

Requirements Elicitation

- **Interviews:**
 - A systematic approach designed to elicit information from a person or group of people in an informal or formal setting by talking to an interviewee, asking relevant questions and documenting the responses.
 - In an interview, the interviewer formally or informally directs questions to a stakeholder in order to obtain answers that will be used to create formal requirements.
 - One-on-one interviews are typically most common. In a group interview (with more than one interviewee in attendance) the interviewer must be careful to elicit responses from all attendees.
 - The most widely used elicitation and fact finding technique and requires the most skill and sensitivity.
 - Conducting an interview requires good planning, good interpersonal skills and an alert and responsive frame of mind.
 - One of the key goals of interviewing is to ensure that the biases and predispositions of the interviewer do not interfere with a free exchange of information.
 - Used to explore issues and can collect some quantitative, but mostly qualitative data.
 - **Structured Interview:** where the interviewer has a **pre-defined set of questions** and is looking for answers.
 - **Unstructured Interview:** where, without any pre-defined questions, the interviewer and the interviewee **discuss topics of interest** in an open-ended way.
 - Build a list of questions and issues to be discussed, probe for details
- **Surveys/Questionnaires:**
 - A questionnaire or “survey” is a document containing a number of standard questions that can be sent to individuals to obtain information **from a large number of people** or when the people are geographically dispersed.
 - Can collect both quantitative and qualitative data
 - They are also appropriate for systems that will be used by the general public and where the analyst has to investigate all the types of users of the system.
 - Questions = Yes/No; MC; Scaled; Open-ended
- **Observations:**
 - In this approach, the analyst **observes** the actual execution of **existing processes** by the users, usually without direct interference.
 - Seeing the environment and domain where the system will be situated in action gives additional perspectives and a better understanding of system functionalities.
 - Observation also allows us to **verify** statements made in interviews and surveys to determine whether the procedures within the domain really operate as they were described.
 - Observe system **as it actually behaves**
 - Avoid disturbing users from their normal activities
 - Observation is essential for gathering some quantitative and mostly qualitative data about people’s jobs.
 - It can verify or disprove assertions made by interviewees.
 - It is **useful in situations where different interviewees have provided conflicting information** about the way existing system works.
 - Be aware of that individual behaviour maybe altered because they know they are being studied.
- **Prototyping:**
 - A prototype is an **initial working model** of a larger, more complex entity, usually a program with limited functionality that is built to test out some aspect of **how the final system will work**.
 - For requirements elicitation, we often build “throwaway” prototypes, mostly for user interface and in conjunction with developing use cases.
 - Operative: a prototype should be a working model. A simple start will be a “mock-up”
 - Focused: To test specific concept or verify an approach, it should be focused on a single objective.
 - Quick: Tools are needed so that the prototype can be built and modified quickly.
 - Prototypes may be constructed with various objectives in mind:
 - To investigate user requirements
 - To focus on human-computer interface
 - Investigate input and output and its form
 - Investigate most suitable interface
 - To determine whether a particular implementation platform can support certain processing requirements

- To determine the efficacy of a particular language, a DBMS, or a communication infrastructure
- **Prototypes:**
 - Paper prototype
 - **UI flows or storyboarding:** A storyboard is a series of drawings used mostly for **identifying user interfaces**; screens that the software will display are drawn.
 - Non-working software prototype
 - Working software prototype
 - Spike
- User interface-flow diagrams (Storyboards) offer a high-level view of the interface of a system, you can quickly gain an understanding of **how the system is expected to work**. It puts you in a position where you can **validate the overall flow** of your application's user interface. For example, does the flow make sense?
- User interface-flow diagrams are typically used for one of the two purposes. First, they are used to **model the interactions** that users have with your software, as defined in a single use case. Second, they enable you to gain a **high-level overview of the user interface** for your application.
- Workshops:
 - A technique used to expedite requirements elicitation, also referred to as “Joint Application Development” or “focused groups”
 - The objective is to compress all of the activities involved in other fact finding techniques into a shorter series of workshop sessions with users and project team members.
 - These sessions are usually conducted in special rooms with supporting facilities: overhead projector, a white board, flip charts, and adequate workspace for the participants.
 - A workshop may be used to scope, discover, define, prioritize and reach closure on requirements for the target system.
 - A workshop may be used to generate ideas for new features or products, to reach consensus on a topic or conflicting views, or to review requirements.
 - Organised process: techniques such as brain storming, top down analysis
 - Documented approach: output of each session is documented in such a way to make it easy to read and understand and agree upon
 - Participants must be selected carefully representing different classes of stakeholders
 - Ensure that all stakeholders participate and have their input heard.
 - Must have a skilled facilitator
 - Meeting room should have all the necessary facilities and the environment be conducive to hold effective meetings
 - Visual aids (e.g. flip charts, whiteboard, large screens, GUI)
 - Goal and Agenda; Elicitation, Review, Sign Off workshops; Who...10-15 people max.;
 - Location and time; Facilitator; Subscriber; Tools; Food; Ice breaker; Follow-up
- **TWO TYPES OF REQUIREMENTS, DIFFERENCE BETWEEN FUNCTIONAL VS NON-FUNCTIONAL REQUIREMENTS?**
 - Functional Requirements [**things products must do**]
 - Describe the behaviour and information that the solution will manage
 - Describe the **capabilities** the system will be able to perform in terms of behaviours – specific information technology application actions or responses
 - E.g. The system shall enable hotel guests to book a room online
 - Non-Functional Requirements [**properties/qualities a product must have**]
 - Capture condition that do not directly relate to the behaviour or functionality of the solution but rather describe the environmental conditions under which the solution must remain effective or qualities that the system must have
 - Also known as quality/supplementary requirements
 - Requirements related to: **Capacity, Speed, Safety, Security, Availability**
 - E.g. software **performance** requirements, external **interface** requirements, software design constraints, software quality attributes
 - E.g. The software system should be able to produce a monthly sales report for a given month in less than 5 seconds and display it on iPad... The system shall be able to process 100 payment transactions per second.

