

### **What is business analysis?**

One should not restrict the BA role to only being a link between Non-IT and IT or only for development projects. A BA is someone who is able to bring in improvements, changes (technology, process, people etc.) in an efficient manner. So a BA could be part of the marketing team who helps the marketing team in providing estimates/high level solutions for a said project which is under the process of procurement. Or he could be someone involved during the Requirement gathering/analysis once the project is initiated. Or he could be someone who brings profit to the company by performing process improvement activities ROIs at process level. Last but not the least BAs could be domain specific as well.

### **What is the career path for a Business Analyst?**

A Business Analyst in the IT field has many varied directions among which to choose a career path. The most direct would lead to a Lead Business Analyst position and then Project Manager whereby the incumbent manages projects through the entire lifecycle from inception to post-implementation including the management of business analysts system analysts quality assurance analysts and most likely development project managers or team leads. That path would then lead to Program Management perhaps PMO management or Product Manager and on to Directorship. In addition a good Business Analyst may find they are heading toward a Customer Relationship Manager position whereby they become the primary IT interface to a given Business Unit (BU). This role most often leads to a position within the BU as a Manager of Applications or a Process Management role. Process Management opens many jobs including process re-engineering quality program development and large scale or enterprise process management programs such as ITIL or Six Sigma initiatives. These roles will continue to proliferate as companies realize the benefits of having a SME in process and quality. And still many Business Analysts find their understanding of business process entirely portable into purely system related positions in the business side that are only peripherally related to IT. These of course may lead to quantitative roles manager roles or operational roles such as supply chain logistics etc. Of central importance to a successful Business Analyst is the interest in speaking to people. Face to face verbal communication is paramount to support other tools such as surveys and diagrams. Incumbents must be interested in understanding not only the pieces that comprise a system but the people that comprise it and the realities that embrace the system. Briefly the Business Analyst must understand and not judge the what should be and the what is

### **How do you resolve issues?**

I would rather focus on issues and the facts related. Origin of issue, severity of the issue, implications and possible solutions to solve the issue. Try not to focus on the person who brought up the issue. Another important part is how to avoid similar issues in future

### **What are the problems solved by business analysis?**

As a BA the most critical part is in gathering requirements (we should understand them very well from a Business User /stake holder point of view!!!)

Reason: There might be a chance for the whole project to go in the wrong path due to wrong understanding of the Business users/ Stake holders' needs and the gathered requirements created for the work following that step... i.e. going from A to C instead of going from A to B.

Notes: (Business Users: are the individuals who work in organizations in different departments like Logistics accounting finance Inventory) in the company who wanted the software in Place for them to work on to help the Customers.

Stake Holders: Someone who is related to the Project? 2 types of People are involved:

Direct Stake holders: business end users customers developers tech team.

Indirect stake holders: management etc.

The Project Manager responsibility (usually) identifies the stakeholders determine their needs and expectations and more important must manage and take their help for the project success. (You should Understand them well to provide them with right service for the right success of the project)...

SME's: are the Subject Matter Experts who know about that project and have in-depth knowledge about that software application used and that particular business domain knowledge like Finance (terms and permutations etc.) Accounting (Business Planning Ledger maintaining Forecasting) Mortgage (Local banking rules Knowledge about compliancy of applications forms/ applications that needs the authorizations of the local Government bodies or counties Underwriting conditions (How flexible the Loan lending organizations at the individuals credit check or History)

So The SME's help the Project Manager or BA to help them understand about the necessities or needs of the Business Users or Stake holders like/interests- (How the Project help save time for the transactions or? how much secure/security is needed the application wise or profitable over long run) and SME's explain How the Stakeholders or Business Users want the application to be or appear to be for the Customers or Business Users).

### **What is the difference between data model and an entity relationship diagram?**

A data model is a model which shows how data is stored and used for e.g. a normal database It has 3 main parts 1) Structural part:- how data is structured 2) Integrity part:- Rules governing structure 3) Manipulation part:- operators used to select, update, query that data, eg select, update, delete commands in sql To further add Data Modelling is when we add this theory to Live instance. ENTERPRISE DATA MODEL (ENTERPRISE RELATIONSHIP MODELING) :- This can be called as a conceptual model or semantic model The sub parts of an ERM are 1) Entity:- It is an object, eg employees, computer 2) Relationship:- It captures how two or more entities are related to each other 3) Attributes:- Every entity has its own sets of attributes (e.g. PAN no in India for each employee or SSN in US) To clarify the point look at eg A employee is an entity belonging to entity sets (All employees) which has a relationship with department, and attributes is emp code

### **Who uses the output produced by business analyst?**

The output will be used by the Both IT and Non-IT People, as IT people use this document as key for the building of the application and Non - IT people use those document where they can see prototype of their application.

### **What is the educational qualification required for a business analyst?**

There is no specific qualification for a business analyst. Well, if you are a management graduate it is an added advantage since you have they have better communication skills. One important thing a BA needs to have is domain knowledge or business knowledge. Unless he/she understands the client's business process thoroughly they cannot draft the requirements properly.

### **Mention the components of UML?**

UML uses many concepts from many sources. 1. For Structure: Actor, Attribute, Class, Component, Interface, Object, Package. 2. For Behavior: Activity, Event, Message, Method, Operation, State, use case. 3. For Relationships: Aggregation, Association, Composition, Depends, Generalization (or Inheritance). 4. Other Concepts: Stereotype. It qualifies the symbol it is attached to.

### **Mention some of the important points a business analyst must take care while preparing business plan?**

While Creating Business Document, Make sure you start from small problems. Don't jump to big problems right way. Keep the Business sponsors and IT folks in the loop. Make sure your document clearly state

Exceptions, Assumptions and Limitations. Sometime you need to keep in mind the legal issues. Business document should be well written for usability and for future projects.

### **Why is business analyst position vital in an organization?**

The position is important because a BA is a people's person when it comes to the users and an IT person when it comes to the developers. He can communicate with the users in jargon that they are comfortable with and is able to understand them in order to collect solid business requirements. Simultaneously he can effectively communicate these requirements and support them with documentation for a developers benefit.

### **Why excellent communication skill is essential for a business analyst?**

A BA is one who sits with the client understands it and then tells the IT people what needs to be done hence BA needs to have excellent communication skills

### **What are the industry and professional standards followed by business analyst?**

Industry standards that have been set for the BAs to follow are OOAD principles and Unified Modeling Language (UML). This is a common language used by business analysts all around the world to draft the functional requirements

### **What are the quality procedures followed normally by a business analyst?**

For quality there is no specific mark of course Six sigma and ITIL (Information technology infrastructural library United kingdom) are certain quality standard establishing organizations and methods. But As a normal the following should be followed:

The quality of communication while gathering requirement should be excellent and outstanding. Sometimes users are just looking for functionality in system and they are not even able to say that what exactly will be their dream functionality which will be most convenient to them. In that case BA should explore them and figure out the exactly demanded requirements.

### **How is requirement analysis done by business analyst?**

Requirement session is usually done through JAD session. Business Folks and Major sponsors are always there along with some technical folks. Business analyst then goes through each requirement and asks for the feedback. If Business Sponsors and Technical Folks think that all the requirements are according to the business and won't be a barrier to existing system. They get the official signoff on Business Requirement document. IT manager and Business manager both do the sign off on that business requirement document.

### **Does the business analyst interact with clients directly? If so state the reason for the same?**

It depends on the project to project it is not always the same that we do interact with the clients directly, some time there will be a team whom might be interacting with the client and gives you the requirement and if have questions either we do talk with that team or our manager.

### **Mention the difference between business process improvement and business process reengineering?**

Business process improvement implies changing a step sub step or any part of the process i.e. process is not completely changed In BPR we actually study the business and find out what is the best way I can carry out the process and change the whole way the process runs(business process redesign)

## **What is UML?**

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems

## **How is business plan evaluated?**

A business plan is evaluated by checking the contents of the plan such as if the plan have based on the resource planning and envisioning phase of the project.

## **What are the problems Business Analyst could face during gathering Business requirements**

The availability of the people (e.g. managers, supervisors and the end users) the BA wants to talk with for gathering business requirements. These people have regular daily works to do and their time to spend in the gathering sometimes hard to schedule and for this reason gathering business requirements is delay.

## **What can a Business Analyst do differently than project or program manager (Design Architect) with respect to successfully getting the project implementation done?**

Business Analyst role is not entirely different than Project manager role but Project Manager is bigger role than business Analyst.

Project manager is responsible for all the deliverables like

- schedules/ timelines
- resources management
- risk management
- Daily/weekly status report to project stack holders etc.

