Business Analyst Interview Q & A

Project Based - Technical - Behavioral

Business Analyst Interview

Questions at the Beginning

Opening questions will be asked at the beginning of an interview. These questions set the stage for more detailed discussions and have most profound impact in the decision making process. Generally are subtle in nature but very important.

BEST PRACTICES THAT MUST BE FOLLOWED

(1) Make it look like a conversation rather than Q&A session and be at ease. If telephonic interview, make sure there is no background noise.
(2) Don't go overboard with your response.
(3) Don't BRAG about yourself.
(4) Don't get into technical aspects of it. Keep it concise and precise.
(5) Don't counter question in this phase. You will get opportunity to ask any questions at the end of the interview.
(6) If it's face-to-face interview, have a calm and composed posture. Don't complain or raise any alarm regarding traffic, commute etc. Keep your body gestures in check and have an eye contact while speaking.
(7) Always support your claim with some examples.

1. Please tell me about yourself? Please introduce yourself? In few sentences, please give me a snapshot of your career.

Ans - Upon completing my MBA, I started my career at <Company Name> as a <Role > and subsequently moved to different sectors in marketing/business development area before starting my career in Information technology.

I started by journey in IT field as a Quality Analysts and gained valuable insights of SDLC processes before I began my career in the field of 'business analysis/requirements management' at an entry level position and subsequently moved into more senior roles such as Sr. business analyst, lead business analyst and into my current role of 'Business Analyst Manager'. During more than <10> years of experience, I have worked in diverse domain, (finance, banking, and ecom.), and various SDLC environments like waterfall to agile. During my course of <10> years of experience, I have also used various tools and techniques to manage the requirements. For example: I have used MS Suite (Word, Powerpoint and Excel), MS Visio (to model various sorts of diagrams), Rational Suite of products (to model UML diagrams), JIRA, TRELLO for project/requirements management and many more.

My last project was with [Client Name] where I worked as 'Business Analyst Manager' and my key responsibilities were (A) Support planning, monitoring and controlling of requirement activities, (B) to hands-on elicit, analyze, document, validate, prioritize and manage the requirements across the project lifecycle (C) as well as lead a team of high performance business analysts (tasks assignment, mentoring BAs, daily status check)

TIPS - The following structure should be followed:

(1) Education background if required

(2) A brief timeline from when you began working (irrespective in which field and position)

(3) Emphasize on the most recent project and what your role was on that project.

(4) Snapshot of your expertise and core strengths

- Worked in various domains.
- Worked in different SDLC environments.
- Used various tools during the course of your career

2. Please tell me about your role and responsibilities in general?

ANS - In most of the project, my day-to-day job consists of the following:

(1)Communicating with the concerned project stakeholders in order to understand project vision, scope and their business needs.

(2) Eliciting business requirements from the business stakeholders and aligning these business requirements with the solution requirements (System level requirements). Since my last couple of projects were in agile environment, I would collaborate with the 'Product Owner' and conduct 'Product Backlog Grooming' sessions to understand their business needs and stakeholder requirements, priorities and translate them in the form of user stories.

(3) Doing analysis of the requirements using various tools and techniques such as business process modeling, context diagrams, Unified Modeling Language (UML).

(4) Documenting the requirements in different forms (Vision document, Business Requirements Document, Functional Requirements Document, Business Process Narratives, Use Cases Specifications, Supplementary Specification Document, Requirements Traceability Matrix (RTM) etc.

(4) Communication requirements across diverse group of stakeholders.

(5) And finally, managing the requirements across the project lifecycle (updating the requirements, version controlling)

3. What have been the key highlights of your career?

ANS - During my [XXX] years of BA career, I would say:

(1) I have successfully transformed form QA to BA and delivered high-quality business/system requirements in some really challenging environments.

(2) Successfully worked with diverse group of stakeholders.

(3) Was highly productive working in different SDLC environments (Ranging from pure waterfall, in-house methodology, highly agile such as scrum).
(4) Was really fortunate to get exposed to so many different business domains (list all the domains you have put on your resume).
(5) Gained tremendous knowledge (via training programs, from coworkers, self-learning etc). I believe there is always room for improvement and there are always good qualities we can learn from others.
(6) Mentored and coached many of my coworkers in the business analysis discipline.
TIPS - Give some examples for 'Challenging Environments
(1) In [Project Name], we had to do lot of reverse engineering since there was no SME and neither any documentation to understand the current system functionalities. The project timelines were very tight and aggressive. The entire team put lot of effort to first understand the legacy application, it's features/functionalities, architecture and based on that we transformed the entire application into new framework.
(2) In [Project Name], there were multiple group of stakeholders giving requirements. Each stakeholder groups were giving requirements in silos, which led to lot of conflicting requirements, prioritization issues, political battle etc. We suggested to have a steering group, and have a impact analysis from the business perspective before analyzing the system requirements. So me, along with other stakeholders, would meet every week to do the impact analysis on the requirements coming from multiple business portfolios before going forward.

4. How many years of experience you have in the field of business analysis and what was your involvement?

ANS - Though I have [XX] years of experience in all, I have [XX] years of experience working as a 'Business Analyst'. During my [XX] years of experience, I have worked in diverse domain, (finance, banking), and various SDLC environments like waterfall, to agile. During my course of [XX} years of experience, I have also used various tools and techniques to elicit, analyze and document the requirements. For example: I have used MS Suite (Word, Powerpoint and Excel), MS Visio (to model various sorts of diagrams), Rational Suite of products (to model UML diagrams), JIRA, TRELLO for requirements management and many more.
My last project was with [Client Name] where I worked as 'Business Analyst Manager' in highly agile environment. My key responsibilities were (A) Support planning, monitoring and controlling of requirement activities, (B) to hands-on elicit, analyze, document, validate and manage the requirements across the project lifecycle (C) as well as manage a team of business analysts (tasks assignment, mentoring BAs, weekly status check) [ONLY IF YOU ARE LOOKING FOR A 'LEAD BUSINESS ANALYST' POSITION]. (D) Conduct 'Product Backlog Grooming' sessions along with the 'Product Owner' and identify user stories for the subsequent iterations/sprint. (E) Help the 'Product Owner' as well as 'Implementation' team to plan, prioritize and estimate user stories (F) Provide

requirements support to the 'Implementation' team during the sprint/iteration execution and finally (G) Conduct 'Sprint Review' and 'Sprint Retrospective' meetings at the end of each iteration.

5. What makes you leave your current job?

ANS - I am looking for a long-term relationship and hence want to get away from a consulting world and short-term (temporary assignments). I am really interested in developing a long term relationship with my employer. Some examples of reasons you might give for leaving your current or previous job and moving to the new job. Below are some of the reasons:
(1) The job position aligns perfectly with your career goals. For example: you career goals can be (a) building new skills/competencies in the area of 'Business Analysis' , (b) taking more challenging work and (c) contribute towards the development of business analysis discipline.
(2) Your job doesn't offer room for advancement. (You've gone as far as you can, despite having done well in your position.)
(3) You want to expand into areas that don't exist within your current job (but do exist with the one you're interviewing for). For example: Want to work in a particular domain.
(4) The new company is one you've always wanted to work for and when you saw the opening you were excited to have a chance to make this career move. For example: You always had a desire to work in IT company if you are being interviewed by <TCS>.
(5) The new company is growing in a direction / areas that you want to be part of.
(6) You're looking for a chance to take on a higher level of responsibility.
(7) The new job offers better working conditions, benefits, commuting, etc. (This isn't always as strong as some of the others on its own, but if combined with one other reason can make a very good case).
(8) You just completed some classes or training in an area you want to pursue and the new company gives you the opportunity to use and grow these skills.
(9) You have researched the new company and feel that it's a much better fit for you and your current career goals. (You would need to have a solid reason(s) for that conclusion.)
TIPS - You want to look for a reason that feels true enough to you to sound totally sincere in the interview. And you want to make sure that the new job actually provides you the opportunity you say are looking for – something that the old job doesn't offer! You want to add value and make a difference to the future company. You should offer something that they need.

6. Please tell us different environments you have worked in?

ANS - During the course of my [XXX] years of career as 'Business Analyst', I have worked in different environments like waterfall to highly agile. I have also worked in a hybrid sort of environment where things were done using waterfall but in iterations and incrementally. For example: my first project [mention the project] was a waterfall where we implemented each phase

in a structured and step-wise fashion. In one of the project, [LIST THE NAME OF THE PROJECT], it was 'In-House' SDLC methodology of our client, it was a mix of waterfall and agile and my most recent project was in a highly agile environment with Scrum framework.

