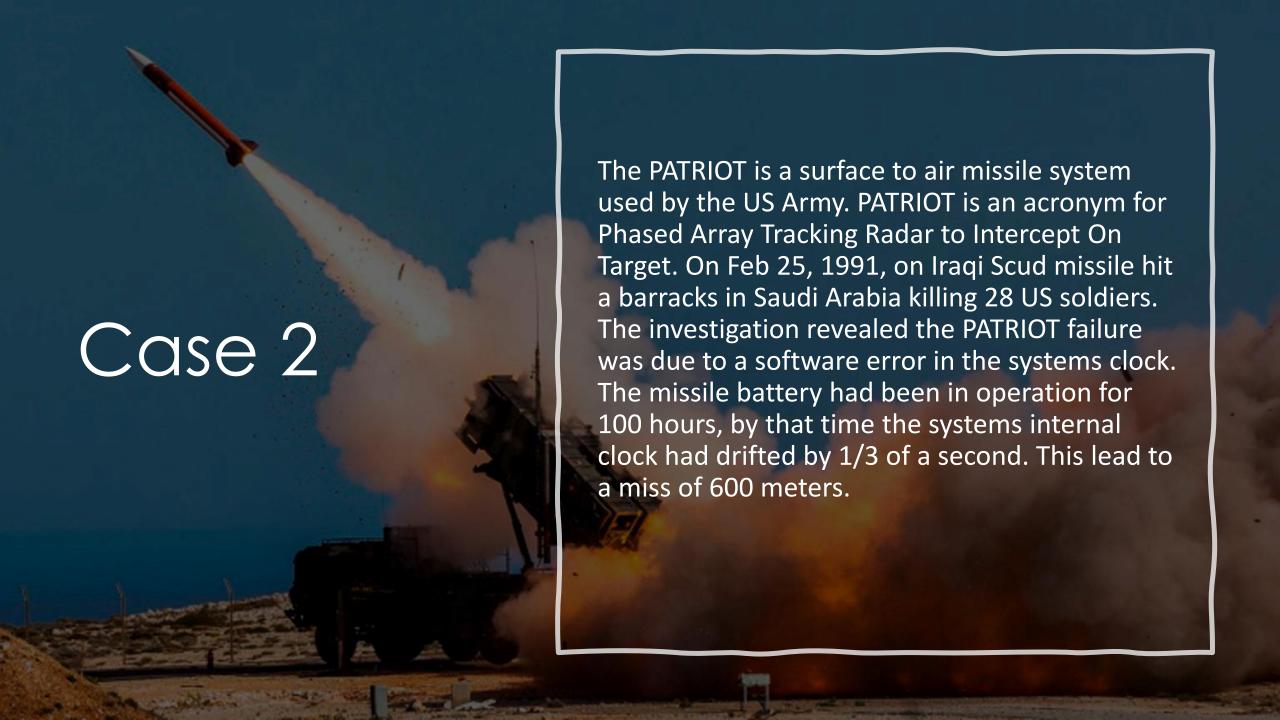
Systems Engineering for DoD

Lesson 2

Requirements Engineering



Statistics from NIST Report

NIST (National Institute of Standards and Technology) has published a comprehensive (309 pages) and very interesting report on project statistics and experiences based on data from a large number of software projects

http://www.nist.gov/public_affairs/releases/n02-10.htm (May 2002)

- 70% of the defects are introduced in the specification phase
- 30% are introduced later in the technical solution process
- Only 5% of the specification inadequacies are corrected in the specification phase
- 95% are detected later in the project or after delivery where the cost for correction on average is 22 times higher compared to a correction directly during the specification effort
- The NIST report concludes that extensive testing is essential, however testing detects the dominating specification errors late in the process

Requirements Errors

- 1) Errors of Conception
- 2) Errors of Specification
- 3) Errors of Implementation
- 4) Errors of Visualization
- 5) Errors of Requirement Management

https://www.proquest.com/openview/e77d01a7e18f3810 b0fcdf59b843abd5/1?pq-origsite=gscholar&cbl=2050634

Does Agile Work?