## **where as business analyst sometimes report to project manager or may report to business manager.**

Business Analyst deals with business users to gather requirements prepare RD, FD and coordinate with development team for development and then do the testing involve with users in testing get the sign off and move component to live environment.

I hope this clarify the roles of PM & BA.

## **Where would you document Functional and Non Functional Requirements (i.e. deliverable)?**

Functional Requirements are documented in the SRS document / Use Case Document. Non Functional requirements are listed in the SRS document.

## **How do you identify the basic flow? What would you do if someone was struggling to determine the basic flow for a use case?**

Basic flow for use case can be identified from Business Requirement Documents or Functional Requirement Documents as these use cases are prepared on the basis of these requirement.

## **What is the relationship between use case and test case?**

A use case is written from a "user" perspective describing the interaction of a piece of software between the user and the software. These are written in common language typically from the business or user

point of view and in enough detail for the developer to create a piece of software. Typically written in a MS Word type tool. Use cases capture the functional requirements of the system. It describes the expected interaction the user will experience, in detail. The audience is the business, for signoff, and technology for development.

A Test Case is written using the use cases for a source. It takes a use case to a deeper level so that software testers can exercise every possible scenario that could occur, negative and positive scenarios. One Use Case can turn into 10 test cases. 10 test cases make up a test script. Typically Test Cases will be written in a testing tool like Test Director, but also can be written in MS Word. The audience is QA testers.

**What would you do if the client says that you and the other analysts cannot directly talk to the users?**

If this happens then explain the purpose of your talk (e.g. capture requirements) and why its important to talk to users directly (e.g. the quality of requirements will be better if they comes directly from the users mouth). Explain them that it will be a high risk to the project if analyst can't talk to the users directly. Client can give access to indirect (surrogate) users but explain that the quality of requirements will be not good. Hopefully your client will agree by now otherwise flag it as a higher risk in Business Requirement Document and highlight during your meeting with your PM and Project Sponsors. Now, it's your PM or project sponsors duty to provide you access to those direct users. If they can't than you are safe anyways.

We are going to a client on Monday to help them with their requirements. We have just received a business case from the client, and they have no tools in place. What would we do the first week? First week in this case is always advisable to do a due diligence of the amount of work, expectations, existing process, time lines with the constraints surrounding. One of major constraints in this case would include lack of tools.

Depending on the project timelines, complexity and volume of the project present your recommendations for tools to be used and the estimated budget allocation required. Document the comparison of productivity and flexibility with and without tools used.

This should help the project sponsors to take a call on going for tools

Version control and configuration management are terms used widely in the business industry, write short notes about the terms.

By definition, version control is essentially a subset of configuration management. It is usually concerned with the handling changes arising in previous documents as opposed to configuration management which essentially handles the individual components.

Good documentation management systems are highly recommended in system development; briefly describe the factors that contribute to a good documentation management system.

For a documentation system to be considered good, the following factors should be prevalent in it: It should be made in such a way that it can accommodate future changes, including version changes, bearing system security features such as providing access only to the allowed users, i.e. have good authentication features. In general, one should take in data as well as information security measures in

place, putting in mind that the documentation should also be able to bend to the changing needs of its users as well as the market conditions.

### **State the different software methodologies.**

The term software methodology, software development methodology and software process mean almost the same thing in computer software or system development, i.e. the activities carried out by computer system engineers or computer software engineers in an attempt to procure particular computer software that servers a certain function or purpose. This includes the framework adopted, structure, plan as well as the control of the resources engaged in the software or system development process.

There are so many software methodologies and the choice as to which one to adopt is usually dependant on so many other factors such as the purpose of the given software, the prevailing conditions regarding the software development environment and the will of the company or the client procuring or intending to use the final software as some clients even look into the software or system engineers' methodology to as one of the factors determining whether to contract him or not.

Regarding the purpose of the software, let's look at the following, example is a situation of a safety critical system such as an aircraft navigation system and a business system, one would find that in a business system, one can have its prototype done and users start using it as they identify its weaknesses and tell the engineers to rectify whereas in an aircraft navigation system, no weakness would be allowed at all for it can cause huge loss of property and life thus all the possible identifiable weaknesses are eliminated first before that system comes into operation. Much stories and arguments apart, the following are the available software methodologies:

- i) SLDC- Software Development Life Cycle, also understood as System Development Life Cycle which encompasses activities such as Analysis, Design, Implementation, Testing, Inauguration and Maintenance in that order and then back to Analysis, note that it is a cycle hence once we get to the last stage, i.e. the maintenance we still go back to the analysis stage and move along to the maintenance once more iteratively
- ii) The RUP – The Rational Unified Process, which when looked into intuitively is an iterative software development process framework that was created by the Rational Software Corporation in the US which is a division of the IBM (International Business Machine). However, this process is usually not considered as a single prescriptive framework yet as an adaptable process which can be tailored by the development team or organization selectively in order to end up with their respective results depending on the needs prevailing either on the client's side, the industry standards or even the development constraints which involve time, scope as well as the budget, Intuitively, this process has characteristics overlapping with other development processes and methodology as will be seen when considering the other methodologies below.
- iii) The iterative process
- iv) The waterfall model
- v) The agile software development methodology
- vi) The XP (Extreme programming)

- vii) The ISO 9000 methodology – provided by the International Standards Organization
- viii) The ISO 15504 model – also provided by the International Standards organization
- ix) The Capability Maturing Model Integration (CMMI) which replaced the former Capability Maturing Model (CMM)
- x) The Six Sigma methodology
- xi) The Test Driven Development (TDD)

**Describe the abbreviation OOAD as used in Object Oriented Programming.**

The term OOAD is an abbreviation of the phrase Object Oriented Analysis and Design. Contrary to the traditional programming, also called procedural programming whereby the entire code of a given program is written line by line, from scratch. There is a new more powerful approach to software development or simply programming referred to as the Object Technology where predefined objects pertaining to particular situations are pre-designed by experienced software engineers and then the programmers just call them into their code in order to implement a given function in their code.

Just the way experienced civil engineers design building blocks for particular situations in a particular house so that the inexperienced mason just lays them appropriately in order to end up with a nice house is the way experienced software engineers make these objects. This helps even novice programmers to use the objects to build nice computer software or a program. It is the analysis and design of these objects with intention to make good software that is referred to as Object Oriented Analysis and Design, the OOAD. Programming languages that use the Object Technology include C++, JAVA, and the PHP among others.

**Describe the meaning of the term data mapping.**

By definition, the term data mapping is the process by which a system developer creates data element mappings that relates two models of data (databases) in order to assist in data integration. This usually assists in the following manner:

- i) Data mediation or transformation between the source and the destination of data
- ii) Assisting in data lineage analysis by identifying the data relationships
- iii) Assists in data masking by discovering sensitive data
- iv) Assists in data de-identification process
- v) Assists in consolidating multiple databases into one thus identification of redundant columns and advising the developers for consideration or even elimination.

**Describe the term black box testing.**

Black box testing is the type of testing whereby the entire unit is tested as a whole without considering the contents or even how the inner components of the unit under test work, the tester's only consideration is to enter a known input signal and check whether the output behavior is the one expected out of that unit given his input signal.

**Give the importance of using a flowchart.**

It is easier to interpret as it is graphical in nature and thus all persons involved in the project development can understand it with ease.

**Briefly explain the use case model.**

This is a model used by software engineers to describe the business environment of a given project. It encompasses of a series of workflow that are pertained to a particular actor.

**What do you understand by the term UML?**

The term UML is an abbreviation of the term Unified Modeling Language which is the standard language used in construction of as well as visualization and documentation of varied system components. It has a collection of graphical notation techniques used in the development of abstract models for certain specific systems.

**Describe the importance of an activity diagram.**

This is the diagram used in a business system to show the workflow involved, activities happening as well as the completed actions. In a company comprising of several departments e.g. the medical department, accounting department, and even the human resource department, usually each department has its own peculiar privileges to the system, for instance the medical department can only be allowed to access the screens related to their activities such as medical records while the human resource department will be allowed to view only the screens which are relevant to them too, thus these activity diagrams assist in showing the relationship between particular activities with their relevant and related departments so that during coding, the coders may refer to them to implement the discrepancies accordingly. Designers too can be guided by these activity diagrams.

**How many types of diagrams do you know and what do you know about them?**

Am aware of two types of diagrams namely the use case diagram and the collaboration diagram, the use case diagram has been discussed above and as a result I will only talk about the collaboration diagram here, these are diagrams put into being by modeling the objects of a given systems and then representing the prevalent associations between the objects in questions with the use of links.

