**Requirement Elicitation and Analysis**

**Requirement Engineering:**

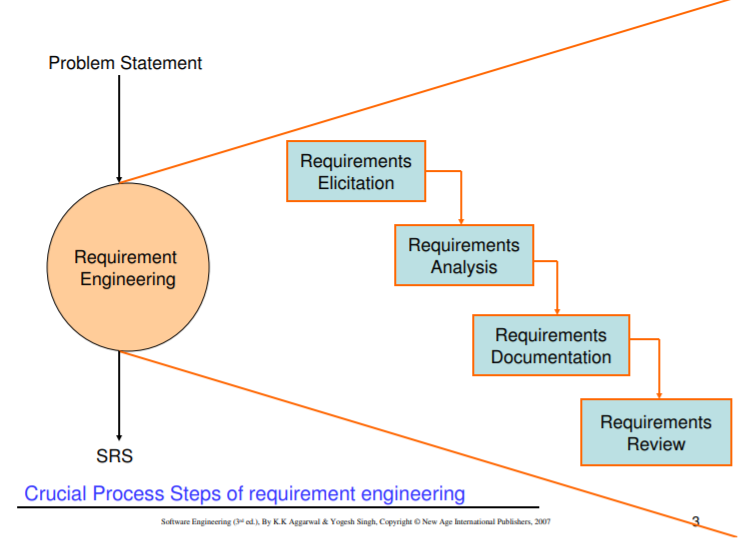
The process to gather the software requirements from client, analyze and document them is known as requirement engineering. The goal of requirement engineering is to develop and maintain sophisticated and descriptive **'System Requirements Specification'** document.

Requirement Engineering Requirement Engineering is the disciplined application of proven principles, methods, tools, and notations to describe a proposed system’s intended behavior and its associated constraints.

Requirements describe:

**What not How**

Produces one large document written in natural language contains a description of what the system will do without describing how it will do it.

****

SRS may act as a contract between developer and customer.

**Requirements Elicitation:** Requirement elicitation is the process of collecting information from stakeholders. It serves as a foundation in documenting the requirements for application development.

There are a number of elicitation techniques to gather requirements or to collect the information from the stakeholders. Some of the requirement elicitation techniques are as follows.

1. Document analysis
2. Observation
3. Interview
4. Prototyping
5. Brainstorming
6. Workshop
7. JAD (Joint Application Development)
8. Reverse engineering
9. Surveys/Questionnaire

**Document analysis**

Document analysis is one of the most helpful elicitation techniques in understanding the current process. Documents like user manuals, software vendor manuals, process documents about the current system can provide the inputs for the new system requirements.

Steps involved in document analysis are:

* Evaluating whether the existing system and business documents are appropriate to be studied.
* Analysing the documents to identify relevant business details.
* Reviewing and confirming identified details with subject matter experts.

There could be a lot of information that can be transferred to a new system requirements document. Evaluating the documentation can assist in making the As-Is process document, and conducting GAP analysis for scoping of the project in question.

**Observation**

This elicitation technique helps in collecting requirements by observing users or stakeholders. This can provide information about the exiting process, inputs and outputs. There are two kinds of observations — active and passive.

In active observation, the business analyst directly observes the users or stakeholders, whereas in passive observation the business analyst observes the subject matter experts.

This helps the business team understand the requirements when users are unable to explain requirements clearly.

**Interview**

An interview is a systematic approach to elicit information from a person or group of people. In this case, the business analyst acts as an interviewer. An interview provides an opportunity to explore and/or clarify requirements in more detail. Without knowing the expectations and goals of the stakeholders it is difficult to fulfill requirements.

**Prototyping**

Screen mockups can support the requirement gathering process, when introduced at the correct time. Mockups help stakeholders visualize the functionality of a system. This can be an advantage to business analysts and stakeholders since this allows them to identify gaps/problems early on.

**Brainstorming**

Brainstorming is an efficient way to define their requirements. Users can come up with very innovative ideas or requirements. This can help gather ideas and creative solutions from stakeholders in a short time.  
Users or stakeholders can come up with ideas that they have seen or experienced elsewhere. These ideas can be reviewed and the relevant ones can then be included in the system requirements.

**Workshop**

Workshops comprise a group of users or stakeholders working together to identify requirements. A requirement workshop is a structured way to capture requirements. Workshops are used to scope, discover, define, and prioritize requirements for the proposed system.

They are the most effective way to deliver high-quality requirements quickly. They promote mutual understanding and strong communication between users or stakeholders and the project team.

