PART 1:

1. What are the differences between analysis modeling and design modeling?

In general Analysis, the name itself suggests that it is a kind of investigation or understanding the scenario or problem in any particular organization. For an instance in any software organization there will be many issues from the side of server’s, clients, systems or employees. Analysis part plays a key role in where we can find the root cause for the actual issue and with the analysis report we can come to decision making phase. Whereas the design phase is the next step towards finding the solution for the analyzed issue.

Analysis and Design, this both phases comes one after other in the software design model. Analysis modeling is something where we design the use case diagram and use case diagram will actually describes the requirements for the project. Analysis modeling involves to provide the functionality of the system, feasibility of the system to make it real or to make user understand or checks system whether it will make user benefit and fulfill his needs. Whereas design modeling will let to code development and also describes the class diagram which is define from requirements phase. Here we could see the mockup designs for the system before the code is developed and then designed actual system.

<http://devhawk.net/blog/2004/3/30/analysis-vs-design-modeling>

<http://www.geekinterview.com/question_details/45049>

<http://www.objs.com/x3h7/a&d.htm>

<https://answers.yahoo.com/question/index?qid=20080922071149AA6snxg>

1. What is meant by service-oriented architecture?

Service-oriented architecture is a collection of services and these services interact with other services to communicate. The kind of communication is happened within the services are transformation of data to each other or coordinating the services to each other for an activity. Initial service-oriented architecture was by using DCOM or object request brokers established on CORBA.

Service means a function which can be defined and which will not depend on other services.

* Service is a self-contained.
* Services are made up of by other services.
* Services are a reasonable activity for a repeated business that has defined results for an instance providing the weather data.

Some of the key Characteristics of Service-oriented architecture is given below:

* ([WSDL](http://www.javaworld.com/#resources)) Web Services Description Language is the formal language to describe the SOA.
* XML Schema is used to communicate information in SOA services.

SOA mainly used because at present they are existing applications or business which is running in IT as heterogeneous throughout the industry. Most of the old business are doing new business processes, to do build the old system to new current business is not possible. Industries should change the new business process by new requirements for the users support. SOA has the feature with his service where old businesses can be upgrades with the new requirements.

<http://searchsoa.techtarget.com/definition/service-oriented-architecture>

<https://www.opengroup.org/soa/source-book/soa/soa.htm>

<http://www.javaworld.com/article/2071889/soa/what-is-service-oriented-architecture.html>

1. How do you know when "enough is enough"? What should you look for when determining when software is ready to go into production? What indicates that enough testing has been done?

From my personal experience, I used to work as a tester in Information technology where I have seen the exact process and how the management team decides to give the approval for the software to deploy in production or when the application is ready to go to productions. I can give an exact answer to this question as a tester point of view.

After the development team has given their best to develop software and then tester will come in to the picture with a single aim to break that code / software with his/her testing approach, they are two kinds of testing (Manual and Automation). Then after the successful test if they are major number of bugs in the software which is actually creating an error in the flow of the software then it has to be redeveloping to avoid those bugs from development team. Management has the reports of the testing results of each and every software and then makes a decision on the software to give signoff to deploy or not, less bugs in testing and bugs will not create any issue in the flow then the software is ready to go to live.

Key points which help management team to decide when "enough is enough" testing:

* We never say that any software is 100 bugs / error free and this is main reason due to less time and resources giving to develop or test the software.
* Existing bugs which will not affect the flow of an application or bug will not turn out to effect huge.
* Some time management decides for a build release with the existing bugs by informing the client to wait until next release to fix the existing bugs. These kinds of bugs are categorized as known bugs.
* Some time the developers are unable to rectify the bug due to less time or more pressure of next development and team will postponed the bug fixing to next build.
* Client budget allocation also plays a key role to work on the accuracy of software. In some case client provides less interest to testing phase where as management team is not responsible for any issue in provided software.