- A study finds 268% Higher Failure Rates for Agile Software Projects. A 2024 study consisting of 600 UK and US software engineers finds projects adopting Agile Manifesto practices are 268% more likely to fail than those which do the opposite.
- https://drj.com/industry_news/268-higher-failure-rates-for-agile-software-projects-study-finds/

Failure Rates

- 31.1% of software projects are canceled before completion.
- 52.7% exceed their original budgets by 189%.
- 75% of executives anticipate heir software projects will fail.
- Only 16.2% of projects are completed on time and within budget
- https://www.betabreakers.com/blog/software-survival-in-2024-understanding-2023-project-failure-statistics-and-the-role-of-quality-assurance/.

Failure Rates

- 40% of the offshored projects did not achieve foreseen advantages
- RE related errors are 48% of the total number of SDLC errors
- McKinsey reports that only 30% of digital transformation projects result in improved corporate performance.
- 90% of digital transformation projects fail to deliver any measurable ROI
- https://pmc.ncbi.nlm.nih.gov/articles/PMC7144980/
- https://www.forbes.com/sites/steveandriole/2021/03/25/3-main-reasons-why-big-technology-projects-fail---why-many-companies-should-just-never-do-them/

Requirements Errors

- Poor Requirements Quality
- Over Emphasis on Simplistic Use Case Modeling
- Inappropriate Constraints
- Requirements Not Traced
- Missing Requirements
- Excessive Requirements Volatility including Unmanaged Scope Creep
- Inadequate Verification of Requirements Quality
- Unprepared Requirements Engineers

-https://www.jot.fm/issues/issue_2007_01/column2/



Why requirements?

- What are the advantages of a complete set of documented requirements?
 - Ensures the user (not the developer) drives system functionalities
 - Helps avoiding confusion and arguments
 - Helps minimizing the changes
- Changes in requirements are expensive. Changing the requirements costs:
 - 3 x as much during the design phase
 - 5-10 x as much during implementation
 - 10-100 x as much after release

According to IEEE 830-1993

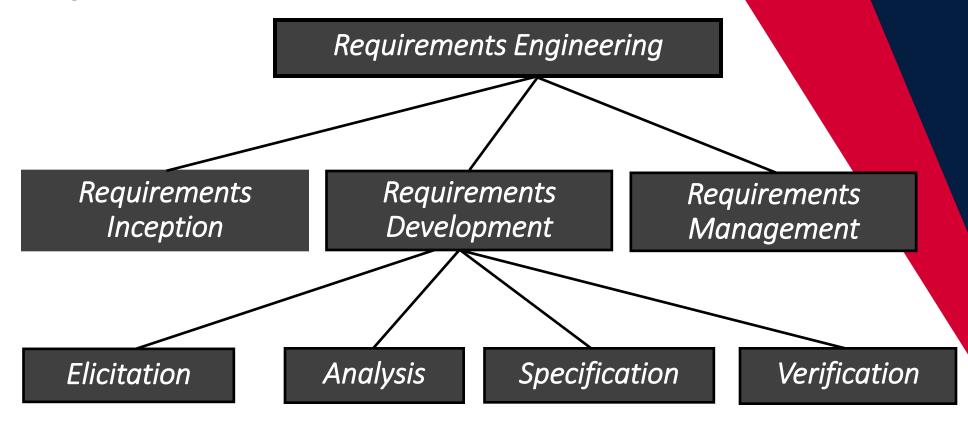
- A **requirement** is defined as:
 - A condition or capability needed by a user to solve a problem or achieve an objective
 - A condition or a capability that must be met or possessed by a system, to satisfy a contract, standard, specification, or other formally imposed document.



What is "Requirements Engineering"?

- Requirements Engineering (RE) is:
 - The activity of development, elicitation, specification, analysis, and management of the stakeholder requirements, which are to be met by a new or evolving system
 - RE is concerned with identifying the purpose of a software system... and the contexts in which it will be used
 - How/where the system will be used
 - Big picture is important
 - Captures real world needs of stakeholders affected by a software system and expresses them as artifacts that can be implemented by a computing system
 - Bridge to design and construction
 - How to communicate and negotiate?
 - Is anything lost in the translation between different worlds?

REQUIREMENTS ENGINEERING ACTIVITIES



RE ACTIVITIES

Inception

• Start the process, determine operational need, feasibility study, system scope, risks, etc.