**Describe your understanding regarding the so called alternate flow in use case.**

These are the contingent flows that arise when a system fails to curb an encountered situation and thus the system doesn't result in the expected results. When the system resorts to the alternate flow under this circumstance, it may still end up yielding the expected results.

**Describe your understanding regarding the exception flow in use case.**

This is generally unpredicted situation that may lead to undesired result under normal circumstance in a system; several methodologies called exception handlers are available to help control such situation

Describe the meaning of the following words as used in the use case scenario:

- i) Extends
- ii) Includes

In the use case scenario, the term extends is used to imply that a certain action needs to have taken place in order for the other to take place too whereas includes implies that it is not important, as in the action may take place or as well may fail to take place but the other will still take place.

**What are the documents related to the use case?**

There are two documents related, namely the FRD (Functional Requirement Document) and the SSD (System Design Document) or the TRS (Technical Requirement Specifications).



**Describe your understanding regarding logical data model.**

It is the data model, which is not actually physical and describes how data is physically stored in the given database.

**Describe your understanding regarding high level and low level use cases.**

The high level use case usually refers to the entire business process whereas when it is divided into smaller units, the outcome or the sub units are what are then referred to as the low level use case

**Describe your understanding regarding the SDD.**

This is the abbreviation of the term System Design Document; it acts as the mediator between business users and the system developers so as the system developers may understand the business requirements of the system they are developing in order to know where to put emphasis and end up with a quality and objective based system.

**Describe your understanding regarding the following terms**

- i) URS
- ii) FS

The URS is the User Requirement Specification whilst the FS is the Functional Specification; traceability matrix is usually used to keep track of these requirements. TEST DIRECTOR can be used to do the traceability of the given requirements during the testing phase.

**How is use case prepared?**

It is prepared using drawing application software such as the Microsoft Visio and the also the Rational rose.

**Describe how you would participate in testing as a BA (Business Analyst).**

As a Business Analyst, I would participate by reviewing the test cases to ensure that all the stipulated requirements have been met by the system in question.

**Describe the main qualities of a good requirement.**

There are several qualities regarding a good requirement but the most outstanding ones include the:

Clarity – the requirement should be clear enough to be understood by its users.

Understandable – the requirements should be put in a manner easy to understand by users of all levels.

Consistent – the requirement should be such that it doesn't contradict itself, it is important noting that during system development, all users need to be consulted, including the managers as well as the junior staff, one would find that the managers would like a wider control of the system so as to monitor the junior staff to the date whereas the junior staff are objecting these view, hence a contradiction. When this issue is not considered carefully, usually through consultation or negotiation either the managers or junior staff may resent the system thus by this, once the users resent the system, obviously it will not be exploited to the maximum thus lowering the benefits the organization derives from it thus consistency must always be considered in the system development requirements.

Verifiable – The requirements of a given system should always be verifiable as in they should be put in a manner that can be checked across in future so as one can clearly identify whether the particular requirement has been met or not, it is usually advised that the requirements are put in a manner that during verification, the answer is either true or false and nothing vague as it is during this stage that legal action can be taken by the either the contractor or client if at all the answer is no and always with the law, matters of doubt are generally not recommended.

### **What is the meaning of the word UML?**

This usually is the abbreviation of the Unified Modeling Language, a standard language in the system development used to implement the understanding, documentation and construction of varied system components

### **Describe the diagrams which should be known by the Business Analyst (BA).**

The Business Analyst (BA) is expected to be conversant with the following diagrams:

- i) Use case Diagram: this is the diagram which gives the details concerning the given business environment, this entails the series of action usually performed by given actors such as analyzing the procurement portfolio, giving out an order to a certain supplier, acknowledging the reception of the goods, processing them as appropriate, doing the relevant marketing, handing the goods to the hands of a customer at a profit, receiving payments, either by cheque or cash, printing a receipt, and entering the transactions into relevant accounts, making payrolls, preparing final accounts including the balance sheets as well as the profits and loss accounts.
- ii) Activity Diagram: this is the diagram which is usually employed in early analysis stages to describe the involved components.
- iii) Sequence diagram: This is the type of diagram used to tell the way particular objects interact with other objects in a manner arranged in both time and sequences. This is usually very useful for system developers as well as the system testers as it enhances the level at which a given system can be understood.

### **Explain where you would use the rational rose and the requisite pro.**

In a situation whereby different modules of a given requirements have been created for varied functions, then collected together and made into a single document, the requisite pro is the one which comes in handy. The other one, the rational rose, is used to create the business model as a visual representation. It is helpful in creating high level and low level use cases, activity diagrams, state diagrams, collaboration diagrams, sequence diagrams etc.

### **What is mean by logical data model?**

Data model tells clear details about the data and how the data is stored physically in a database.

### **What do u mean by high level & low level use case?**

A broad view of a business process is called a high level use case. And if we divide the big view into different small sub use cases, then it is called low level use case.

### **What do you know about SDD ?**

It is also called system design document. My role as a BA is just a mediator or a middle layer between business users and developers and we make developers to understand the business requirements.

### **What do understand by URS & FS ?**

User requirement specifications and Functional specifications. To keep track of these requirements, we generally use Traceability matrix.

By using Test director we can do traceability of requirements n testing phase.

How do you prepare use cases?

BY using MS Visio and Rational rose.

How do you participate in testing as a BA?

I participate mainly in reviewing the test cases to see if all the requirements have been met.

#### **1. What is the main quality of a good requirement?**

The requirement should be good, clear, understandable, and consistent and should be easily verifiable.

#### **2. What do u understand by UML ?**

UML is basically Unified Modeling Language. This is the standard language used in the system to understand, document, construct different components in the system.

### 3. What are different diagrams to be known by a BA?

Entity relationship diagram, data flow diagram, use case diagram, class diagram, activity diagram, state chart diagram, sequence diagram, collaboration diagram, component diagrams, deployment diagrams etc..

**Use case diagram:** basically explains the business environment. Series of all related actions performed by actor.

**Activity diagram:** Used in the early stage of analysis and designing level. It describes each individual component.

**Sequence diagram:** It tells the objects interactions with each others arranged in time sequence. Very useful for developers and testers to understand the system better.

### 4. Where did u use rational rose & requisite pro ?

When we created different modules of requirements for different functions, and finally collected all together and made a single requirement document, we used requisite pro to do this.

And we used rational rose to create the business model as a visual representation.

Created High level & low level use cases.

- Activity diagrams
- State diagrams
- Collaboration diagrams
- Sequence diagrams

### 1. What do understand by version control & configuration management ?

Basically version control is a part of configuration management. Mainly it handles when the previous document changes. Where as configuration management handles the individual component.

### 2. What is meant by good documentation management system ?

Should allow to make any changes if required.

Good security features.

Should be able to change versions.

Authorizations to only required people. (renditioning capability)

Hide imp information from others. (redaction capable)

### 3. What are different software methodologies.?

SDLC, RUP, SEI-CMM, Six sigma, SWOT, Cost benefit analysis, Risk analysis, Gap analysis.

### 4. What is OOAD ?

Object oriented analysis and designing. Used in coding od object oriented languages like c++, Java, and SAP Badis etc.

### 5. What is UAT ?

User acceptance testing.

If the UAT fails, BA did not understand the requirements properly.

### 6. What do u mean by Data mapping ?

It is the mapping of data from source system to a destination system.

### 7. What is black box testing?

It is completely a functional testing. i.e the tester need not know how it works technically. He only bothers what input he is giving and what output he is getting.

### 8. What do u mean by white box testing?

It requires slight programming knowledge to examine the outputs.

### 9. What is bug?

Mainly used to see the performance issues and system hangs.

### 10. How do u measure the quality of a product?

We do it by seeing min bugs in the product according to standards maintained by company.

### 11. What is RAD ?

It is called as rapid application development.

It is a development process that is used to build applications in smaller periods like 50-70 days i.e with some compromises.

## 12. What is ETL ?

Extraction Transformation and loading. Used mainly in data warehousing.

## 13. Types of testing ?

Unit testing : by developer  
Black box testing : Functional and module level.  
Ad hoc testing : Random testing..no particular process.  
White box testing : Very detailed..into the code.  
Exploratory : ad hoc testing with some purpose/ goal.  
Front end : for web based applications.  
Back end : database level  
Regression : Testing again and again the same application.  
UAT : User acceptance testing  
Integration : testing the interaction of 2 or more than 2 modules at a time.  
System testing : Testing all the modules together.