7. What sorts of tools have you used in your career so far?

ANS - Please provide the names of all the tools you have mentioned in your resume with regards to requirements. Also mention the reason for using the tool. For e.g. "I used MS Visio to map out the 'As-Is' business and 'To-Be' business processes'
(1) MS Visio to capture the 'As-Is' and 'To-Be' state of business processes.
(2) MS Suite (Word, PPT and Excel) for documentation, presentation and data analysis sort of work.
(3) Rational Software Architect tool and MS Visio to do UML (Unified Modeling Language) models.
(4) JIRA, TRELLO, StarTeam for requirements management.
(7) Balsamiq for developing wireframes
(8) MS Project to build project plan (schedules)
(9) MS Excel and Tableau for Data analysis and visualization

8. Describe a typical day at your most recent job?

ANS - On a day-to-day basis, my job is to communicate with the stakeholders (external as well as internal), plan and conduct requirements meetings with the customers/SMEs, resolve any issues pertaining to the project (requirements related or sometimes even non-requirements related), and support implementation team from the requirements perspective.
(a) Work very closely with the product owner to groom the product backlog, write user stories and help the product owner to prioritize them.
(b) Analyze the user stories using any suitable techniques (such as business process diagrams, use case models, context diagrams, data flow diagrams, data dictionary etc)
(b) Facilitate the sprint planning meetings with the product owner and the entire scrum team to help them understand the user stories, it's acceptance criteria. Also, assist the team in estimating the user stories (in story points) and finally scope the sprint.
(c) Support the scrum team, during the execution of the sprint, from the requirements perspective (if they have any questions or concerns)
(d) Conduct scrum ceremonies such as sprint planning meetings, sprint review, daily scrum and sprint retrospective meetings.

9. What are your career goals?

ANS - I have set some long term as well as short term goals for myself. As part of my long term goals, I plan to advance my career in the field of 'Business Analysis' and take it to the next level by learning the industry best practices, latest tools & techniques, constantly acquiring knowledge and

apply all these towards delivering quality requirements. Hence, helping projects meeting their goals and objectives and able to meet the business needs of the clients. As part of my short term goals, I will be taking CBAP (Certified Business Analysis Professional) certification and (a) enhancing my BPMN (Business Process Model and Notation) skills by the end of <Year> or (b) enhancing my knowledge on agile practices such as Test Driven Development (FDD), Extreme Programming (XP) etc.

Project Based

1. **Please describe your most recent project.**

ANS - My most recent project was with [CLIENT NAME] wherein I worked as a 'Sr. Business Analyst' supporting 'ABC' project. To provide some background about the project, the client had experienced a lot of issues in the recent years. The scope of the project was to conduct business process re-engineering on the client's core processes, capture data in a formatted manner wherein Team can review, evaluate and negotiate the pricing and technical data in the most effective manner. My role in this project was to elicit, analyze, document and manage the requirements across the lifecycle of the project.

TIPS - Your answer should be in the following structure:
(1) You client name, project name and in which capacity your supported.
(2) Brief background about the project. Key issues, pain points, bottlenecks faced by your client.
(3) What was the scope of the project.
(4) What was the project environment (SDLC)
(5) What were your day-to-day activities.

2. What were some of the challenges you faced in your projects?

ANS - I have seen different challenges in different projects and sometimes in the same project as well. For example, in [PROJECT NAME[, we have stakeholder management as an issue. We had multiple stakeholders who were providing their requirements. But often times, the requirements would conflict and there was no one authority on the business side who would assist us to decide which requirements takes priority. This would often result in a deadlock situation where neither of them would budge. We proposed to set up a 'Review Board' where business portfolios would iron out such issues. We also started having working 'Requirements Workshop' sessions, where all the stakeholders were invited and requirements were done in a live session as compared to doing it in silos.
In another project, [PROJECT NAME], we observed that many team members (Developers, testers) were accustomed to seeing big set of documentation such as Functional Requirements Document (FRD), Business Requirements Document (BRD), System Use Cases etc etc. Since this project was supposed to be done in an agile environment, we had to conduct numerous training sessions with the team to help them explain the fundamental differences between the waterfall and an agile methodology, how the requirements are elicited, and documented using user stories as compared to FRDs or use cases.

3. **What kind of SDLC environments have your worked in before? Can you give some overview?**

ANS - I have worked in varying environments, ranging like waterfall to an highly agile. For example, my [PROJECT NAME] was in pure 'Waterfall' environment. Whereas, my [PROJECT NAME] was agile (SCRUM).

TIPS - Mention all the projects that are in your resume. Make sure this is in sync with your resume. For e.g. if you have in your resume only 2 projects as agile, stick to that. SAY SOMETHING THAT CAN BE JUSTIFIED.

4. **Since you have worked in traditional as well as agile environments, which one do you prefer, Traditional waterfall or Agile?**

ANS - I would say the SDLC model should be adopted based on the nature of the project. I have seen many projects jumping into the 'Bandwagon' without understanding the big picture and the 'Pros' and 'Cons' of waterfall/agile and later failing to deliver. For example, if the business stakeholders are themselves not clear of the requirements and what their needs are, it's worth spending some time in 'Requirements Analysis' instead of straight away starting with an 'Iteration'. This will avoid lot of rework. It's important that somebody has a 'Big Picture' (Vision) of the project crisp and clear, laid out in black and white. In one project [PROJECT NAME], client had laid out very stringent milestones for each phases. Also, client had a fixed budget in which we had to deliver. So in that case we had to adopt 'Waterfall-like' SDLC model (Though we built in 'iterative' and 'incremental) elements into it.

DON'T be selective. Convey message that you are flexible for both.

Technical Questions

1. How does the Business Analyst role change on an Agile project compared to projects using other software development methodologies?

ANS - The role of the BA should actually change very little between different software development methodologies, although the tools and techniques used by the BA can vary according to the needs and attributes of any given project or development lifecycle. The core responsibilities of a BA on a software development project include requirements elicitation, requirements analysis and requirements management – regardless of the project methodology. The type and format of requirements documentation are just tools, and a good BA has a wide range of tools at his or her disposal.

(A) Accurate and effective elicitation of requirements from stakeholders is one key part of the BA role on any software project. The BA is responsible for ensuring that requirements are clearly articulated, resolving inconsistencies and ambiguities, and synthesizing individual requirements into a unified solution. An Agile project may utilize specific tools and techniques for collecting and documenting requirements, but the elicitation role still exists on an Agile project as it does on any other project type.

(B) Analysis of requirements is a second key part of the BA role on any software project. The BA is responsible for addressing gaps and conflicts within requirements, identifying and coordinating inter-dependencies and relationships between different requirements, and ensuring that requirements fit seamlessly together to produce the envisioned solution. This analysis role is equally applicable whether requirements are documented as user stories, use cases or a functional requirements document.

(C) The third key part of the BA role on a software project is requirements management. The BA is responsible for ensuring that requirements remain linked to business value and business outcomes, tracing and overseeing requirements from initial elicitation through to final delivery, and preserving the integrity of the business solution from project start to finish. This role is necessary whether the project is Agile, iterative, waterfall or anything in between.

2. What are some of the core responsibilities of a Business Analyst?

ANS - The core responsibilities of a BA on a software development project include requirements elicitation, requirements analysis and requirements management – regardless of the project methodology. The type and format of requirements documentation are just tools, and a good BA has a wide range of tools at his or her disposal. Accurate and effective elicitation of requirements from stakeholders is one key part of the BA role on any software project. The BA is responsible for ensuring that requirements are clearly articulated, resolving inconsistencies and ambiguities, and synthesizing individual requirements into a unified solution.

Analysis of requirements is a second key part of the BA role on any software project. The BA is responsible for addressing gaps and conflicts within requirements, identifying and coordinating inter-dependencies and relationships between different requirements, and ensuring that requirements fit seamlessly together to produce the envisioned solution. This analysis role is equally applicable whether requirements are documented as user stories, use cases or a functional requirements document.
The third key part of the BA role on a software project is requirements management. The BA is responsible for ensuring that requirements remain linked to business value and business outcomes, tracing and overseeing requirements from initial elicitation through to final delivery, and preserving the integrity of the business solution from project start to finish. This role is necessary whether the project is Agile, iterative, waterfall or anything in between. These tasks all require certain expertise, skills and techniques that have been developed, promoted and refined under the Business Analysis profession. Even if these tasks are assigned to any other project member from developer to product owner, that person is still fulfilling the Business Analyst role.

3. As a newly hired Business Analyst, what steps would you take to familiarize yourself with your role in the organization and to prepare for your assigned project(s)?

ANS - In addition to any formal new hire procedures, there are a number of things that a Business Analyst can do to acclimate to a new organization, and prepare to tackle new project assignments.

Company protocols – in addition to basic administration and office protocols, find out about policies and procedures that affect what you do as a BA and how you do it, such as:

Information security policies – how is information classified within the organization, what information are you authorized to collect, what procedures need to be followed when disclosing confidential or protected information, what information can and cannot be sent through email, etc. Does the company have a clean desk policy where everything has to be put away or locked up before you leave?

Record retention policies – what types of information are classified as transitory records (that can be shredded or deleted when no longer being used), and what types of information have to be kept as official records?