Requirement's elicitation

 Requirements discovered through consultation with stakeholders

Requirement's analysis and negotiation

Requirements are analyzed and conflicts resolved through negotiation

• Requirement's specification

• A precise requirements document is produced

Requirement's validation

The requirements document is checked for consistency and completeness

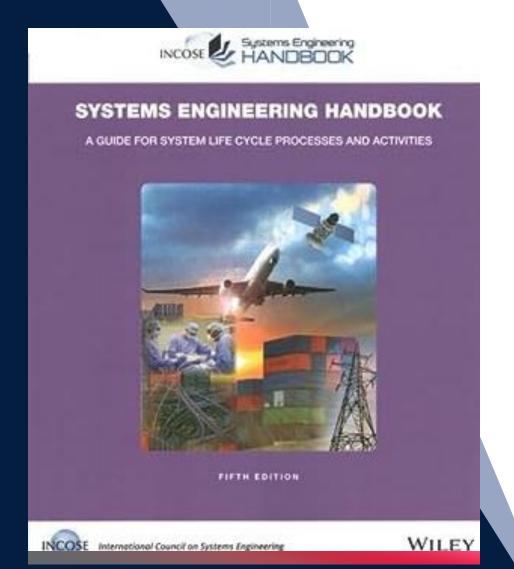
Requirements management

Needs and contexts evolve, and so do requirements!

INCOSE Handbook

The chapter in this book relevant to this lesson is:

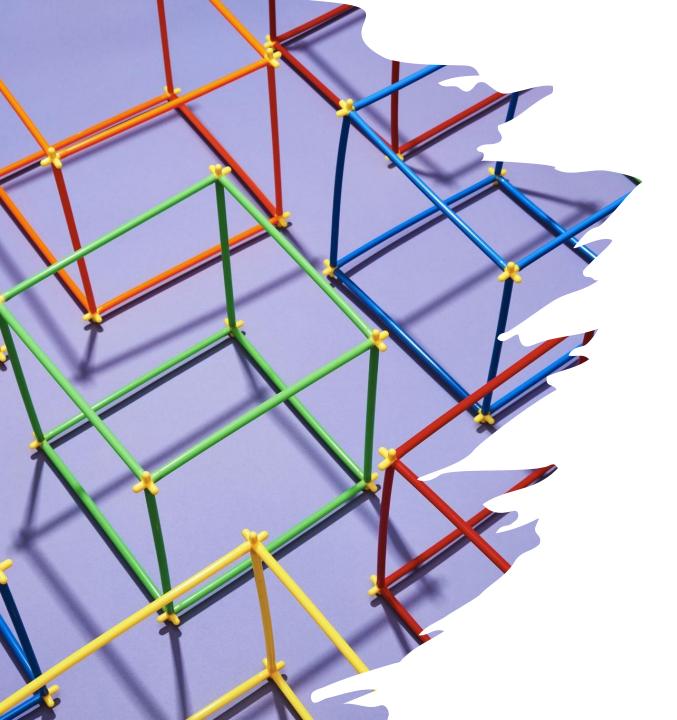
4.3 SYSTEM REQUIREMENTS DEFINITION PROCESS



Why requirements

- 2/3 of finished system errors are requirements and design errors
- A careful requirements process doesn't mean there will be no changes later
 - Average project experiences about 25% changes in the requirements
 - This accounts for 70-80% if the rework of the project Important to plan for requirements changes
- The case of critical applications





System Requirements

• System requirements are all of the requirements at the system level that describe the functions which the system as a whole should fulfill to satisfy the stakeholder needs and requirements, and are expressed in an appropriate combination of textual statements, views, and non-functional requirements; the latter expressing the levels of safety, security, reliability, etc., that will be necessary. - SeBOK

