**Module 2: Requirements Elicitation**

1. **What is Elicitation?**
   * Process of drawing out latent or implicit knowledge from stakeholders​(Module 3 - ReqsElicitat…).
   * Involves making explicit what is known but not clearly stated.
2. **Key Players in Elicitation**
   * **Users**: End-users of the software (can also be other systems).
   * **Buyers**: Those responsible for purchasing the software (may differ from users).
   * **Experts**: Domain experts who offer specialized knowledge​(Module 3 - ReqsElicitat…).
3. **Information Sources**
   * People (users, buyers, experts), documents, reference works, and informal channels like conversations and forums​(Module 3 - ReqsElicitat…).
4. **Elicitation Techniques**
   * **Interviews**: Structured/unstructured, rich information, but time-intensive.
   * **Group Meetings**: Semi-structured workshops, good for consensus building.
   * **Storyboarding/Prototyping**: Create visual representations for feedback.
   * **Questionnaires**: Good for reaching a broad audience, but may have poor response rates.
   * **Observation (Ethnography)**: Time-consuming but reveals natural user behavior​(Module 3 - ReqsElicitat…)​(Module 4 - Requirements…).
   * **Joint Application Development (JAD)**: Collaborative workshops that include multiple stakeholders​(Module 3 - ReqsElicitat…).
5. **Common Issues in Elicitation**
   * **"Yes but..."**: Users continually adding features.
   * **"Undiscovered Ruins"**: New requirements emerging during elicitation.
   * **User-Developer Communication Gaps**: Addressed by multiple techniques​(Module 3 - ReqsElicitat…).

**Module 2: Requirements Analysis**

1. **Purpose of Requirements Analysis**
   * Transform "what" (user needs) into "how" (design)​(Module 4 - Requirements…).
   * Focus on creating a communicative model between stakeholders and developers.
2. **Requirements Baseline**
   * A formalized document (SRS) that states system functionality and constraints​(Module 4 - Requirements…).
   * Acts as the foundation for the design and implementation phases.
3. **Analysis Techniques**
   * **Data Models**: ER diagrams, object-oriented analysis to represent data relationships​(Module 4 - Requirements…).
   * **Behavioral Models**: Use cases, statecharts, and sequence diagrams to represent system behaviors​(Module 4 - Requirements…).
   * **Flow Models**: Data flow diagrams (DFD), process models, and activity diagrams to illustrate functional flows​(Module 4 - Requirements…).
4. **Architectural Methods**
   * **Top-down approach**: Starting from high-level analysis and breaking it down.
   * **Bottom-up approach**: Building solutions to smaller problems and integrating them later​(Module 4 - Requirements…).
   * **Leveraging legacy systems**: Incorporating existing components into new designs​(Module 4 - Requirements…).
5. **Key Concepts**
   * RUP’s **4+1 View Model**: Logical view, process view, deployment view, implementation view, and use-case view​(Module 4 - Requirements…).
   * Multiple analysis models may be required to express different perspectives.

**Study Focus:**

* **Elicitation**: Techniques, key players, and common issues.
* **Analysis**: Transition from requirements to design, various modeling methods, and architectural approaches.