- **ADVANTAGES AND DISADVANTAGES OF REQUIREMENTS ELICITATION TECHNIQUES?**
 - **Interviews**
 - Advantages:
 - Allows the interviewer and participant to have **full discussions** and **explanations** of the questions and answers
 - Personal contact allows responsiveness and adapting to what the user says.
 - Analyst can probe in **greater depth** than any other methods of elicitation
 - Allows interviewees to **express opinions in private** that they may be reluctant to express in public.
 - Disadvantages:
 - Can be time consuming and costly
 - Requires considerable commitment and involvement of the participants.
 - Interview results have to be transcribed and written and transcription and analysis of interview data can be complex and expensive.
 - Can be subject to bias
 - If conflicting information is given, it can be difficult to resolve and interviews are not an ideal means of reaching consensus across a group of stakeholders.
 - There is a risk of unintentionally leading the interviewee.
 - **Questionnaire/Survey**
 - Advantages:
 - An **economical** and **quick** method of gathering data from a **large sample**.
 - Can reach many people with low resource.
 - Used for answering specific questions.
 - Can be administered remotely.
 - Depending on how well it is designed, the results can be analysed easily by software applications.
 - Effective and efficient when stakeholders are not located in one location.
 - Disadvantages:
 - Effective questionnaires are hard to design (e.g. leading questions, misinterpretation of questions).
 - Questions are usually not answered completely.
 - The response rates for surveys are often too low for statistical significance.
 - There is no automatic way of follow up unless you do interviews later
 - **Observation**
 - Advantages:
 - Provide first-hand experience of the way the current system works
 - Data are collected in real time and can have a high level of validity
 - Can be used to verify information from other sources or to look for exceptions
 - Baseline data about the performance of the existing system and of users can be collected
 - Disadvantages:
 - Could be very time consuming
 - Need to analyse huge amounts of data
 - Most people do not like to be observed and may be disruptive to the person being observed.
 - Requires trained and skilled observer to be most effective.
 - There may be ethical problems if the person being observed deals with sensitive private or personal data or directly with members of public.
 - There may be logistical problems if the staff being observed work shifts.
 - Unusual exceptions and critical situations that happen infrequently may not occur during the observation.

- **Prototyping**
 - Advantages:
 - A prototype allows for **early user interaction and feedback**.
 - Can be an inexpensive means to quickly uncover and confirm a variety of requirements.
 - Supports users who are more comfortable and effective at articulating their needs by using pictures, as prototyping lets them “**see**” **the future system’s interface**.
 - Provides a vehicle for designers and developers to learn about the users’ interface needs and to evolve system requirements.
 - Disadvantages:
 - Depending on the complexity of the target system, using prototyping to elicit requirements can take **considerable time**.
 - A prototype may lead users to develop **unrealistic expectations** regarding the delivered system’s performance, completion date, reliability and usability characteristics. This is because an elaborated, detailed prototype can look a lot like a functional system.
- **Workshops**
 - Advantages:
 - Workshop sessions are very successful in **reducing project development efforts and shortening the schedule**.
 - Is able to gauge reaction to stimulus material (e.g. storyboards, screenshots)
 - A requirements workshop provides a means for stakeholders to collaborate, make decisions and gain a mutual understanding of requirements.
 - Disadvantages:
 - Risk involved in speeding up the decisions. Sometimes the decisions made about the requirements are not optimal.
 - At times, details are inappropriately defined or missed altogether.
 - May suffer from dominance and submission