Types of System Requirements

| Types of System Requirement | Description | | | | | |
|--------------------------------|--|--|--|--|--|--|
| | | | | | | |
| Functional | Describe qualitatively the system functions or tasks to be performed in operation. | | | | | |
| Requirements | | | | | | |
| Performance | Define quantitatively the extent, or how well and under what conditions a function or task is to be performed (e.g. rates, velocities). These are quantitative requirements of system | | | | | |
| Requirements | performance and are verifiable individually. Note that there may be more than one performance requirement associated with a single function, functional requirement, or task. | | | | | |
| Usability Requirements | Define the quality of system use (e.g. measurable effectiveness, efficiency, and satisfaction criteria). | | | | | |
| Interface | Define how the system is required to interact or to exchange material, energy, or information with external systems (external interface), or how system elements within the system, | | | | | |
| Requirements | including human elements, interact with each other (internal interface). Interface requirements include physical connections (physical interfaces) with external systems or internal system | | | | | |
| | elements supporting interactions or exchanges. | | | | | |
| Operational | Define the operational conditions or properties that are required for the system to operate or exist. This type of requirement includes: human factors, ergonomics, availability, | | | | | |
| Requirements | maintainability, reliability, and security. | | | | | |
| Modes and/or States | Define the various operational modes of the system in use and events conducting to transitions of modes. | | | | | |
| Requirements | | | | | | |
| Adaptability | Define potential extension, growth, or scalability during the life of the system. | | | | | |
| Requirements | System Requirements | | | | | |
| Physical Constraints | Define constraints on weight, volume, and dimension applicable to the system elements that compose the system. | | | | | |
| Design Constraints | Define the limits on the options that are available to a designer of a solution by imposing immovable boundaries and limits (e.g., the system shall incorporate a legacy or provided system | | | | | |
| | element, or certain data shall be maintained in an online repository). | | | | | |
| Environmental | Define the environmental conditions to be encountered by the system in its different operational modes. This should address the natural environment (e.g. wind, rain, temperature, fauna, | | | | | |
| Conditions | salt, dust, radiation, etc.), induced and/or self-induced environmental effects (e.g. motion, shock, noise, electromagnetism, thermal, etc.), and threats to societal environment (e.g. legal, | | | | | |
| | political, economic, social, business, etc.). | | | | | |
| Logistical | Define the logistical conditions needed by the continuous utilization of the system. These requirements include sustainment (provision of facilities, level support, support personnel, spare | | | | | |
| Requirements | parts, training, technical documentation, etc.), packaging, handling, shipping, transportation. | | | | | |
| Policies and | Define relevant and applicable organizational policies or regulatory requirements that could affect the operation or performance of the system (e.g. labor policies, reports to regulatory | | | | | |
| Regulations | agency, health or safety criteria, etc.). | | | | | |
| Cost and Schedule | Define, for example, the cost of a single exemplar of the system, the expected delivery date of the first exemplar, etc. | | | | | |
| Constraints | | | | | | |

Characteristics of Individual Requirements

| Characteristic | Description | | | | |
|----------------|--|--|--|--|--|
| Necessary | The requirement defines an essential capability, characteristic, constraint, and/or quality factor. If it is not included in the set of requirements, a deficiency in capability or characteristic will exist, | | | | |
| | which cannot be fulfilled by implementing other requirements | | | | |
| Appropriate | The specific intent and amount of detail of the requirement is appropriate to the level of the entity to which it refers (level of abstraction). This includes avoiding unnecessary constraints on the | | | | |
| | architecture or design to help ensure implementation independence to the extent possible | | | | |
| Unambiguous | The requirement is stated in such a way so that it can be interpreted in only one way | | | | |
| Complete | The requirement sufficiently describes the necessary capability, characteristic, constraint, or quality factor to meet the entity need without needing other information to understand the requirement | | | | |
| Singular | The requirement should state a single capability, characteristic, constraint, or quality factor | | | | |
| Feasible | The requirement can be realized within entity constraints (e.g., cost, schedule, technical, legal, regulatory) with acceptable risk | | | | |
| Verifiable | The requirement is structured and worded such that its realization can be proven (verified) to the customer's satisfaction at the level at which the requirement exists | | | | |
| Correct | The requirement must be an accurate representation of the entity need from which it was transformed | | | | |
| Conforming | The individual requirements should conform to an approved standard template and style for writing requirements, when applicable | | | | |