Communication protocols – there will be more informal protocols than formal ones in this category, for most organizations. Find out which audiences you can / should contact directly, and what forms of communication they prefer. Find out which types of communications need more formal controls, such as review or approval of messages before they are sent. Some organizations have more formal controls on communications to external stakeholders, for example.

Project protocols – Find out how time is to be tracked and reported on your project, and get familiar with the project governance structure (who reports to whom, and who is responsible for what). Find out where documents are to be stored and which documents need version history. Ask if there are established protocols for contacting and engaging stakeholders, and confirm any boundaries for your communications with stakeholders. Become familiar with all project protocols for risk reporting and management, issue reporting and management, change control, quality assurance and deliverable approvals.

Project information – Learn as much as you can about any projects assigned to you. Read any existing project documentation, and ask if there is additional background or reference material that might be useful for your project. Learn as much as you can about project stakeholders, their needs and interests, and their work-styles as this will help you engage them more effectively. Find out about any constraints or competing priorities that might impact their availability for participation on your project. Also find out if key stakeholders have experience with this type of project, as this is an important factor in effective engagement and communications.

Business analysis standards – find out about any standards (formal or informal) for business analysis techniques, tools and deliverable. If possible, look at similar deliverable produced by other projects and find out what worked well – and what didn't.

Relationship building – begin establishing relationships and rapport with stakeholders, project team members, organization team members (if different from your project team) and your BA peers. Strong relationships are key to successful business analysis, and the strongest relationships combine both formal and informal interactions.

4. What are a few characteristics of a top business analyst?

ANS - 1) They deeply understand the business problem being solved.
Most business analysts end up solving the wrong problem. Well, maybe not the wrong problem but not quite the right problem. Often analyst might be solving a related problem but haven't really identified the root problem. Top business analyst have the ability to properly frame and structure complex business problems such that they are more readily understood by the entire team and the root problem is the one being solved. They also repeatedly question who, what, when, where, why and how to ensure that they right problem is fully revealed.

2) They are excellent translators and negotiators.
Business analysts are constantly interacting with different types of team members from business people with almost no detailed knowledge of technology to the most skilled developers. One group

speaks the language of business and the other speaks the language of IT. Rarely the two groups are able to smoothly communicate with one another. The business analyst is the translator between the two. Top business analysts understand both languages and are excellent at translating between these two groups.
Business analysts must also be adept negotiators. Different business stakeholders will often have competing needs. Similarly, those things that are important to the IT group might not be what is important to the business stakeholders, and vice verse. Business analysts must be able to explain the positions of each group and negotiate a common plan and direction to move forward.

3) They must be able to view the project from the highest to the lowest levels.
As the project carries on, the business analyst will get involved at an increasingly detailed level. The team will be making hundreds, if not thousands, of decisions about the features and behavior of the application. This is the critical point when the business analyst must be able to view the project from a high-level as well. The best business analysts have the ability to keep all involved parties focused on the big picture and longer term benefits of the software even when in the midst of detail and fast approaching deadlines.

4) They understand technology and its limits.
Business stakeholders rarely have enough technical knowledge to know what capabilities exist that will best solve their problem. Similarly, they often don't realize that technology decisions impact each other. Once one specific architectural decision is made, other technical capabilities may no longer be supported. Top business analysts will have enough technical understanding to be able to advise the business stakeholders of what can and can't be done when it comes to software solutions.
In addition, some technical solutions while possible are just too costly to make sense. The business analyst should be able to communicate software decisions in terms of TCO (Total Cost of Ownership) and ROI (Return on Investment) and direct the business stakeholders to make the most cost effective decision.

5) They have credibility with the business.
While business analysts can come from either a business background or an IT background they must have credibility with the business stakeholders. Usually, this means that strong business analysts first start out on the business side and slowly develop an interest in the technical side over time. Since they already have a detailed understanding of the business processes and business problems that are being automated and solved the have almost immediate credibility with the business stakeholders.

It’s not impossible for an IT person to become a business analyst, but convincing people on the business side of the organization that an IT person has a strong understanding of what they do is a difficult feat. Additionally, many IT workers have difficulties translating the IT speak into something that the business stakeholders can comprehend.

6) They enjoy interacting with people and are excellent communicators.
Communication and collaboration are vital skills for any business analyst. The best business analyst are adept communicators and prefer interacting with team members rather than keeping to themselves in their cubicles or office. Being an active communicator minimizes the chance that information doesn’t get fully disseminated across the team and it increases team cohesion. Additionally, the superb communication skills of a strong business analyst ensures that the true requirements of the software are understood and conveyed to the development team accordingly.

5. What is your understanding on Scrum?

ANS - Scrum is one of several light-weight agile framework that uses 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 usually 2 to 4 weeks (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 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 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, Re-factored, and Documented. If this life-cycle is not fully accomplished during the 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 sprint, each day is started with a short meeting lasting approximately 15 minute 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 sprint and set the priority of each. This is typically a high-level stakeholder in organizations where a true Product Manger/Product Owner role doesn't exist.
Scrum Master – acts much like the project manager. While the Scrum Master does not micro-manage the teams deliverable, this person ensures that the 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.

6. What sort of existing documents should Business Analysts refer to when starting on a new project?

ANS - Few analysts are brought on to a project at the very beginning. For those that are, they will often have a hand in creating some of the important documents that other analysts should reference when they first join.

First, get your hands on the project charter. The project charter, while high level, will provide critical information on the project such as:

The reasons for undertaking the project, including the high level business goal or goals that are to be satisfied by the project and a calculation of Return on Investment (ROI), objectives and sub-goals of the project as well as major constraints due to current business processes or existing technology infrastructure,
the high level vision and scope of the project outlining the initial direction for the solution being developed, major risks which need to be avoided while developing the solution, the important stakeholders involved which should include not only a project sponsor and steering committee members but also the business representatives that will have final sign-off on requirements.

Find out as much as you can about the project management processes that are being used to manage time lines, risks, communications, costs, etc. Ideally, these processes are outlined in a formal document. If not, be sure to talk to the project manager(s) on your project to fully understand the processes that should be followed.

The same goes for the analysis process and artifacts being used. Understand the methods that will be used for eliciting and documenting requirements. How will these requirements be communicated? How will they be captured and translated into functional specifications? Being an

analyst, this is something that you will be taught at some point on the project, but the sooner you learn the details of the analysis process and artifacts being used the better off you will be.

If requirements elicitation has already begun, review the existing requirements documentation. Reviewing requirements will bring you up to speed rapidly. Record any questions you have regarding the system requirements and get answers to them. If a prototype has been created as a method for identifying and clarifying requirements, understand the prototype inside and out. Take the time to understand why each screen was designed a particular way. Was it to support a business requirement, or was it merely a design decision that could have been handled in a different way.

7. What approach should a business analyst take when gathering requirements from high level executives versus the end users?

ANS - For the most part, the methods and techniques for gathering requirements are the same regardless of whom the stakeholders are: interviews, focus groups, questionnaires, requirements workshops. However, there are definite things that the business analyst should consider when gathering requirements from high-level executives vs. the end users. The business analyst should tailor his approach and expectations based on the stakeholder type.
For example:
Requirement Types:
For High-Level Executives, generally provide vision and high-level guidance on the direction of the project. They address key features and major problems to be addressed by the software.
For End Users, can provide detailed needs and requirements as related to their specific day-to-day jobs.
Usability:
For High-Level Executives, are less focused on detailed usability issues but may be interested in general UI and usability guidelines.
For End Users, are more interested in the usability of the system as they are the power-users. They want the system to be easy to use and efficient (ex: minimize keystrokes and clicks).
Budget/Cost:
For High-Level Executives, Are very concerned with the cost and budget of the system therefore will be more likely to prioritize requirements and focus on the ones with higher ROI.
For End Users, Less concerned with cost, therefore might provide all kinds of "nice to have" suggestions which may not solve real business problems.:
Requirements Workshop:
For High-Level Executives, Effective method of gathering requirements from this group. It is desirable to have all the decision-making stakeholders in the room when identifying the requirement rankings and when making key design decisions.

For End Users, Effective method of gathering requirements from this group.
It is a good idea to have a number of workshops one for each user type (ex: advanced users vs. most users vs. new/inexperienced user).

8. How do you gain a sufficient understanding of an industry in which you previously had no experience?

ANS - To gain a quick yet sufficient understanding of a new industry the business analyst must choose tools and techniques that generate a high-level, overarching view of the space with the least amount of time investment possible.

The first, and possibly most effective, tool for this purpose is the business entity model. The business entity model (or logical data model) documents the “things” or “nouns” that the business uses or interacts with in order to accomplish their work. Examples might be a bond, a loan, a borrower, a stock, a savings account, etc.

After the entities of the business are identified, attributes which describe each entity are also identified. So a “Borrower” may have attributes such as Name, SSN, Date of Birth, etc. In this way, the attributes further refine our understanding of the "things" within our domain.

Finally, the relationships between entities are identified. For example, a Borrower can have a Loan, and a Loan can have Collateral.