## 14. Roles of a Business Analyst

**Business Analyst** – this is a term which has come into prominence in the past few years especially with the advent of the software industry. Who is a Business Analyst and what's his role in an organization? These are the questions which we will be trying to answer here.

**Business Analyst** can be termed as an analyst who can delve deep into the business, understand the processes and make use of the knowledge in the betterment and success of the organization. But the term Business Analyst is used very generically in today's professional environment. It can mean analyzing the system, business processes, requirement analysis, supporting the business or system functions, handling the sales or financial KPIs' (Key Performance Indicators). But we will discuss the main responsibilities of a Business Analyst in a generic environment and it may happen that in some cases, it's an amalgamation of roles or may be a subset of another role.

**A Business Analyst** is responsible for a host of processes and activities which are elaborated as follows:

a) **At the Project Initiation process**, it's the responsibility of the Business Analyst to cover the high level scope and objectives of the project and establish communication channels

b) **Understanding the business** processes of a section or whole of the organization in a very clear cut manner so as to implement that knowledge in any required manner.

c) **Clear Understanding and communication** of Requirements is a very important aspect of a Business Analyst as it ensures that there is minimum gap between the expectations of the end users and the final deliverable from the technical team.

d) **Analysis and Documentation** should be very precise and clearly understandable so that starting from the end users or stakeholders to the developers can understand the underlying stated expectations in the requirement documents.

e) **Solution assessment and validation** is one of the main roles of a business analyst as it should be ensured that there are no gaps in the requirement process to the development stages.

f) **Regular interactions** by the business analyst with the developers and the module leads is essential as the knowledge transfer of the user expectations should be made clearly

g) **The business analyst** has a major role to play in the testing phase where he can actually take part in the systems testing phase and also provide support during the acceptance testing phase.

**g) After the implementation** of the software system, the business analyst also may need to handle the change management process if there are any new requirements or changes proposed.

**The business analyst** profile actually encompasses different roles like that of a process analyst, system analyst, project manager, application support, data analyst and tester. Gaining all round knowledge in all these different role types will definitely give the Business Analyst an edge and will enable him to overview the project from all angles.

Implementation of such responsibilities will help the Business Analyst become the interface between the users and the technical team. The organization should also be responsible for guiding the Business Analyst through his correct responsibilities for the better advancement of the individual as well as the company as a whole.

#### 15. What is Business Analysis?

**Business analysis** can be described as the sequence of activities which are implemented in order to assess the business requirement needs and to fit the required solution so as to bring around the success of the organization and business. So, this sequence of task is normally performed by a "Business Analyst" or BA. Business Analysis can cater to different industries and so there are specialists created among business analysts. For e.g the business analysts who solely work on developing IT systems are the Technical Business Analysts, and the ones which cater to the functional parts of the business processes and their improvement or re-engineering are known as Process Analysts.

**Business Analysis** is as such a vast subject and hence we will categorize it into subsets for better understanding of the various stages in any business process or software management. Business or Enterprise Level Analysis is the study and analysis of the business needs and the identification of initiatives to steer the organization on the path towards its strategic goals. This can include the finalization of the project scope, purpose and objective. This is the stage where the actual feasibility analysis occurs wherein the actual cost benefit analysis (CBA) of the project occurs and its evaluated whether the project should go ahead or not. Requirements elicitation and communication is vital to the basis of **business analysis** as it involves the actual collection of data from the stakeholders and ensuring that their requirements are clearly illustrated and conveyed to each and every member involved in the project. This is the level at which the actual requirement needs are captured using various requirement methodologies like Zachman framework etc. Requirements analysis or engineering has been synonymous with Business Analysis always and represents the requirements planning, development and management processes. At this stage the actual analysis of the requirements is done wherein the raw requirements are processed into functional objectives. The documentation of the requirements is done at this stage and may include the functional as well as the non-functional documents together with supplements like prototypes or UML diagrams. Finally, we come to Solution assessment and validation which ensures that the proposed solution design is in line with the requirements and there are no gaps in understanding which will trickle down the software development life cycle. At this stage, the requirements have taken form and have been converted to a solution design which can be developed and implemented as an application or software system. So its essential that analysis of the solution is done properly to ensure that the requirements are in synchronization with the solution. Business Analysis is not limited to this stage and can extend to the other parts

of the project life cycle with significant contributions at the design, development, testing as well as implementation stage.

***Business Analysis, in summary, is the art of managing the requirements and the business needs and synchronizing them in line with the strategic objectives of the organization. In order to implement this management methodology, one needs to understand that Business Analysis forms the base of the successful implementation of any business process or software management event in an organization.***

#### 1. What is a Sequence Diagram

**Sequence diagrams** are part of UML(Unified Modeling Language) diagrams and come under the interaction view as they depict the interactions between the entities and the transactions that are taking place with the trigger point and the end point clearly distinguished. The diagram shows the different processes as vertical columns or lines and the messages or interactions between them is represented by arrows with the arrowhead pointing towards the receiver away from the sender. The name of the message is written above the messenger arrow line. It also includes the sequential order of events which will occur from the start to the end of the process(es). An important part of the sequence diagrams is that time passes from the top to the bottom. A message sent between two entities can be synchronous or asynchronous type. A synchronous type of message indicates that the sender will wait till the receiver has finished processing the message and then only proceed while in asynchronous message type, the sender will not wait for a response that the receiver has received and finished processing the message. A synchronous message is represented by a filled up arrowhead while an asynchronous message type is represented by an open arrowhead.

**The sequence diagrams** are helpful in detailing the flow of transactions between the entities such as actor, database, controller etc. Hence, for a sequence diagram to be prepared, its essential that the “use case diagram” would have been finalized, else it could mean rework might be required if the use case diagram is revised. Sequence diagrams can be used by business analysts in their functional documentation process or by solution architects or designers in their design models. But whether the sequencediagrams are created by the analyst or technical designer, whats important is that the diagram conveys the right message across to both the user groups and the development team.

**Sequence Diagrams** are a clear and simple way of depicting to the users, stakeholders and the technical team how the processing of messages will happen and an assessment of this will go a long way in clearing up any gaps or misunderstandings at the requirement level.

#### 1. What is a Class Diagram

For **understanding class diagrams**, we would need to understand UML first. So what is UML? UML is short for Unified Modeling Language and is a second generation notation for diagram-based object-oriented modeling. It was first developed by the company Rational Corp.(Booch). After that UML was advanced as an industry standard by the Object Management Group (OMG).

**Class Diagrams** are a part of the structural view of UML as they represent the static structure of a system. Class diagrams are basically used by Business Analysts or Solution Architects to design the static view of the classes involved. The diagrams depicts the grouping of classes which have the same attributes and behavior(operation or functions) and also it includes the interrelationships between two class. A class is an entity which is represented by a simple rectangle and is divided into three parts. At the top we have the Class Name, in the middle the list of the attributes specific to the class is included and lastly comes the class operation or function. If a simplified version of the class diagram is depicted then the last two compartments are not included or are left blank. The interrelationships are shown in the

form of interconnecting lines between the classes and the dependencies are represented by symbols such as 1, 0, \*(many). This part is similar to the data modeling diagram – entity relationship diagram.

**As class diagrams** are essential to all object oriented analysis, its used in 90% of the software projects with UML diagrams. But you should keep in mind that even though there are a number of UML notations, the class diagram should be as simple and clear as possible without complicating it with unnecessary notations. Classification of classes should be done keeping in mind the object orient principles and after listing the relevant classes you can depict them with the help of the class diagram. An **example of a class diagram** is given in Figure A to give you an idea of the structure of class diagrams:

#### **Figure A : Example of Class Diagram**

In Fig A, lets take the “Dishware” class, it has three compartments. At present the attributes and operations compartment have been left blank, Each operation is prefixed a “+” sign to depict that its a function and suffixed by “()”. The variables which will be input or passed through the function can be included within the symbol“()”.

Also included in the example are six other classes “Plate”, “Bowl”, “WoodenPlate”, “GlassPlate”, “WoodenBowl”, “GlasBowl”. The two classes “Plate” and “Bowl” are generalization of the main class “Dishware”. This can be depicted by the hollow triangle symbol as shown in Fig A. The two classes “WoodenPlate” and “GlassPlate” are generalization of the class “Plate” and similarly for the class “Bowl”, the two classes “WoodenBowl” and “GlassBowl” are generalizations. Generalization means that the sub classes will inherit the behavior of the main class but will have attributes their own as well. Lets take the example of “GlassPlate”, it will have the attributes of the class “Plate” like shape etc nut will also have its own attributes.

