**Contemporay methods to determine a system requirements**

**1. Analyzing Software and Hardware Needs**

* Scope of business
* Resource needs
* Hardware requirements for customized software
* Scalability of management for future use
* Data storage (type and amount of data to be stored)

This step involves identifying the software and hardware necessary to meet business objectives. It includes evaluating computing resources, network infrastructure, and system scalability for future expansion. Additionally, it assesses data storage requirements based on system demands.

**2. Security Considerations (non functional requirements)**

* Security policies and compliance
* Structured security approaches

Security is essential to protect data and system integrity. This includes implementing encryption, authentication, and access control measures. Compliance with industry standards ensures protection against cyber threats.

**3. Stakeholder Analysis**

* Identifying key stakeholders
* Gathering user requirements

Stakeholder analysis identifies key individuals or groups affected by the system. Understanding their expectations helps in defining clear and relevant requirements. Engaging stakeholders early improves system usability and acceptance

**4. Requirement Gathering Techniques**

* **Interviews** – Engaging with users and stakeholders
* **Surveys** – Collecting quantitative insights
* **Observation** – Understanding existing workflows

Requirements are collected using methods like **interviews, surveys, and observations**. These techniques help in understanding user needs and operational challenges. A well-defined requirement-gathering process ensures that the system meets its intended purpose.

**5. Agile Model and Prototyping**

* **Iterative Development** – Continuous improvement with feedback loops
* **Rapid Prototyping** – Creating UI/UX designs with tools like **Figma**
* **Sprint Planning** – Breaking down requirements into manageable tasks
* **User Stories** – Defining system features from an end-user perspective
* **Collaboration** – Close interaction between developers, testers, and stakeholders

The Agile model follows an iterative approach, allowing continuous feedback and improvements. Prototyping tools like Figma help visualize system design before development. Agile fosters collaboration between developers, testers, and stakeholders to refine requirements.

**6. Data Analysis**

* Evaluating data structures and processing needs
* Optimizing system performance

Data analysis ensures the system can efficiently process and store required information. It involves structuring databases, analyzing data flow, and ensuring optimal storage solutions. This step helps maintain system performance and scalability.

**7. Functional and Non-functional Requirements**

* **Functional** – Features, modules, and user interactions
* **Non-functional** – Performance, security, and reliability

**8. Validation and Testing**

* User acceptance testing (UAT)
* Ensuring system meets business objectives

**9. Documentation and Traceability**

* Maintaining requirement documents
* Tracking changes throughout the project lifecycle

Proper documentation helps track system requirements, design decisions, and changes throughout the project. Traceability ensures that every requirement is implemented and tested. This step is essential for maintaining system quality and compliance.