As attributes and relationships are identified for each entity, the business analyst and the rest of the project team can quickly understand the details of the business domain regardless of the level of previous domain knowledge. While a business entity model will continue to be updated and refined over time, creating a basic business entity model early on does not require a great deal of time.

Another great tool is the context diagram. The Context Diagram (a specialized version of a data flow diagram) shows a 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.) So it maps the domain in terms of processes, the data used by those processes, and the movement of data throughout an organization. In this way, the Context Diagram works hand-in-hand with the Business Entity Model using the information already documented.

9. What is the role of the Business Analyst (BA), at each point in a typical project or programme of work?

ANS - Typically the role of the BA starts even before a project is conceptualized and initiated. For example:

(1) Working with the Customers and business stakeholders, BA gets involved with performing enterprise analysis, finding the root cause of the issues, finding alternative solutions to solve the problem (Cost benefit analysis, financial tools such as Return on Investment, Net Rate of Return etc, and defines the measure of success and target state.

(2) As part of the strategy formulation process, BA helps to conduct market research, assist in creating a strategy, and provide any sort of support to the management.

(3) Once project is initiated, BA supports the PM help develop project charter, identify the stakeholders of the project.

(4) During the project planning phase, BA's role is mainly to assist the PM by formulating 'Requirements Management Plan', and most importantly perform the core requirements tasks (requirements elicitation, analysis, documenting and communicating with the stakeholders).

(5) During the project executing phase, BA's role is mainly of a supportive one. He/she must address any requirements-related issues/concerns.

(6) As part of monitoring and controlling phase, BA must ensure the integrity of the requirements, manage the requirements and ensure that there is no scope creep taking place.

(7) Finally, as part of project close out phase, BA must actively participate in all the activities such as lessons learned (to improve the next projects), store all the project-related artifacts as per project standards set by the PM.

10. How do you differentiate between yourself and a systems analyst?

ANS - While the differences between the business analyst (BA) and systems analyst (SA) vary from organization to organization, there are some acceptable differences:

Role and Responsibilities

BA: The business analyst's main role is to understand the business processes and procedures (how the business works), to identify areas of improvement (problem areas), and to work with the business stakeholders to identify suitable solutions. If the solution involves the development or modification of information systems, the business analyst is also responsible to gather, analyze, and document the business requirements which must be fulfilled by the software.

SA: The systems analyst's main role is to understand the business requirements (documented by the business analyst) and to design and document the functional and, often, technical implementation of the software/IT system. The systems analyst also helps solve problems and issues with existing systems through research of these systems, code reviews, data analysis, etc.

Skills and Competencies

BA: Great communication and negotiation skills, vertical business/domain knowledge, ability to map and document business processes, requirements elicitation and documentation.

SA: Strong technical skills, excellent troubleshooting and debugging abilities, solid technical writing skills, system modeling/documentation skills (class diagrams, sequence diagrams, etc.), data analysis and modeling skills.

Summary

BA: Talks the business talk; understands the business.

SA: Talks the IT talk; understands the technology.

The reality is that the business analyst and systems analyst roles often overlap. Many analysts who perform both roles have the title of "Business Systems Analyst.

11. How do you differentiate between a Business Analyst and a Technical Writer?

ANS - The role of a Business Analyst is a broad and encompassing role with many different specializations. Business Analysts may specialize in the areas of Business Process Analysis, Systems Analysis, Requirements Engineer, Data Analyst, Functional Architect, Product Manager, Usability/User Experience Analyst, and at times Technical Writer. However, while a Business Analyst may perform the role of a technical writer at times the profession of technical writer can stand on its own. A technical writer's role is a collaborative and interactive one. Their primary deliverable are:

Technical manuals that describe the specific features of a product or application

Producing online step-by-step tutorials with illustrative graphics and images to aid the reader

Producing web-based training and other forms of training materials

To produce these deliverable the technical writer must acquire a detailed knowledge and understanding of the product or application for which they are producing the deliverable. This requires them to work with analysts and developers of the system, end users of the systems, and often to test certain features of the system themselves

12. What is Active Listening and how can it benefit the business analyst?

ANS - Active Listening is a method used to listen and respond to others in a structured and deliberate way. It requires a listener to understand and actively evaluate what he or she heard. Actively listening can be used to achieve a number of goals.

One of the more common goals of actively listening is to ensure that the listener accurately understands what the speaker has said by replying back to the speaker and paraphrasing what they believe they have just heard ("So, if I understood you correctly…"). The speaker can either acknowledge that the listener's understanding was accurate or can quickly identify any misunderstanding that the listener may have. Actively listening helps the listener avoid incorrect

conclusions due to unintentional assumptions that the listener may have made. It's important to note that a listener that employs active listening is not necessarily agreeing with the speaker.

Another goal of actively listening is for the listener to extract additional information from the speaker. While listening to the speaker, the listener may notice something in the speaker's tone or body language. By responding to the speaker with phrases such as "you seem to feel …" the speaker has the opportunity to confirm or correct the listener's understanding. This is a non-confrontational approach to asking follow-up questions which clarify the speaker's intent.

Active Listening can be a powerful tool for business analysts during requirements elicitation. Requirements elicitation often occurs during a period of a project where not everyone has the same background knowledge and understanding of the project. Because of this, there are typically many assumptions that are being made by each person as they build a framework in their mind of the project and its problems and challenges. Actively listening can verify correct assumptions and dismiss false ones resulting in a clearer and more accurate set of requirements.

13. What does the term 'Problem Domain' mean to a business analyst?

ANS - In broad terms, the Problem Domain describes the area undergoing analysis. The scope of the problem domain needs to be identified upfront by the business analyst. The size and scope of the problem domain can vary greatly depending on the goals of the project being undertaken. The scope may align with the boundaries of an entire organization or it may be much more granular, aligning with a single organizational unit, a specific business process, or a particular system.

Even when the scope of the problem domain aligns with the boundaries of a particular group or system it may also include stakeholders outside of the process or organizational group such as customers, suppliers, or any other stakeholder which provides an input or accepts an output of a process, organization, or system.

In short, the Problem Domain is anything and everything that is needed to define the area under analysis, fully understand the inputs and outputs of its processes, and achieve the goals of the area under analysis, but nothing more.

14. What are the four fundamental methods of requirement verification?

ANS - The four fundamental methods of verification are Inspection, Demonstration, Test, and Analysis. The four methods are somewhat hierarchical in nature, as each verifies requirements of a product or system with increasing rigor. I will provide a description of each with two brief examples of how each could be used to verify the requirements for a car and a software application.

Inspection is the nondestructive examination of a product or system using one or more of the five senses (visual, auditory, olfactory, tactile, taste). It may include simple physical manipulation and measurements.

Car: visually examine the car to ensure that it has power windows, power adjustable seats, air conditioning, a navigation system, a tow package, etc.
Software Application: visually examine the software for screens that were requested, check for the fields needed for data entry, verify that the necessary buttons exist for initiating required functionality, etc.
Demonstration is the manipulation of the product or system as it is intended to be used to verify that the results are as planned or expected.

Car: use the automatic switches to verify that the windows and seats work as intended, start the vehicle and ensure that the air conditioning produces cold air, take the car for a test drive to sense the acceleration and cornering as it was described based on the requirements.
Software Application: enter all required fields on a screen and select the button to return a specific report. Ensure that the report is returned with the type of data needed.
Test is the verification of a product or system using a controlled and predefined series of inputs, data, or stimuli to ensure that the product or system will produce a very specific and predefined output as specified by the requirements.

Car: accelerate the car from a complete stop to 60 mph, and verify that it can be done in 5.2 seconds. Accelerate through a turn under controlled conditions, producing .8G of force, without the car loosing traction.
Software Application: enter the type and model of car, automatic windows, power steering, and all other options as stated in the predefined test plan, select the price now button and receive back a price quote of precisely $43,690.
Analysis is the verification of a product or system using models, calculations and testing equipment. Analysis allows someone to make predictive statements about the typical performance of a product or system based on the confirmed test results of a sample set or by combining the outcome of individual tests to conclude something new about the product or system. It is often used to predict the breaking point or failure of a product or system by using nondestructive tests to extrapolate the failure point.

Car: complete a series of tests which rev the engine at a specific rpm for a set length of time, while monitoring engine vibration and temperature, to verify that the expected results are achieve. Use this information to model the failure point of the engine, i.e. max rpm sustained over a specific period of time.

Software Application: complete a series of tests in which a specified number of users input the characteristics of the car they are attempting to price and initiate the pricing functionality at the same time. Measure the response of the system to ensure that the pricing function returns its results within the time specified. Analyze the relationship between increasing number of system users and the time it takes for pricing to be returned. Record the results to capture system degradation. Use this information to predict at what point the system no longer meets the maximum allowable time to return pricing as defined by the requirements.

