** Defines the reason for the activity, deliverables, and delivery dates
- **Stakeholder Engagement plan:** It gives an overview of the stakeholder engagement process
- **Quality Management:** It describes the activities that will ensure the quality of project deliverables
- **Target Condition:** Defines how critical issues identified will be addressed

Quick tips for Business Analyst

- Ask questions in meetings
- Be prepared before stakeholder meeting or review
- Be adaptable to change and new experience
- Manage expectations
- Respond to feedback