#### **1. What is RUP (rational unified process)?**

**RUP** stands for Rational Unified Process. Its a software development process framework which has been taken forward by Rational(part of IBM corporation)

Its a proper guidelines to be followed so to achieve standardization and improvement of the existing processes. RUP has phases and iterations which have to be followed with the help of templates and guidelines implemented at each stage. The intention of usage of RUP is to provide standards for all stages of the software development life cycle. RUP is supported by different tools and methodologies among which is **Rational's UML(Unified Modeling Language)**.

In the figure, the Rational Unified Process Model can be diagrammatically viewed. As depicted, RUP has four distinct project life cycle phases:

**a) Inception** – is the part where the actual exploration of the concept happens with the stakeholders management, definition of the project scope, cost benefit and feasibility analyses. This is the starting phase of the project and actually provides the foundation for each of the ensuing processes.

**b) Elaboration** – This is the phase where the project starts to take shape. The requirements are more or less frozen and the design of the system gets into its first draft.

**c) Construction** – This phase is the actual building phase of the project where the software takes shape and it involves the major coding part out of the four phases. Testing is also part of this phase. The first cut release of the software is the objective of the phase

**d) Transition** – This phase is the wrapping up of the system with the release to the client and the final support phase of the software system starts.

**RUP** also has six engineering disciplines and three supporting disciplines. These nine RUP disciplines are part of each iterations required in the project life cycle. The six engineering disciplines are somewhat

similar to the waterfall model phases whereas the three supporting disciplines are unique to the RUP framework.

### **I. RUP Engineering Disciplines**

- a) **Business Modeling** – Initial modeling of business with the analysis and scope formation
- b) **Requirements** – gathering and communication of requirements
- c) **Analysis and design** – analysis and formation of the solutions designs
- d) **Implementation** – Implementing the solution
- e) **Test** – testing the system as a whole and in units
- f) **Deployment** – finally deploying the system at client site and going into production level

### **II RUP Supporting Disciplines**

- a) **Configuration and Change Management** – configuration of the versions of documents and code. Management of changes to requirements, solution and codes as required
- b) **Project Management** – Planning, estimation, resourcing and overall managing the team and customers as part of the project
- c) **Environment** - Ensuring that the project team is aware of all aspects and of the RUP implementation.

### **Fig A – The Rational Unified Process Model**

As per the Fig A , the **RUP framework** is depicted as a “hump chart” with the matrix showing the RUP phases and the RUP disciplines. Lets say for example, the requirement discipline is peaked at the initiation and elaboration stage after which it flattens but does not completely disappear. Similarly project management discipline is required to be at a constant peak throughout the project and cannot flatten or lag behind in any phase.

**RUP** is one of the most complex software development project life cycles and involves proper project planning for the successful implementation of the system

#### **1. What is UML?**

**UML or Unified Modeling Language** is a modeling standard in the field of object oriented software systems. It has been standardized by OMG(Object Management Group) after being developed by Rational Corp(Booch Group). UML is a modeling language which puts together several diagrammatic views which can be used for any stage of the software development life cycle. UML was designed basically to provide a common platform for all the stakeholders in a project starting from the end users, analysts, designer, developers etc who are vital to the success of the project. So, in a way it cut down the miscommunication of the requirements and the display of the design in one common language which is understandable by everyone concerned.

But **UML** provides you with the syntax of the diagrams and views but does not actually give you the context in which it has to be used. Its up to the designer or analyst to find the best fit for the particular UML Diagram. The best thing about UML is that its code independent. As long as its object oriented programming, UML diagrams can fit into it.

**UML** provides us with the different diagram types which can be used in the design of an object oriented software systems. They are broadly classified into Structural Modeling diagrams(which depict the static structure of the system) and Behavioral Modeling diagrams(which depict the behavior and transactional movements of the system). Some of the most commonly used UML diagrams are:

- a) **Use Case Diagrams** – is a high level diagram depicting the system boundary and the interaction between the actors(users/external interfaces) and the system
- b) **Interaction Diagrams** – Shows how the different objects of the system interact with each other
- c) **Activity Diagrams** – shows the business process flow and makes use of the use cases similar to data flow diagrams



- d) **Class Diagrams** – depicts the properties and behaviors of the classes to be used in the system. Objects are instances of Classes and an Object diagram depicts the objects in a similar manner to the class diagram.
- e) **Sequence Diagrams** – are used to display the orderly sequence of message transfer between entities of the system
- f) **Component Diagrams** – shows the component types and their dependencies in the architecture of the system as a whole
- g) **Deployment Diagrams** – shows the physical architecture and the deployment components.

### **Fig A – UML Diagrams**

Out of these **UML diagrams** which are as shown in Fig A , Business Analysts worldwide would mostly use the Use Case Diagram, Activity Diagram and sometimes, Sequence and Class Diagrams. Apart from these , the majority of the rest of the UML diagrams are designed by the solution architect or designers. It is not essential that for any project all of the UML diagrams will have to be made. The UML diagrams are vital for a business analyst as they help him in getting the requirements validated and assessed. UML diagrams also add clarity to the functional specification documentation and hence are widely used by the Business Analysts to corroborate their requirement elicitation.

**Business Analyst** is a very prominent term in any successful organization. So the role of a business analyst is not simply understanding the business process but it extends its wings to other branches such as complete business process, understanding the nature of the business and its flow from bottom level to the top level. A business analyst has to take care of all aspects which can take the business to a top level and in a successful path.

A business analyst will make the new processes and these processes and these processes have to be designed in a detailed way so that the other layers of the business can implement these new processes. These business analyst will have to consult on regular basis with al the key roles in the business unit and understand the process how the business is running. The main key persons that a business analyst will have contact are management, ie top layer, finance department, HR department, production department, sales department, marketing department, and has to understand the customer mentality with regards to this business etc..

On a regular basis a business analyst will have to change or modify the existing business process in order to improve the business profits and as well as customer satisfactin..

#### **1. What is Class diagram**

Class diagram is an important one to know if you are a business analyst. Like use case diagrams, these class diagrams also explain the application in a pictorial representation, but in more technical way where a common user cannot understand by looking at these. A class diagram represents the relation between each class of the entire application. In one way it is a static representation of a structure of an application. It clearly tells the different classes and its attributes and relation between each class.

Normally for each member there will be few notations to tell more about the class member. These notations are placed before class. They are PUBLIC, PROTECTED, PRIVATE & PACKAGE.

In a nut shell, class diagrams can be defined as object oriented analysis and design documents. These class diagrams explain the relation between each class in a system and they properties, attributes, etc.. A typical class diagram will have three sections to represent each class.. In the below example I will explain one such class for your understanding.

#### **1. Business Analyst Roles and responsibilities**

**Business analyst** is a key person between business users(clients) and software engineers(typically technical & functional consultants & developers). They are the bridge between business users and technical people who will typically understand the business terminology & business language of the client and as well as technical language at the development place.

The essential duties of a **business analyst** is basically to gather the requirements of the client from business point of view and understand them thoroughly make a draft in MSWord (or some other form), and write use cases from that document in such a language that technical people will understand.

These use cases will be send to technical people from there they will make functional specification, technical specification, programming (coding, development) and finally testing of the product they made.

**Business analyst** will not sit idle until this process completes. In the entire process of development, technical persons will get some or many questions regarding the business and what ever the problems they get, will be reported back to business analyst and the business analyst will intern contact business users to discuss about the issue and understand the logic from business point of view and make a documentation and will give it to technical people.

This whole process will repeat until the product is completed or there are no questions from business understandability.

### **What is RUP (Rational Unified Modelling) ? with examples**

#### **Inception Phase:**

Interacting with business users , developers and project manager to make business requirements by using tools like MS Visio and MS word. MS excel also used for some documentation purpose.

Using RUP (rational unified modeling) to make BRD (business requirement document) specifications.

Performing requirement analysis and design work with rational rose and analyzed, documented the system specifications, business requirements and detail design of the existing business by using a the tool requisite pro for the requirement tracking and analysis.

#### **Elaboration Phase:**

In this phase created Activity diagrams, sequence diagrams and collaboration diagram etc by using Microsoft tool MS Visio.

Documented the critical path analysis and extensively analyzing the ER diagrams (entity relation diagrams).

Also finding out the different opportunities and parameters that can improve the business process and performance.

Finally coordinating the project scheduling with IT development manager.

#### **Construction Phase:**

Developed a prototype for actual system and developed project blue print. Also developed a mock web page generation for a part of the system goal.

Writing use case specification for the given business requirements.