15. **How do you ensure that you've identified all of the necessary project stakeholders?**

ANS - Most Business Analysts know that identifying project stakeholders is important. But how do you identify them and know with any degree of certainty that you haven't missed someone important? After all, if you overlook a key stakeholder then the project is likely to suffer later due to scope change or a lack of acceptance by the business users.

Different techniques can be applied to identify stakeholders. First, start with some context diagram. This is an excellent way to identify entities involved with the project. Second, prepare a 'Stakeholder Chart' (Onion Diagram) and 'RACI' matrix, listing down each stakeholder and his/her role and responsibilities. Third, project management artifacts such as (a) Project Charter (b) Project Management Plan (PMP) (c) Project Scope Statement.

While just an example, the following questions can be used to help uncover a stakeholder that might have been overlooked since different groups within the organization may have a different perspectives.

Who is requesting the project deliverable?
Who is the champion of the project?
Who controls the budget for the project?
Who is currently responsible for the business process that may be impacted by the project?
Who is currently responsible for the systems that may be impacted or replaced by the project?
Whose job is directly impacted by the delivery of the project?

16. What is the difference between a business use case and a system use case?

ANS - A business use case describes a customer, business worker, or other interested party and how they interact with the business in order to achieve an expected outcome. While the business process defined within the business use case may be supported by systems that automate some tasks, business use cases make no mention of systems. It's focus it on the business and it's processes or workflow.

A system use case describes how an actor (customer, business worker, or even another system) interacts with a specific system. While the system use case will describe step by step interactions it still should make no mention of specific screen constructs such as screens, buttons, or drop downs. The system use case should merely define the information passed into the system, what the system does with it including interfacing with other human or system actors, and the resulting information or state.

17. **What is an epic, and what purpose does it serve on an Agile project?**

ANS - Business epics are large, typically cross-cutting initiatives that encapsulate new development needed to realize certain business benefits. One or more epics can form the basis for the business case that justifies and initiates a project.
Epics are often used as placeholders for new ideas that have not been thought out fully, or whose full elaboration has been deferred until actually needed. Epics are then evolved into split into multiple user stories that help Agile development teams effectively manage and groom their product backlog.

18. **How does your role as a Business Analyst changed on an Agile project compared to projects using other software development methodologies?**

ANS - The role of the BA should actually change very little between different software development methodologies, although the tools and techniques used by the BA can vary according to the needs and attributes of any given project or development lifecycle.

The core responsibilities of a BA on a software development project include requirements elicitation, requirements analysis and requirements management – regardless of the project methodology. The type and format of requirements documentation are just tools, and a good BA has a wide range of tools at his or her disposal.

Accurate and effective elicitation of requirements from stakeholders is one key part of the BA role on any software project. The BA is responsible for ensuring that requirements are clearly articulated, resolving inconsistencies and ambiguities, and synthesizing individual requirements into a unified solution. An Agile project may utilize specific tools and techniques for collecting and documenting requirements, but the elicitation role still exists on an Agile project as it does on any other project type.

Analysis of requirements is a second key part of the BA role on any software project. The BA is responsible for addressing gaps and conflicts within requirements, identifying and coordinating inter-dependencies and relationships between different requirements, and ensuring that requirements fit seamlessly together to produce the envisioned solution. This analysis role is

equally applicable whether requirements are documented as user stories, use cases or a functional requirements document.

The third key part of the BA role on a software project is requirements management. The BA is responsible for ensuring that requirements remain linked to business value and business outcomes, tracing and overseeing requirements from initial elicitation through to final delivery, and preserving the integrity of the business solution from project start to finish. This role is necessary whether the project is Agile, iterative, waterfall or anything in between.

These tasks all require certain expertise, skills and techniques that have been developed, promoted and refined under the Business Analysis profession. Even if these tasks are assigned to any other project member from developer to product owner, that person is still fulfilling the Business Analyst role.

19. How are personas used in requirements elicitation on agile projects?

ANS - A persona is a description of a fictional person, representing key characteristics of a specific user class or group. Personas provide models of the user community that make it easier for project teams to understand and predict how users will act and react, based on characteristics such as age, education, knowledge, and attitudes.

A key purpose that personas serve on agile projects is in the formulation of user stories, so that each story is written from the perspective of a distinct person with known attributes – rather than a generic "user" or job role.

Personas are also invaluable in making design decisions, and determining how usability can be optimized for the user community.

As with all other aspects of Agile projects, personas should be developed collaboratively amongst the team and refined iteratively over the course of the project.

20. Should user stories be created to plan for system maintenance and support?

ANS - This question is likely rooted in a misunderstanding of what agile development is all about. Agile project teams are intended to be assembled for specific projects. Once a certain amount of product planning, backlog planning, and sprint planning takes place, the first sprint kicks off. From that point on, the goal of the agile team is to deliver incremental user value at the end of each sprint and to maximize the velocity at which the team delivers such value (Velocity is just a fancy term for the number of Story Points completed during a sprint, and Story Points are the unit of measure used to size user stories relative to each other).

None of this means that the velocity incorporates all costs associated with developing the features defined by the user stories. Overhead still exists within agile teams. In fact, story points don't track absolute hours at all. They are intended to show the relative size and complexity of user stories for prioritization and sprint planning purposes.

Typically Project Management and Product Management time is lumped into overhead costs. One major reason for this is that much of the work done by these roles cannot map one to one with user stories. There is a lot of project management work and product management work that occurs which never results in value to the user. For example, a product manager may document or lead the effort to create a series of user stories that, once estimated, end up being prioritized out of existence. Perhaps the decision is made that there isn't enough value in the user stories to have them developed. So this is one example where the time spent by people who work on or with the agile project team isn't tracked by a user story.

So back to our original question, should distinct user stories be created for maintenance and support. The short answer is no. Once a system or application is developed, there will continue to be hardware and infrastructure management that needs to occur. Someone needs to monitor server loads, bring new virtual servers online when system load rises to a point where server degradation begins to occur, etc.

The slightly longer answer is that while user stories should never be created solely for system support or maintenance, support and maintenance tasks can sometimes be sized into new user stories when it makes sense. If it's known in advance that a feature defined within a user story is going to require a significant amount of infrastructure to support the new functionality, then the time require to set up and configure new servers should be included in the user story estimate. Of course, it may make sense to distribute this time over several user stories that require the new back end resources. So, really it depends on whether the work is a result of the user story itself.

It's also worth noting that refactoring is a natural and necessary part of agile development. Refactoring describes the act of restructuring the internal structure of existing code without impacting the external functions of the system as viewed by the user. Refactoring may be needed to improve non-functional aspects of a system such as response times, or refactoring may be required to support a new feature that can't be supported with the current architectural structure. In agile projects, refactoring is expected and should be planned for. The incremental cost of refactoring should be spread across the impacted user stories whenever possible. User stories contain acceptance tests that need to be retested once the refactoring occurs to ensure that the existing functionality hasn't been broken.

21. How do you explain Planning Poker in Agile?

ANS - Planning Poker is a variation of the Wideband Delphi Method, a consensus-based technique for estimating effort. It is primarily used in agile development to estimate the size of user stories, but can be equally as powerful for estimating effort of key tasks in a project plan using a traditional work breakdown structure.

In Planning Poker, each person involved in the estimation process will submit an estimate with all estimators revealing their estimates at the same time. This is one of the primary benefits of the Planning Poker technique -- it avoids anchoring. Anchoring describes the behavior that occurs when one estimator knows the estimate given by another. If each estimator overhears that one of the others was planning to estimate a feature at 3 story-points this could have an impact on the other estimates, even if only subconsciously.

Planning Poker got its name from the use of playing cards within the estimation process. It's intended to add a fun spin to the estimation process. While it began with a simple deck of playing cards, other variations of card decks have been created over time.

Using a standard deck of cards, each estimator is given an Ace, 2, 3, 5, 8, and King. These correspond to either the story points needed to complete the development of a particular user story (agile) or the estimate of hours or days required to complete a task (traditional work breakdown structure). A King represents 13, or in some cases, the need to break a user story into smaller user stories. Units of effort (story points, hours, days) are determined ahead of time.

Here are the steps used in the Planning Poker process. It assumes that the estimation process is occurring with everyone present in the same location, but with minor changes the process can be performed in a distributed manner.
A moderator chairs the meeting to ensure that each member adheres to the guidelines
A product manager or similar team leader provides an overview of the feature set or user story that is to be estimated. The team can ask clarifying questions about assumptions and risks. However, there should be no mention of numbers that might indicate the specific amount of effort a feature will take in order to avoid anchoring. A summary of the discussion is recorded.
Each person involved in the estimation process will determine how much effort is required and lay a card face down in front of them reflecting that estimate. Then everyone turns their cards over simultaneously.
Those who have significantly higher or lower estimates are given the chance to justify the estimate.
The process repeats until all estimates converge around the same value. A timer can be used to ensure that each round of estimation doesn't take too long.