**JAD (Joint Application Development)**

Joint Application Development (JAD) technique is an extended session to the workshop. In the JAD session stakeholders and project team works together to identify the requirements. These sessions allow the business team to gather and consolidate large amounts of information. Identification of stakeholders is the critical to the overall success of the JAD session. The JAD team includes business process owners, client representatives, users or stakeholders, business analysts, project managers, IT experts (developers, quality assurance, designers, and security).

**Reverse engineering**

This elicitation technique is generally used in migration projects. If an existing system has outdated documentation, it can be reverse engineered to understand what the system does. This is an elicitation technique that can extract implemented requirements from the system.  
There are two types of reverse engineering techniques.  
Black box reverse engineering: The system is studied without examining its internal structure (function and composition of software).  
White box reverse engineering: The inner workings of the system are studied (analysing and understanding of software code).

**Surveys/Questionnaires**

Questionnaires are useful when there is a lot of information to be gathered from a larger group of stakeholders. This enables the business team to gather requirements from stakeholders remotely. The design of the questionnaire is very important, since it can influence the answers that people provide.

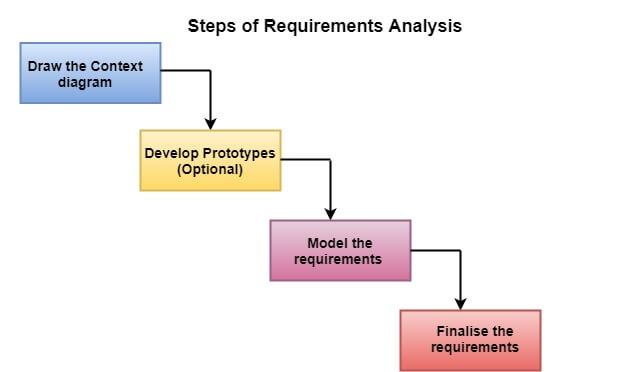
In addition to the above-mentioned elicitation techniques, there are many more are on the market. It is very difficult to say that which elicitation technique is suitable for all projects. Not all elicitation techniques can be executed for every project.

When selecting an elicitation method, factors such as the nature of the project, organizational structure and type of stakeholders are taken into account by the business team before deciding which technique works best. Having said that, brainstorming, document analysis, interviews, prototyping and workshops are the most widely used requirement elicitation techniques.

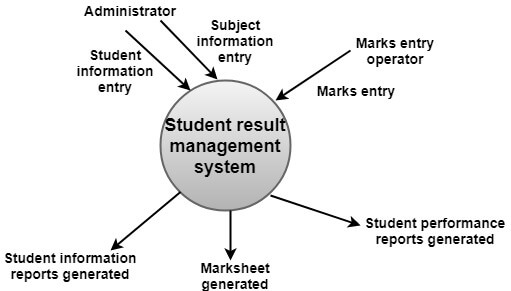
**Requirements Analysis:**

Requirement analysis is significant and essential activity after elicitation. We analyze, refine, and scrutinize the gathered requirements to make consistent and unambiguous requirements. This activity reviews all requirements and may provide a graphical view of the entire system. After the completion of the analysis, it is expected that the understandability of the project may improve significantly. Here, we may also use the interaction with the customer to clarify points of confusion and to understand which requirements are more important than others.

**The various steps of requirement analysis are shown in fig:**



**(i) Draw the context diagram:** The context diagram is a simple model that defines the boundaries and interfaces of the proposed systems with the external world. It identifies the entities outside the proposed system that interact with the system. The context diagram of student result management system is given below:



**(ii) Development of a Prototype (optional):** One effective way to find out what the customer wants is to construct a prototype, something that looks and preferably acts as part of the system they say they want.

We can use their feedback to modify the prototype until the customer is satisfied continuously. Hence, the prototype helps the client to visualize the proposed system and increase the understanding of the requirements. When developers and users are not sure about some of the elements, a prototype may help both the parties to take a final decision.

Some projects are developed for the general market. In such cases, the prototype should be shown to some representative sample of the population of potential purchasers. Even though a person who tries out a prototype may not buy the final system, but their feedback may allow us to make the product more attractive to others.

The prototype should be built quickly and at a relatively low cost. Hence it will always have limitations and would not be acceptable in the final system. This is an optional activity.

**(iii) Model the requirements:** This process usually consists of various graphical representations of the functions, data entities, external entities, and the relationships between them. The graphical view may help to find incorrect, inconsistent, missing, and superfluous requirements. Such models include the Data Flow diagram, Entity-Relationship diagram, Data Dictionaries, State-transition diagrams, etc.

**(iv) Finalise the requirements:** After modeling the requirements, we will have a better understanding of the system behavior. The inconsistencies and ambiguities have been identified and corrected. The flow of data amongst various modules has been analyzed. Elicitation and analyze activities have provided better insight into the system. Now we finalize the analyzed requirements, and the next step is to document these requirements in a prescribed format.