Identifying the actual network and human resources to utilize properly for development, documentation and training purposes.

#### **Transition Phase:**

In this phase we concentrate in vendor software compatibility , data base integrities and other performance issues.

Successfully deploying the finished product.

Configuration, implementation and deploying the developed software in various cross platforms to see the products efficiency.

## 1. What is JAD session

### JAD session:

- 1 It brings together business area people (users) and IT (Information Technology) professionals in a highly focused workshop.
- 2 JAD participants typically include:
  - o Facilitator – facilitates discussions, enforces rules
  - o End users – 3 to 5, attend all sessions
  - o Developers – 2 or 3, question for clarity
  - o Tie Breaker – senior manager. Breaks end user ties, usually doesn't attend
  - o Observers – 2 or 3, do not speak
  - o Subject Matter Experts – limited number for understanding business & technology
- 3 Advantages:
  - o Shortening of the time.
  - o Improves the quality of the final product by focusing on the up-front portion of the development lifecycle.
  - o Reducing the likelihood of errors that are expensive to correct later on.

## 1. What is GAP analysis

### Gap Analysis:

- 1 The process of determining, documenting, and approving the variance between business requirements and system capabilities.
- 2 The process of determining and evaluating the variance or distance between two items' properties being compared.
- 3 The study of the differences between two different systems or applications, often for the purpose of determining how to get from one state to a new state.
- 4 A gap is sometimes spoken of as "the space between where we are and where we want to be." Gap analysis is undertaken as a means of bridging that space.

### 1. UAT (User Acceptance Testing):

- 1 Final phase in a software development process in which the software is given to the intended audience to be tested for functionality.
- 2 UAT is either done by making the software available for a free trial, typically over the Internet, or by using an in-house testing panel comprised of users who would be using the product in real-world applications.
- 3 UAT is done in order to get feedback from users to make any final adjustments to the programming before releasing the product to the general public.
- 4 UAT also is called **beta testing**, end-user testing or application testing.

Daily job duties of a business analyst:

Probably you must be wondering what exactly a business analyst is. After all those stereotypical definitions of a business analyst, you still must have some questions as to what exactly does a business analyst do and how does he fit in a firm. Well, this article will let you know how a typical day looks like in the life of a business analyst.

A business analyst needs to have knowledge of both the business and the IT of the firm. Business would include policies, processes, business models and infrastructure while the IT part involves the implementation of these. When firm is facing a strategic or organizational obstacle- that is where the business analyst jumps into action. A business analyst needs to have the right blend of both in order to carry out his tasks effectively. Hence a BA is required to have both the IT skills to understand the current implementation and also knowledge of business to remodel it.

When a BA is presented with a case, the following would be the typical steps that he would take to solve it:

1. Defining the problem
2. Staffing the team
3. Working with the client
4. Discoveries along the way
5. Impact

Defining the problem will include doing the initial analysis of the case to give it more clarity and to eliminate possible miscommunications between the BA and the client. This would require setup of meetings with stakeholders either for clarification or follow-up, getting hold of relevant documentation pertaining to the problem and documenting your analysis and findings. Making a business decision requires hard data along with wisdom and leadership. While defining the problem, the BA will require to couple quantitative analysis with strategic thinking in order to arrive at the way of approach to solution.

While staffing the team, one must remember that it is not necessary for the entire team to be present at one location. Teams could be divided by geographic location and time zones. In such cases, proper communication with all sub-teams forms an important part for everyone to stay in the same page. Another task of the BA would involve getting an heads-up from different teams in different time zones. Also while leaving for the day; they would have to inform the next set of duties for the team in a different geographic location. This show why being able to work in a team and proper communication is some of the required key skills in a BA.

A BA distills his discoveries into solutions creating an impact in the way the client organization will function from now on. A simple example of the role of the business analyst could be in the task of cost cutting. In the current economy, many organizations and firms would have implemented cost cutting in their business models in order to survive the bad times. A BA is the one who identifies the issues (in this case, the extravagant expenditures), forms various hypotheses and analyses, distill their conclusions into recommendations and present it to the client management who will incorporate that into their business. A BA could either work closely at the client side or at the parent organization, but in any case his job would be to get as much information about existing systems from the client, have brainstorming sessions with team to input ideas and help build effective solutions.

What is a Communication Diagram?

A UML 2.0 Communication Diagram models the objects or parts of a system, the interactions (or messages) between them, and the sequence in which these interactions occur. There are a lot of similarities between communication diagrams and sequence diagrams in terms of the information they show, but because of how each diagram presents the information, one diagram may be better at conveying or emphasizing specific information over the other.

Communication diagrams use a free-form arrangement of objects or parts of a system. This can be compared to how classes and objects are laid out in UML class and object diagrams. Then the interactions between the objects or parts of the system are show and labeled to indicate the chronological sequence in which they occur. The free-form arrangement of objects lends itself wekk to showing the sequenced interactions in a more compact space, allowing the analyst to place objects that have the highest number of interactions with each other near one another. This is the advantage of the communication diagram over the sequence diagram. While showing nearly all of the same information as a sequence diagram, the communication diagram can, at a glance, place a strong emphasize on which objects are interacting with one another

While communication diagrams are formally intended to show system objects and the interactions between them, many analysts choose to create them at a higher level of abstraction. Instead of showing

the interactions between objects of a system, larger parts of a system may be represented such as the interaction between web methods, web services, or entire systems.

By using the communication diagram in this way, it shows some similarities to a system context diagram. The primary differences between the two are that a system context diagram places a focus on a single system in context along with which actors and systems outside of the scope of the system interact with it. Additionally, a system context diagram does not show the sequence of interactions.

Describe the basic classification of requirements as defined by the BABOK (v2.0).

The requirements classification schema used by the BABOK (v2.0) places requirements in one of the following categories.

Business Requirements

Stakeholder Requirements

Solution Requirements

Functional Requirements

Non-Functional Requirements

Transition Requirements

Business Requirements define the goals and objectives of the business at the enterprise level. These are requirements that apply to the organization as a whole rather than a specific group within the organization. Business requirements are developed and documented as part of ongoing enterprise analysis activities. Business Requirements, or Enterprise Requirements as I prefer to call them, offer everyone within the organization a common understanding of why certain projects are initiated. They are a compass for the organization providing a clear direction that can be followed. While all requirements ideally should define something in measurable terms, this is even more important with business requirements. Therefore, business requirements should outline a corresponding metric and target that needs to be achieved by the business.

Stakeholder Requirements describe the goals and objectives of a particular group within an organization. Like Business Requirements, they are intended to provide a higher level direction for the stakeholder group, but often they are developed while considering the contending goals and objectives of other areas of the organization that interact with each other. So, the stakeholder requirements of various groups need to reflect an overall balance across the organization to support and achieve the Business Requirements in the best possible way. This tighter coupling and dependency between requirements causes Stakeholder Requirements to change more frequently. Stakeholder requirements do not define what needs to be supported by a particular solution (whether the solution is a business process or system solution), rather they span the gap between Business Requirements and more specific Solution Requirements.

Solution Requirements describe the various characteristics of a solution that must be met. The solution may be a process solution or a system solution. Solution requirements should be written in a way that they also support and align with the Stakeholder and Business Requirements. Solution requirements are defined throughout the requirements analysis process. They can be further classified into two sub-categories:

Functional Requirements describe the behavior and information that the solution will manage. In the case of a non-system solution, the behavior typically refers to a workflow and the information refers to the inputs and outputs of the workflow. Additionally, the requirements describe how the data will be transformed and by whom. In the case of a system solution, the functional requirements describe the features and functionality of the system as well as the information that will be created, edited, updated, and deleted by the system.

Non-functional Requirements describe the qualities of the process or system. Instead of describing what the solution must do non-functional requirements describe how well the solution must do something. Non-

functional requirements often describe qualities of a process or system such as its repeatability, usability, reliability, interoperability, scalability, extensibility, etc.

Transition Requirements describe any capabilities of the solution that aren't permanent but instead exist only to facilitate the transition from the current state to the future state. Once the process or system has been developed and the transition of users and information from the current solution to the new solution has occurred, these capabilities will no longer be needed or supported. Transition requirements cannot be developed until both the current state and the future solution have been defined.

What are the advantages and disadvantages of using screen mockups in the requirements gathering process of a system solution?

Screen mockups can support the requirements gathering process when introduced at the right time, but if introduced too early they can become problematic. Here are a few key points that an analyst should remember.