22. How do you measure Velocity of an agile project?

ANS - The Velocity of an agile project describe the speed, in story points, at which user stories are completed from sprint to sprint. This is the ratio of completed story points over the duration of the sprint.

Velocity = Completed Story Points / Sprint

Example) If a team completes 150 story points over the course of 3 sprints then the average Velocity = 150/3, or 50 Story Points/Sprint.

Pretty straight forward. So how is velocity useful?

Calculating the team's velocity after the first sprint provides the information necessary to assign user stories to future sprints, as well as estimate the time required to complete all of the sprints required to deliver the desired functionality. The team velocity can also be used as a starting benchmark from which improvements in team efficiency can be made. This may be by removing friction within processes or team dynamics, or by eliminating or improving on any number of other things which slow the rate of progress.

23. What are User Stories?

ANS - Extreme Programming (XP), one of many Agile methods, introduced the practice of User Stories to describe what a system or piece of software should do. User stories have since been adopted by many of the agile methods used today.

User Stories are short descriptions of functionality that will be valuable to a user or purchaser of the software or application. They describe the users' goals when using the system. The initial descriptions can be written by the users, customers, product managers, or developers, and are just a few sentences at most (1-3 sentences being typical). This isn't the entire user story, but it is all that is created at first.

The development of user stories occurs in three parts; the Card, the Conversation, and the Confirmation.

The Card: Named for the standard index cards on which a user story is often captured, Cards include the brief description of the user story, its relative size to other user stories (called story points), and the priority of the functionality. The cards are used for planning the work that will be completed during each iteration of development. If the size of the user story gets too big to

complete within a single iteration then it should be broken into smaller stories. The term used to describe a user story which needs to be further broken down into smaller stories is an "Epic".
The Conversation: While the conversation itself is not an actual deliverable, it is a critical step in the user story development process. Discussions about each user story are had with the users/customers of the system to flesh out details. The details of the conversations are documented in the form of acceptance tests called "The Confirmation".
The Confirmation: Acceptance tests are details which are captured from the Conversation that can be used to verify that the user story has been successfully implemented. When index cards are used, the acceptance tests are typically written on the back of the card itself. Acceptance tests can and should be captured whenever they are thought of, however, at the beginning of each iteration there is a defined period of time which is set aside to generate acceptance tests.
Using these three parts, the goal of the user story is to plan which functionality will be developed during each iteration, provide enough detail that a developer pretty much understands what needs to be coded, and provide a means to verify that they have achieved the goal. If the developer needs more details, more conversations are had, the details of which are documented as more acceptance tests.

Here are some sample user stories (the Card) for a job board:

I want to post a resume
I want to search for a job
I want to electronically submit my resume for jobs I like

24. What are Story Points and how did you use them?

ANS - Story points are a unit of measure used to estimate the relative size and complexity of user stories in agile development. If one user story is 1 point and another is 2 points then the 2 point user story is expected to take twice as much effort to develop as the first.

Sample Product Backlog Estimates

User Story #1 - 4 points
User Story #2 - 1 points
User Story #3 - 2 points
User Story #4 - 2 points
User Story #5 - 8 points
So, why not just estimate in hours? The answer is in how the human mind deals with numbers.

There are two primary examples to consider. First, we don't estimate things to take 26 or 27 hours. Instead, we say a task might take 24 hours because we convert this in our minds to 3 work-days. Or we might even give an estimate of 20 hours which makes us strain just a bit harder and think in terms of half work-days, but we are willing to do the mental conversion because we feel its worth the additional granularity. Second, if I ask you to tell me whether a specific task will take 26 hours or 27 hours, you probably couldn't reliable say either way. A difference of 2 hours when compared to 26 feels negligible.

These two examples are reflections of two challenges:

1) Anything over about 8 hours of effort causes us to begin doing mental conversions to internalize the amount of time involved.

2) When numbers grow in size we have difficulties having conviction in our estimates. How can you make a strong argument to management that a task, feature, or user story can't be done in 48 hours, that it requires 56?

Story points solve these problems by normalizing estimates around a unit of 1 story point. So, the smallest user stories may take 1 story point of effort to complete while harder ones may take more (2, 4, 8, etc.)

This immediately solves our first problem, since I no longer have to convert estimates during the estimation process. I'm making relative comparisons of overall complexity and effort. The human mind lives in the domain of relative comparisons. So, it's much easier for our brains to determine that one user story is about twice as big as another than to say one will take 16 hours and another 32 hours. So much easier, in fact, that some companies have found that by estimating projects in terms of story points their time spent on project estimation has reduced by 80%.

Story points also solve our second problem of both defending and having conviction in our estimates. Since we are dealing with story points, as estimates get larger we don't worried about maintaining the same level of granularity as we might if dealing with hours. We don't size user stories in half story points. In addition, management is less likely to argue that a particular user story should be 2 points versus 4 points. However, if they are reviewing estimates in hours, all too often they will argue than 80 hours "seems to long" for a particular piece of work.

25. How can the acronym INVEST assist the analyst during the development of user stories?

ANS - INVEST is an acronym that can help a Product Manager or Developer create quality user stories. INVEST stands for Independent, Negotiable, Valuable, Estimable, Sized-Appropriately, Testable.

I - Independent: The user story should be self-contained if at all possible to avoid dependencies on other user stories. Since one characteristic of agile methodologies is the ability to be flexible and re-prioritize what's important, independent user stories allow for flexibility during iteration planning. If you do find that your user stories are dependent upon one another, you may be able to combine smaller user stories together that have a dependency between one another. Similarly, you can divide larger dependent user stories into smaller stories such that one of the new smaller stories contains and isolates the overlapping portion of the larger stories.

N - Negotiable: User stories can always be changed or rewritten up until the point of coding. This further supports the flexibility associated with agile methodologies. Since requirements often evolve or rise and fall in priority, user stories should be able to adapt with the changing requirements.

V - Valuable: A user story represents a goal of an end user or purchaser and should deliver functionality that is deemed valuable. This means that specifics of the technical design are not something that you would document as user stories. However, some technical requirements have a component which is valuable to a user. A user might expect pages to load within 2 seconds. The user story would specify the need for 2 second page load times while the specifics of the physical implementation of this would be left out.

E - Estimable: You should always be able to estimate the size of a user story. Sometimes, developers won't have the experience required to size a particular situation or needed for a user story. When this occurs the user story can be split into two separate user stories. The first is a "spike" which is where developers do some quick research to determine the feasibility of something or get a better idea of how long it might take to implement the particular feature. The spike is always time-boxed, meaning it is limited to a pre-defined amount of time. The "spike" user story might be named "Research (something) to determine…)", while the second user story is where the functionality will actually be delivered. These two user stories should be scheduled into two separate iterations such than the spike can be completed and the feasibility of the second user story assessed before coding begins. This gives the team time to react if problems arise from the spike.

S - Sized Appropriately: User stories shouldn't be too big or too small. So how do you decide what size is right. First, any user story that can't be completed by a developer within a single iteration (or by a developer pair when paired programming is being used) is too big. The user story should be subdivided into two or more smaller stories. Similarly, there is no need to make user stories too granular just for the sake of decomposing features. If features group well together and complement each other then it makes sense to make a single user story. For instance, "As a job seeker I want to be able to add, delete, and edit a job skill on my electronic resume so that I can maintain an accurate listing of my skills." There is no reason to split "add, delete, and edit" into multiple user stories unless one of them creates a significant amount of work that would make the user story too large for the iteration.

T - Testable: User stories must be testable in order to ensure that development is complete and has been done correctly. So when are user stories not-testable? Often, if the analyst isn't careful, non-functionality requirements are written in a manner which is not testable. Consider the example, "pages should always load quickly". There are two not testable components of this statement; "always" and "quickly". A testable statement would be "pages should load within 1.5 seconds 97% of the time".

26. How do you describe the lifecyle of a user story?

ANS - User Stories are used by agile methodologies to capture the functionality that a system or software should support. For details about what a user story is and how to write one reference What are User Stories.

At the beginning of a project user stories are identified and developed in a story writing workshop – a brainstorming session in which the agile team comes up with as many user stories as possible (the agile team commonly includes some combination of customers, product manager, developers, testes, etc.). Each story is sized and prioritized for the first time. This prioritized list creates the Product Backlog (Release Backlog). During the workshop the team selects an iteration length (usually between 1 and 4 weeks) and also the rate at which they will be able to complete user stories (call the velocity), both of which become important when determining how to schedule user stories later in the process.

During the iteration planning process, the user stories contained in the product/release backlog are segmented into iterations or sprints. The user stories for the first sprint make up the sprint backlog.

Once the first iteration is ready to kickoff, conversations begin for each user story between members of the agile team. The user stories get updated with details of the conversation, captured in the form of acceptance tests.

The user stories can be updated at any point as needed up until the iteration coding has been completed.

If a user story cannot be completed during the iteration for some unforeseen reason, it is returned back to the product backlog and rescheduled for another iteration.

Once the user story has been coded, typically the user story is discarded. It does not live on beyond the development process.