1. Why do many projects end up having unreasonable deadlines? How should a project manager manage unreasonable demands?

This is truly depends up on the management and lead team decision which was agreed to delivered the product at the fixed time interval. To meet the actual deadline of the project, management team has to make a clear matrices table for the projects or modules in the project to calculate the deadlines of each and every employee work assigned. Today we have many project management reporting tools to track the deadlines of each employee with their assigned work. But why many projects end up having unreasonable deadlines is the key to answer over here.

As I came from a small origination I believe that the work culture in the small scale industry is very fast and management expect more productivity from the employees these will be the reason why there will be a unreasonable deadlines seen for the project. Here organization expects more output from the employees than with the time frame given.

Unreasonable deadlines in the project also seen with the Unrealistic project planning done by the management and lead team, this could be done with the pressure of client or organization when though everybody aware if its negative results.

In this case I consider the main responsible team is the management team to blame where they have to make understand the higher authority about the hardest part of completing the assigned project or issue which will be faced during the process but failed to do it because of job security.

* Inexperienced Management team or Lead members in a project.
* Wrong selection of [Software Development Methodologies](http://www.itinfo.am/eng/software-development-methodologies/) for the appropriate projects.

As I have mentioned today we have many monitoring tools which will show the deadlines of completion of the project by using tools management should measure the outcome. Also by updating to higher team about the progress of the project so that it will let to over come the unreasonable demands from client or organization.

* Project manager manage unreasonable demands by adding more experience employees to the project team so that they can work faster and their experience will add strength to the team where it required.

<http://www.askamanager.org/2012/05/dealing-with-unreasonable-deadlines.html>

<http://www.techrepublic.com/article/managing-projects-with-unrealistic-deadlines/>

<https://www.themuse.com/advice/reality-check-how-to-handle-unrealistic-expectations-from-your-boss>

5. How do non-functional requirements affect the design of a system? Please provide an example.

Functional requirement tells us what the system does and it is very important

<http://www.codingthearchitecture.com/2007/07/09/the_influence_of_non_functional_requirements.html>

<http://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/>

PART 2:

LibraryThing is a social media for book lovers. It promotes itself as ‘Facebook for books’. It is an online service to help people discover and catalog their books easily. Using the Principles for User Interface Design, evaluate LibraryThingand provide recommendations to improve the user experience.

Link: <https://www.librarything.com/>

The given LibraryThing website basically has four different web pages. I have created my login using my email id and see that LibraryThing website provides a lot of different pages with lot of features and options to make user benefit. As a beginner I did not understand the application once I have logged in as a first time and spent some more time to understand myself. Using the principles of User Interface Design, below I have given some of the recommendations to improve the user experience.

**Explain the rules** /**A crucial moment:** I recommend that the LibraryThing website has to provide the initial directions and guidelines tour of the website so that as a new user we can learn the application without any difficult. This guidelines feature makes new user to navigate the website faster for the next time. As there are more features to use and I really do not know the exact benefits from the options given in the website so by navigation option user can lean the importance of different options given in the website because ultimately website need to make benefit from user by providing the user what he/she needs .

**Don’t create busy user interfaces**: as we can see in the LibraryThing website under the pages of (Groups, Talks, and Zeitgeist) there is a lot of information available with the screen and it is making more complicated information within the single screen. In my opinion I would like to see the same information in different sub pages so that we can break down the pages and check the information which looks more simple when compare to user experience.

that they are

**Navigation within a screen is important**.

* Topics made to discuss among group of authors and should be made public so that we can know how they think and work.
* Audio and video Recordings by famous speaker’s.

**Events:** Library can be provided the information about the world events / conference which are held around the world so that people can follow their interested author in to that even and gain more knowledge from the meeting.

* I would suggest that there should a video featuring about the library history and articles which has video. Need to post the videos of author who is attend any conference and thus people can know more about the author.
* I would also recommend that there should be more departments like Education, News, History, Copyright information and Digital to further development in the website.
* All the information should be in single standard Format so that users can feel a formal communication in the website.

Citation:

"User Interface Design Tips, Techniques, and Principles." *User Interface Design Tips, Techniques, and Principles*. Web. 09 May 2016.

"Principles of User Interface Design." *Principles of User Interface Design*. Web. 09 May 2016.