1) Mockups are nice because they help the business representatives or clients visualize the functionality of the system. This can be a big advantage to help analysts and stakeholders identify problems early on. However, if introduced too soon in the process the natural tendency is for the business reps/clients to try and be screen designers. Instead of stating that the system shall support "x", they begin saying that they need a dropdown to capture "y" and a button to do "z". The client is not a UI designer, in fact few business analysts truly are, so this can lead to a screen design which does not have an appropriate emphasis on usability. Similarly, specifying the controls needed on a screen detracts from the true requirements of the system and often results in an inadequate level of discussion around why a system must support certain functionality.

2) When requirements are captured in screen mockups with no supporting requirements list, it becomes impossible to know whether an early screen design decision was made because it supports a necessary requirement or if it was made for some other reason. How can the analyst and developers know whether they can eliminate or alter the screen feature without losing an important requirement. Questions like, "Do we really need to have the control on this screen, or can we capture the data at a later point in the process?" becomes unanswerable without going back to the original stakeholders. And, on complex projects no one stakeholder may be able to answer the question.

3) Screen mockups alone cannot capture the flow through the system. Often analysts will accompany screen mockups with a written description of what happens when certain buttons are clicked or when certain values are entered within a field or dropdown. These descriptions are helpful, but they fall short of describing the end to end processes that the system must support. Further document such as process flows or use cases are required, but often overlooked when too much emphasis is placed on screen mockups during the requirements gathering process. While analysts and stakeholders who are involved in the screen mockup process may have a basic understanding of the processes supported, developers and testers will not.

Ultimately, the introduction of UI mockups can be very helpful, but this should only occur after an exhaustive list of features and usage scenarios (what business process flows need to be supported by the system) have been documented. Only then can the UI mockups be generated without introducing major pitfalls.

What is a Context Diagram and what are the benefits of creating one?

The Context Diagram shows the system under consideration as a single high-level process and then shows the relationship that the system has with other external entities (systems, organizational groups, external data stores, etc.).

Another name for a Context Diagram is a Context-Level Data-Flow Diagram or a Level-0 Data Flow Diagram. Since a Context Diagram is a specialized version of Data-Flow Diagram, understanding a bit about Data-Flow Diagrams can be helpful.

A Data-Flow Diagram (DFD) is a graphical visualization of the movement of data through an information system. DFDs are one of the three essential components of the structured-systems analysis and design method (SSADM). A DFD is process centric and depicts 4 main components.

Processes (circle)

External Entities (rectangle)

Data Stores (two horizontal, parallel lines or sometimes and ellipse)

Data Flows (curved or straight line with arrowhead indicating flow direction)

Each DFD may show a number of processes with data flowing into and out of each process. If there is a need to show more detail within a particular process, the process is decomposed into a number of smaller processes in a lower level DFD. In this way, the Content Diagram or Context-Level DFD is labeled a "Level-0 DFD" while the next level of decomposition is labeled a "Level-1 DFD", the next is labeled a "Level-2 DFD", and so on.

Context Diagrams and Data-Flow Diagrams were created for systems analysis and design. But like many analysis tools they have been leveraged for other purposes. For example, they can also be leveraged to capture and communicate the interactions and flow of data between business processes. So, they don't have to be restricted to systems analysis.

A Context Diagram (and a DFD for that matter) provides no information about the timing, sequencing, or synchronization of processes such as which processes occur in sequence or in parallel. Therefore it should not be confused with a flowchart or process flow which can show these things.

Some of the benefits of a Context Diagram are:

Shows the scope and boundaries of a system at a glance including the other systems that interface with it  
No technical knowledge is assumed or required to understand the diagram

Easy to draw and amend due to its limited notation

Easy to expand by adding different levels of DFDs

Can benefit a wide audience including stakeholders, business analyst, data analysts, developers

When performing Cost-Benefit Analysis using discounted cash flows, how do you select and appropriate discount rate?

Cost-Benefit Analysis (CBA) is a critical activity in the work of a business analyst. While many business analysts may be brought onto a project well after this activity has occurred, nearly all projects which require a large amount of resources (time, people, or money) are assessed based on the CBA of the project. The activity is typically performed by a business analyst, often one specializing in the area of financial analysis, though any business analyst can learn this straightforward activity.

The most prominent and well known aspect of CBA is Discounted Cash Flow (DCF) analysis which discounts future cash flows (both negative and positive) using a Discount Rate to arrive at a Net Present Value (NPV). This is represented by the following equation.

$$NPV = (FV_{pos} - FV_{neg}) / [(1 + i)^t]$$

where,

NPV = Net Present Value

FV<sub>pos</sub> = Future Value of a Positive Cash Flow at "t" years

FV<sub>neg</sub> = Future Value of a Negative Cash Flow at "t" years

i = Discount Rate

t = time (years from present)

Future cash flows are discounted using a Discount Rate (i) that is chosen to accurately reflect several factors:

Time Preference – The theory that a person or institution would rather have money in hand now to spend on immediate wants or needs rather than waiting for future cash flows.

Interest – Accounts for the fact that a person or institution that doesn't receive money for several years also loses the opportunity to gain interest on the money for that period of time.

Risk Premium – Reflects the additional return that a person or institution requires on later cash flows to account for the risk of future payments not materializing.

While the Discount Rate used for DCF calculations will vary, it becomes clear that the discount rate should be larger than any single factor above. If the interest that can be attained on the money that is being tied up is %5 per year, the discount rate should certainly be higher than 5% to account for Time Preference and the Risk Premium.

What are the contents that go into the object model and domain model during GAP analysis. how are the AS-IS and To-BE system documentation prepared

When performing analysis between an existing system (AS-IS) and its desired future state (TO-BE) you could take into account all aspects/dimensions.

Having said that, I realize that this approach may not always be practical nor necessary. So you should probably determine the focus of your gap analysis by answering the following questions:

- What are they key characteristics of the system?
- What do you know about the existing system?

For example, if your system is workflow centric then it would probably be very beneficial to focus on the differences in workflow and document them by creating a workflow diagram which focuses on the differences.

On the other hand - if the only documentation you have of the existing system is a list of requirements then you should start by identifying he requirement gaps.

So getting back to the domain model....

If you are doing a Domain Model gap analysis focus on and document the following:

- Differences in business entities (new entities, entities no longer needed, etc.)
- Differences in entity relationships (diffs in types of relationships, diffs in multiplicities, etc.)
- Differences in attributes for a given entity (new attributes, attributes no longer needed, etc.)
- Differences in methods/behavior

How do you follow business rules in a project while working on a project?

Business rules are the key in defining any business process. example credit card company wants to prepare the process and the key business rule is - check applicant's credit score and income. the process will need to include these two sub-process to complete the main process and without with it will through an exception. you can use the use case diagram or process diagram to include this information in your Use case document or Functional Requirement Document.

What is the difference between high-level and low-level use case? How business analysts can performs that job?

High-level Use Case Diagram gives you snap shot for your system and contains many process within it.

Low-level Use Case Diagram will describe one process in details with alternative flow, exception and include/exclude descriptions.

What roles business analysts play during change management?



The Business Analyst will check the requested change against the original requirement. If its part of the requirement and need to fix it as a defect, it will be done through defect management. If its part of the new requirement (which is not mentioned in Business Requirement Document), he/she need to analyze the impact of change in terms of feasibility, schedule, resource planning and project scope. The detailed discussion and sign off required if the change will have major impact on the project

How would you assess your value as a BA?

This question is not intended to determine how much you are looking to make. If an interviewer asks you a question like this, they are likely looking for answers to a number of other unspoken questions such as: Do you understand the real value a business analyst brings to an organization?

Do you ever think about the cost associated with employing you as a business analyst?

Do you have the skillset required to be a marketable business analyst?

Are your expectations of the value of a business analyst realistic?

Companies don't hire business analysts for fun; they hire them to save money. It's that simple. So how does a business analyst save an organization money? You might mention a few examples such as: Make a process more efficient saving the company time and resources (which translates as money) Drive out the real requirements of a system, instead of a half-baked solution, ultimately reducing the amount of rework and re-development required to develop a system that delivers the intended value (rework means lost money)

Identify opportunities for increased customer satisfaction leading to greater customer retention and greater new customer conversions (more money)

If you have quantifiable examples of work that you have produced in the past and know precisely how much your work saved a company (not the work of the entire team, but YOUR actual contributions) this information can be very powerful. This is your true value as an analyst within a similar organization and role. But few people have the information required to make this kind of assessment. In addition, the value you bring to an organization is very different from your "potential value". If an organization has you writing specs for a system, your opportunity to bring value may be much lower than if you are re-engineering a multi-million dollar business process to eliminate hundreds of thousands or even millions of dollars of waste.