27. How do you explain Burndown Chart?

ANS - A Burndown Chart is a tool used by multiple software engineering methods to track the progress of work completed. It compares the amount of work remaining (typically measured along the vertical axis) against time (measured along the horizontal axis). The amount of work remaining can be measured in whatever way works best for the project, i.e., work-hours, work-days, story points, or any other work unit. Similarly the time axis can be measured using a variety of units, the most common being days or iterations. The burndown chart gives a quick view of the amount of work that is completed over time.

When applied to Agile methods such as Scrum, this tool can be used to track progress at the Sprint level (a specific iteration of development) or at the release level (multiple iterations that deliver the total functionality for a product release). After the amount of work completed has been measured over several units of time, the burndown chart can be used to forecast the completion of an overall release or project.

28. What are some key characteristics of good set of Requirements Interview questions?

ANS - In order to walk away from the requirements interview with good requirements, the questions asked need to be properly thought out and strike the right balance. Here are a few key characteristics that help the analyst produce a set of quality requirement interview questions.

Some questions need to focus on the as-is process.
Other questions should ask about the major pain points in the process or what might be missing.
Questions should be a balance of open-ended and close-ended questions. Close ended questions may not yield as much breadth of understanding, but too many open ended questions can make it difficult for the interviewee to give that analyst structured or relevant feedback (i.e., avoid too many questions that start with “Why”).

Ensure that the questions are not leading questions which imply an answer; this can negatively influence the information that the analyst receives.
Keep it brief. The primary list of questions should be only 7-10 high level.
Each primary question may have a number of more specific follow up questions.

29. **What is a Vision Document?**

ANS - The Vision

Let's discuss first what is considered the "Vision". In general, the Vision represents the end user's or customer's ideas and views of the software product to be developed. Think of the vision as the "idea". In an entrepreneurial environment, the entrepreneur starts with an "idea" which is later turned into a tangible product, service, etc.

Same is the case with software projects; they all start with an idea or vision related to the types of needs that might be addressed by a system having certain features.

The Vision Document

Many organizations capture the Vision in a "Vision Document" which generally contains the key business needs and features of the system from the stakeholder perspective. The Vision Document is simply a mechanism to put down on paper "the idea".

There are many different types of templates for a Vision Document depending on the methodology, organization, project size, etc.

The Business Case

Yes, the Business Case is different than the Vision. Whereas the Vision represents the "idea", the Business Case is the "rationalization" as to why the Vision is a good idea and how the Vision will be implemented. The Business Case begins the discussion, at the high level, on how the project might be implemented: cost, resources, timeliness, etc.

30. Are use cases the functional requirements or do you think functional requirements are different from use cases?

ANS - It is generally accepted that use cases, specified in narrative form (also known as use case specifications), depict functional requirements. This is because a use case, via the main and alternate flows, shows how a user interacts with a system in order to achieve a desired result. That's exactly the purpose of a "functional requirement" to describe the functions and behaviors

that a system is or should be capable of. Therefore, if use cases are used and narrated in detail for a project, there is no need for separate documentation to describe the functional requirements because the totality of all the use cases represent the set of functional requirements for a given system/project.

31. What are the 5 basic categories of elements in BPMN?

ANS - BPMN is a robust notation designed to balance two competing needs. The notation should be simple enough for all stakeholders to understand, yet robust enough to handle complex orchestration of events to a level of detail which can be made executable. Not an easy thing to do. However, by organizing elements into distinct categories, a sizable notation can be more easily understood.

BPMN organizes elements into 5 basic categories:

Flow Objects (Activities, Events, and Gateways)
Data Objects
Connecting Objects (Sequence flows, Message flows and Associations)
Swimlanes (Pool and Swimlanes)
Artifacts (Text annotations and Grouping).

Behavioral - Situational Questions

1. **Why should we hire you? / Please tell me how your skills and expertise can help us?**

ANS - Focus on what strengths you bring to the table. These should be consistent with the four things most employers are looking for in candidates during the job interview: competence, professionalism, enthusiasm, and likability. Remember, they are looking for chemistry between you and them. Be prepared to summarize in 60 seconds why you are the best candidate for the job. Also, let the employer know you want the job and you will enjoy working with them. A lack of interest in the job may indicate a lack of enthusiasm for the job and them.

I see myself as a good fit for the following reasons:
(1) One of the key reason is that I have worked in a similar role. For example, in my [PROJECT], as a 'Sr Business Analyst', my job responsibilities included [LIST DOWN ALL THE RESPONSIBILITIES LISTED IN THE JOB DESCRIPTION] which is required for this position.
(2) I can greatly contribute towards taking the requirements area to next level. For example, in my [PROJECT], I worked with our Project Manager and lead Business Analyst to define the requirements processes, designing templates for various deliverable (Requirements Management Plan, BRD, FRD, Use Cases, RTM etc). I also assisted our Lead BA in drafting Requirement Management Plan and many other artifacts (Vision document, scope statements etc).
(3) ONLY IF YOU HAVE EXPERIENCE IN THE DOMAIN. Since I have already worked in the [DOMAIN], there will be none to minimal learning curve.
(4) I have worked in a similar project environment. IF THE JOB NEEDS AN INDIVIDUAL TO WORK INDEPEDENTLY, SITE AN EXAMPLE FROM A PROJECT WHERE YOU WORKED AS A BA INDEPENDENTLY. SIMILARLY, IF THE JOB NEEDS TO WORK IN A TEAM, SITE AN EXAMPLE FROM A PROJECT WHERE YOU TELL THEM THAT YOU WORKED AS A GREAT TEAM PLAYER AND ENJOYED IT THOROUGHLY.

2. **Why do you think projects fail? Can you give an example where a project failed?**

ANS - I believe projects fail mostly because of non technical reasons such as lack of stakeholder participation, lack of buy-in from the end user community, desire to maintain/gain control over business processes and applications supporting those business processes, poor project management and last but not the least poor requirements. For example, in my [NAME THE PROJECT], though one of the project got initiated, it was shelved half way through since end-user community resisted to the changes. Their buy-in was very important for the success of the project since ultimately they were the folks who would be using the application.

3. **Could you describe an issue/problem you encountered in one of the projects and how you dealt with it?**

ANS - Some of the issues/problems I have seen in my past projects are (a) Conflict between two or more stakeholders. This is the most common issue I have seen. Each groups gravitates towards their individual goals rather than collective goal. Also, I have seen stakeholders like to keep control over a business process/ and the systems that are supporting those business process. These are very commonly seen. (b) sometimes project does not address all stakeholder needs. For example, in [NAME OF THE PROJECT], the project champion, who was key point of contact in giving us the stakeholder needs, failed to address business needs of one of the stakeholder groups. This led to lot of rework later in the project. second (c) The project goals are not linked to the client's overall vision. Though the projects were successfully rolled out it still didn't completely solved the problems clients were facing.

4. What motivates you to join our project/team?

ANS - IF YOU HAVE WORKED IN THE DOMAIN BEFORE:
Since I have worked in this domain before, I am really looking forward of being part of the team. Also, I have all the required elements (education, skills, expertise) that are needed for this position. Also, just like I will bring all my experience to the table and contribute towards the success of your project (in terms of domain experience, requirements processes, requirements management activities and tools and techniques), I am sure I will also have great things to learn from your project team. This is exactly what motivates me to join your project team.

5. **What were some of the things you did not like about your last job?**

ANS - You should tell things in such a way that it does not get interpreted in a negative way. I.e. you were under utilized and is not able to deliver value to the full potential.

6. **What do you consider your most significant weaknesses?**

ANS - The most obvious thing that comes to my mind is that I am too much of a perfectionist. For example, I will make sure all my work products (artifacts) are defect free, complete, free of grammatical mistakes, and gaps. My coworkers have told this in the past that I am very picky on such things. Second, I don't know whether it is a good or bad quality, but I cannot say 'No' very easily to anyone, be it work place, home or friends. Third, I get concerned/anxious if I don't understand something though it may not be directly related to the requirements. I try to understand each and everything that is connected to the problem domain/project scope.

7. What are your biggest accomplishments?

ANS -My greatest accomplishment was when I helped our client [NAME OF THE CLIENT] in [NAME OF THE PROJECT] to establish a set of requirements processes, helped them identify key

issues/bottlenecks in their existing processes, and established a governance framework for the entire requirements area. My efforts were truly appreciated by the senior leadership and more importantly, these were very beneficial to the entire team during the project execution.

8. How do you accept criticism?

ANS - I really take such comments in a very positive way since it lets you know where exactly to improve and be a better person.
Give few real life examples if possible and how it made a positive impact.

9. What is the most difficult situation you have faced?

ANS - The most difficult situation I have ever faced is a project getting stalled due to two different group of stakeholders trying to gain control over the requirements and for the project. The main contention was each group wanted to gain control over the key business processes, which were being added to the existing ones. At one time the project was at risk of getting shelved off. Our senior leadership did a great job in managing these two groups of stakeholders. Finally, we were back on track after few weeks. My role was to provide inputs to the senior leadership team.