This line of reasoning leads us to the question "Do you have the skillset required to be a marketable business analyst"? You may have expert knowledge of traditional SDLCs and be able to create complex analysis diagrams, but if the organizations that are hiring all require you to work in an Agile environment then what value can you bring them? Even though your potential value for some organizations may be quite high, your value to others that use a different range of skills may be quite low. This example shows how your value is dependent upon the industry environment and the tools, competencies, and methodologies that are popular at that time. It also shows the need to keep your skillset current and then stress those skills that are most relevant to the interviewing organization.

If you can talk through these concepts with an interviewer, then you will have demonstrated that you don't take your value as a business analyst for granted and that you are the type of person that will maximize your value within the organization that hires you.

2. How would you influence people when you do not have decision making authority?

Few business analysts have the final authority to make critical decisions on projects. That's why it's so important for business analysts to polish their influencing skills.

The process used to influence people can be a formal, well thought out presentation, or it can be an informal conversation. Either way, it never hurts to think through a standard framework by which to structure your thoughts before attempting to influence someone. The following is a concise 5-step framework that can be used for both formal and informal communications that involve influencing another person or audience

Define the What, Why, and Who

Prepare your case

Deliver your message

Obtain a commitment

Agree to a specific action plan

These 5 steps provide a framework to structure and plan your communication to maximize your influence over a person or audience, but it's the details of each step that will determine how influential you will be.

1. Define the What, Why, and Who.

It's important for you to have a well define and thoroughly understood objective. What is your goal or objective? Why are you championing your particular position? Who do you need to influence?

Answering these questions will focus your case.

2. Prepare your case.

Notice that I didn't say "prepare your argument". Influencing is very different from coercion or even selling. While there may be some selling involved, it should be a soft sell of the benefits that you are espousing.

Consider how you can customize your case for the person or audience. What does the person or audience value? What do you have to offer the person or audience? Do you have specific technical knowledge? Do you have a strong network which could benefit the person or audience?

For more formal communications, while you are preparing your case you should outline a number of potential options for the action plan that might be used if you get the person or audience to commit.

3. Deliver your message.

When delivering your message be direct in your thoughts and language. You want to come across as respectful and open to the feedback of the audience, but do so without obscuring the point of your message. Don't hint at what you want, but also don't demand.

Use powerful questions to engage your audience. Open-ended questions that lead the audience to realize the advantages of your case work best.

4. Obtain a commitment. Whenever possible, it is best to obtain a commitment immediately following the delivery of the message while the benefits of your case are still fresh in the audience's mind. Steer the conversation to help your audience arrive at the stage where they are comfortable committing.

5. Agree to a specific action plan.

Even with a commitment, you are only truly successful once you've realized your objective. That's why agreeing to an action plan is so important.

Develop milestones for reaching the final objective

Identify resources and assign roles and responsibilities

Develop a method for tracking progress

To re-iterate, if you are planning a formal communication or presentation you will have a lot more time to spend thinking through the details of this 5-step process. However, even for brief communications within a short conference call, mentally thinking through these steps for just a few second can help guide your conversation and increase your degree of influence with your audience.

3. What is meant by Agile? Describe:

Agile is a general term and conceptual framework used to describe a number of "light-weight" methodologies, such as Extreme Programming (XP), SCRUM, and Rapid Application Development (RAD), which exhibit a series of common characteristics. Some of these characteristics include:

Iterative analysis and development

Time-boxed iterations of a predefined length

Delivery of the most critical features and functions first

Delivery of a complete build with an initial set of limited features within a few months (often 1-2 months)

Small cross-functional teams usually of 6-9 team members.

Daily team communication meetings

#### Reduced levels of documentation

Most Agile methods begin with a prioritized feature list where features are grouped together into deliverable chunks and assigned to a particular iteration in which they will be developed and delivered. Using small teams and daily communication among all team members the Agile team can achieve a high level of efficiency.

Agile methods are intended to overcome or circumvent many of the recurring challenges that are encountered during software development projects. The iterative nature of these methods, along with the desire to deliver smaller sets of defined features per iteration, help mitigate risk due to evolving requirements, unclear project stakeholder direction, and unforeseen project complexities that typically arise during the latter stages of analysis and development. Some of the most salient advantages of Agile methods include:

Availability of working software much sooner which allows for more immediate feedback from application users.

More immediate, and therefore larger, Return on Investment from software features that are developed in short iterations and release to production immediately.

Less project overhead due to smaller team sizes.

Avoidance of large schedule overruns.

Avoidance of large budget overruns.

#### 4. What is Scrum Method?

Scrum is one of several light-weight agile methods that use an iterative and incremental approach for the development of information systems. The Scrum method brings a small team together to work on a specified set of features over a period of 30-days (called a sprint).

Both the term Scrum and sprint are borrowed from the sport Rugby. A scrum is where the two teams are engaged in a huddle to begin play following a period where play has been stopped. The fast moving period of play from the point of the scrum until play ends again is called a sprint.

The Scrum method starts each 30-day sprint with a kickoff meeting (a period where the entire team comes together). The kickoff meeting lasts a full day and the features of the system to be developed are discussed. The outcome of the kickoff meeting is a set of features that will be developed over the 30-day sprint along with estimates of how long the analysis and development of each feature will take.

In order for a feature to be considered completed, it needs to be Analyzed, Designed, Coded, Tested, Refactored, and Documented. If this life-cycle is not fully accomplished during the 30-day sprint, perhaps due to an initial underestimation of the time required, the feature will be pushed to a later sprint.

Following the kickoff meeting, and throughout the duration of the 30-day sprint, each day is started with a short meeting lasting approximately 15 minutes called a daily scrum meeting (also called a daily stand-up meeting). The purpose of this meeting is for the team to discuss what they accomplished the day before, what they will accomplish over the coming day, and to raise any obstacles that they have encountered that may impede progress.

One aspect of Scrum, that is intended to keep the Scrum team and method very agile, is its size. Most Scrum teams consist of no more than about 7 people with each falling into 1 of 3 roles.

Product Owner – identifies the features that will be included in the next 30-sprint and set the priority of each. This is typically a high-level stakeholder in organizations where a true Product Manager/Product Owner role doesn't exist.

Scrum Master – acts much like the project manager. While the Scrum Master does not micro-manage the team's deliverables, this person ensures that the 30-day sprint is on track and enforces the key rules that guide Scrum such as; no new features can be added to the sprint once it is kicked off, and team members cannot be pulled off to work on other side project in the middle of a sprint.

Team Member – unlike traditional software development methods, in Scrum there is little separation of duties between team members. Each team member may fill the role of analyst, designer, coder, tester, and documentation writer.

5. Please describe the similarities and differences between UML and Object Diagram.

Class Diagrams and Object Diagrams use almost identical notations. Both represent a static view of a system, however Object Diagrams represent a snapshot in time, whereas Class Diagrams are not time dependent and are an abstract view of the types of objects that may exist within a system, the relationships between them, and how and when one type of object can exist in relationship to another. Since Object Diagrams represent specific object instances that exist at a single point in time, they are sometimes called Instance Diagrams.

A few notable differences in the notation and rules used to represent Class and Object Diagrams are: Class Diagrams represent each class via a rectangle and display up to 3 types of information; the class name (not underlined), its attributes, and its operations. The Object Diagram also uses a rectangle to represent an object instance, however the object name is underlined and lists the name of the object followed by a colon and then the class name which describes its type (e.g. Joe: Student, where Joe is the name of the object instance and Student is class). Additionally the Object Diagram will list an object's attributes, but it will also list the value of that attribute at that point in time (e.g. SSN = 555-55-5555). Class Diagrams enforce multiplicity rules between associated classes. For example, a Class Diagram may display an association between a Car and Passengers. The Class Diagram would show a single rectangle to represent the class car and a single rectangle to represent the class passenger, but display multiplicities stating that each car may have 1..4 passengers. An Object Diagram being a snapshot in time would show a rectangle for the Car and up to 4 separate rectangles, one for each passenger that exists at that movement in time.

Many of the constraints or association types that exist in a class diagram have no relevance in an object diagram. Multiplicities in a class diagram may constrain the number of passengers in a car to 4, but this rule would be enforced within the code itself such that a 5th passenger object could never be created. Therefore, multiplicities are not shown within an object diagram. Only the actual numbers of objects that exist at that moment in time are shown. Similarly, other constraints such as "at least one passenger be of type driver" could also be captured and displayed in a class diagram but would not be shown in an object diagram since these rules are enforced within the actual code.