10. Do you prefer working with others or alone?

ANS - I am flexible either ways. I am equally comfortable working an individual contributor as well as part of a team. For example, in my [NAME OF THE PROJECT], I was the only BA who was responsible to elicit, analyzed, document, communicate, validate and manage the requirements. On the contrary, in my [NAME OF THE PROJECT', I was part of the team wherein I was responsible for doing requirements for a specific module of the entire application.

11. What are some of the things you and your supervisor/team member have disagreed on and how would you resolve them?

ANS - I remember an instance where I and my requirements lead were not in agreement with some estimates. We had submitted some estimates based on certain set of high level requirements. These estimates were used by our leadership to convince our customer and get their buy-in for the project. But when the project got initiated and we found a whole new set of requirements, I provided revised estimates to my lead. He was bit reluctant to take them back to the senior leadership and wanted us to stick to the old estimates.

Initially he shot down the idea of taking new estimates back to the customer. But when I sat down with him and explained the ripple down impact it would have on the downstream activities, he was convinced that it's better to be transparent and truthful than to deal with these issues later.

12. **Why did you wanted to become a BA in spite of having a different education and experience background?**

ANS - Though I began my career in QA and was successful in performing my duties, I slowly started feeling boredom to do repetitive tasks. Second, I was not feeling challenged enough in terms of career growth. I selected business analysis career exactly for these reasons. First, each project is unique and there is always a new issue/problem/opportunity, which clients wants us to solve. Also, It's very challenging and exciting to see something new (in terms of domain, requirements processes, SDLC environments, tools and techniques used for requirements) all the time. Also, as a business analyst, I get an opportunity to involve myself in almost all phases of SDLC, right from eliciting requirements from the business stakeholders, coordinating closely with development/test team to help them understand the requirements, performing User Acceptance Testing (UAT).

13. How do you manage if requirements are changing rapidly?

ANS - I believe this is where a good business analyst makes a difference. Since business environment changes, so do the business needs of our customers and with that we see lot of requirements getting changed. That's exactly why we see all the agile methodologies getting popular in recent time i.e. to be able to handle changing customer needs and requirements. The best way to manage such situations is, based on the requirements, get a prototype early on in the project lifecycle. I have experienced this that once we show a working prototype of an application, it clears up lot of confusion in the customers mind and the next time, it's very much likely that they are able to articulately tell us what their business needs are. Also, I would also create an atmosphere of collaboration and commitment across the stakeholder group.

14. How do you deal with difficult stakeholders?

ANS -As a business analyst, I am generally diplomatic towards the stakeholder community. Though I can read in between the lines, I focus purely on getting the business needs and the requirements that would meet their business need. For example, in [NAME OF THE PROJECT], we had a customer who was very aggressive in terms of adding additional requirements to the scope. Without offending them, I would let them know 'Loud and Clear' that the amount of resources (manpower, cost and time) it would take to do the additional requirements and we would be more than happy to add them. Finally, customers themselves would say 'No' to it.

15. What will you miss most about your current job?

ANS - I will definitely miss my entire team (customers, project team our customers and SMEs). Unfortunately, it is time to say goodbye and wish them luck.

16. How would you deal with a situation wherein your project team members starts complaining about one of your subordinates?

ANS - I would see try to find out what the root cause of it rather than start acting on it. Sometimes it's just a perception of people which makes us to believe something or judge someone and each one of us may have different perceptions towards things, people, incidents, events etc. For example, in one of our project, team members didn't go along with one of our developers. Initially I thought the same. But upon knowing that person, I realized that he was just very outspoken and upfront in his discussions. Sometimes people don't like to listen such things but at least he was transparent in his thoughts and ideas. I saw it like that.

17. You joined a new project in a new domain. How would you proceed in getting the domain knowledge?

ANS - To gain a quick yet sufficient understanding of a new industry the business analyst must choose tools and techniques that generate a high-level, overarching view of the space with the least amount of time investment possible.

The first, and possibly most effective, tool for this purpose is the business entity model. The business entity model (or logical data model) documents the “things” or “nouns” that the business uses or interacts with in order to accomplish their work. Examples might be a bond, a loan, a borrower, a stock, a savings account, etc.

After the entities of the business are identified, attributes which describe each entity are also identified. So a “Borrower” may have attributes such as Name, SSN, Date of Birth, etc. In this way, the attributes further refine our understanding of the "things" within our domain.

Finally, the relationships between entities are identified. For example, a Borrower can have a Loan, and a Loan can have Collateral.

As attributes and relationships are identified for each entity, the business analyst and the rest of the project team can quickly understand the details of the business domain regardless of the level of previous domain knowledge. While a business entity model will continue to be updated and refined over time, creating a basic business entity model early on does not require a great deal of time.

Another great tool is the context diagram. The Context Diagram (a specialized version of a data flow diagram) shows a 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.) So it maps the domain in terms of processes, the data used by those

processes, and the movement of data throughout an organization. In this way, the Context Diagram works hand-in-hand with the Business Entity Model using the information already documented.

18. **What type of decisions do you have difficulty making?**

ANS - You need to convey the message that you are very good in decision making but in certain situations, you have to be more careful. For example: Initially in my career, I would never say 'No' to anyone. But slowly I have realized that it's more important to be realistic and transparent rather than over committing and lose trust. Also, sometimes choosing between two great ideas would be challenging too.

19. How would you assess your value as a business analyst?

ANS - 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 you 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.

20. **How do you identify stakeholders?**

ANS - Most Business Analysts know that identifying project stakeholders is important. But how do you identify them and know with any degree of certainty that you haven't missed someone important? After all, if you overlook a key stakeholder then the project is likely to suffer later due to scope change or a lack of acceptance by the business users.

To start, the business analyst should list the current project team members and identify each team member's functional manager. Many of the functional managers will likely be stakeholders. Also, an examination of the org chart can be helpful. By identifying which areas of the business are impacted by the project and then identifying people in leadership positions within those areas, the Business Analyst can ferret out stakeholders that might be overlooked.

So far, we've mostly described stakeholders directly impacted by the project. The business analyst also needs to be sure to consider other interfacing entities and organizations such as vendors or government agencies.

Finally, the business analyst doesn't need to rely solely on themselves. Instead, use the collective knowledge of previously identified stakeholders. A short survey can be used to gather from stakeholders who else they believe need to be involved. While just an example, the following

questions can be used to help uncover a stakeholder that might have been overlooked since different groups within the organization may have a different perspectives.

- ✓ Who is requesting the project deliverable?
- ✓ Who is the champion of the project?
- ✓ Who controls the budget for the project?
- ✓ Who is currently responsible for the business process that may be impacted by the project?
- ✓ Who is currently responsible for the systems that may be impacted or replaced by the project?
- ✓ Whose job is directly impacted by the delivery of the project?

Closing

Your closing questions will be asked toward the end of the interview and are almost always directly related to your current position within the interview process. These are designed to give you a better idea of where you stand in the interview process and what the prospective company's interest is in you.

1. **How quickly can you join our team?**

ANS - Option 1: Since I have just finished my last project, I am available to join your project as early as you would like me to.
Option 2: If selected, I will need to 2 to 3 weeks to finish my work-in-progress tasks and transition all project to my coworker.

2. If given this position, what value will you add to our project?

ANS - (1) My previous experience in various domains, (2) Business analysis experience during the past 7 years and (3) exposure to various tools, techniques, and SDLC models.

3. **Is this something you are looking for?**

ANS - After talking talking to you all regarding the project and reading the job description, this seems to be very promising and interesting position.

4. **We are looking for someone who can hit the ground running.**

ANS - I am quite confident to handle the work pressure and start contributing to the team. For e.g. in one of my project I was told to produce an RTM in the very first week of joining.

5. **We have a strict policy regarding telecommute. Is it OK with you?**

ANS - Don't negotiate or show some displeasure.

6. **Tell me anything else you would like us to know about you that will aid us in making our decision.**

ANS - (1) Friendly person. (2) Someone who fits in the environment very quickly (3) Have good people skills and (4) Always eager to learn new things (5) Highly adaptive to the any environment I am put into For e.g. I have worked in pure waterfall and immediately I was exposed to agile methodologies. I enjoyed working in both environments.

Questions To Be Asked At The End

1. What is the scope of this project? What is the duration of the project?
2. May I know who is the end client and domain (Only if they have not disclosed)?
3. Was eager to know about your requirements processes followed by the team?
4. If selected, what is an approximate time-frame of joining the project?
5. What is the location of the project team?
6. Any special company policies I should know? [For example: Some companies don't allow VPN (Virtual Private Network) usage, telecommuting etc [ASK THIS QUESTION VERY JUDICIOUSLY]
7. What are the tools used by the team?
8. What am I expected to achieve/deliver in first 3 months?
9. How is team structured?