Characteristics of A Set of Requirements

| Characteristic | Description | | | | | | |
|----------------|--|--|--|--|--|--|--|
| Complete | The requirement set stands alone such that it sufficiently describes the necessary capabilities, characteristics, constraints, and/or quality factors to meet the entity needs without needing other | | | | | | |
| | information. In addition, the set does not contain any to be defined (TBD), to be specified (TBS), or to be resolved (TBR) clauses. | | | | | | |
| Consistent | The set of requirements contains individual requirements that are unique, do not conflict with or overlap with other requirements in the set, and the units and measurement systems they use | | | | | | |
| | are homogeneous. The language used within the set of requirements is consistent, i.e., the same word is used throughout the set to mean the same thing. | | | | | | |
| Feasible | The requirement set can be realized within entity constraints (e.g., cost, schedule, technical, legal, regulatory) with acceptable risk. (Note: Feasible includes the concept of "affordable".) | | | | | | |
| Comprehensible | The set of requirements must be written such that it is clear as to what is expected by the entity and its relation to the system of which it is a part. | | | | | | |
| Able to be | It must be able to be proven the requirement set will lead to the achievement of the entity needs within the constraints (such as cost, schedule, technical, legal and regulatory compliance). | | | | | | |
| validated | | | | | | | |

Pitfalls with definition of system requirements

| Pitfall | Description | | | | |
|-----------------------------|--|--|--|--|--|
| Insufficient Analysis of | If the receivers of the stakeholder requirements do not perform a sufficient critical analysis of them, the | | | | |
| Stakeholder Requirements | consequence could be difficulties translating them into system requirements and the obligation to come back to | | | | |
| | the stakeholders, losing time. | | | | |
| Insufficient Analysis of | The operational modes and operational scenarios are not sufficiently analyzed or defined by the person in charge | | | | |
| Operational Modes and | of writing the system requirements. Those elements allow the structuring of the system and its use early in the | | | | |
| Scenarios | engineering process and help the designer to remember functions and interfaces. | | | | |
| Incomplete Set of System | If the system requirements are not sufficiently precise and complete, there is a great risk that the design will not | | | | |
| Requirements | have the expected level of quality and that the verification and validation of the system will be delayed. | | | | |
| Lack of Verification Method | Delaying the capture of verification methods and events for each system requirement; identification of the | | | | |
| | verification approach for each requirement often provides additional insight as to the correctness and necessity of | | | | |
| | the requirement itself. | | | | |
| Missing traceability | Incorrect or missing traceability of each requirement, both to an upper-level "parent" requirement as well as | | | | |
| | allocation to an inappropriate system or system element. | | | | |

Proven Practices fo system requirements

| Practice | Description | | | | |
|-----------------|---|--|--|--|--|
| Involve | Involve the stakeholders as early as possible in the system requirements development process. | | | | |
| Stakeholders | | | | | |
| Presence of | Capture the rationale for each system requirement. | | | | |
| Rationale | | | | | |
| Always Complete | Check that stakeholder requirements are complete as much as possible before starting the definition of the system | | | | |
| before Starting | requirements. | | | | |
| Peer Reviews | Organize peer reviews of system requirements with applicable subject matter experts. | | | | |
| Modeling | Use modeling techniques as indicated in sections above. | | | | |
| Techniques | | | | | |
| Requirements | Consider using a requirements management tool, especially for more complex projects. This tool should have the capability | | | | |
| Management Tool | to trace linkages between system requirements to display relationships. A requirements management tool is intended to | | | | |
| | facilitate and support the systematic managing of system requirements throughout the project life cycle. | | | | |
| Measures for | Use typical measures for requirement engineering; for further information, refer to the Systems Engineering Leading | | | | |
| Requirement | Indicators Guide (Roedler et al. 2010). Both process and product measures should be used for requirements engineering. | | | | |
| Engineering | To get the desired insight to facilitate risk-managed requirements engineering, it may be necessary to use more than one | | | | |
| | measure based on the information needs (risks, objectives, issues) for the requirements. Useful measures include: | | | | |
| | Requirements Volatility | | | | |
| | Requirements Trends | | | | |
| | Requirements Verification Progress (plan vs. actual) | | | | |
| | Requirements Validation Progress (plan vs. actual) | | | | |
| | TBD and TBR Closure Per Plan | | | | |
| | Peer Review Defects | | | | |

-https://sebokwiki.org/wiki/Stakeholder_Requirements_Definition

NASA
Technical
Requirements
Definition
Process

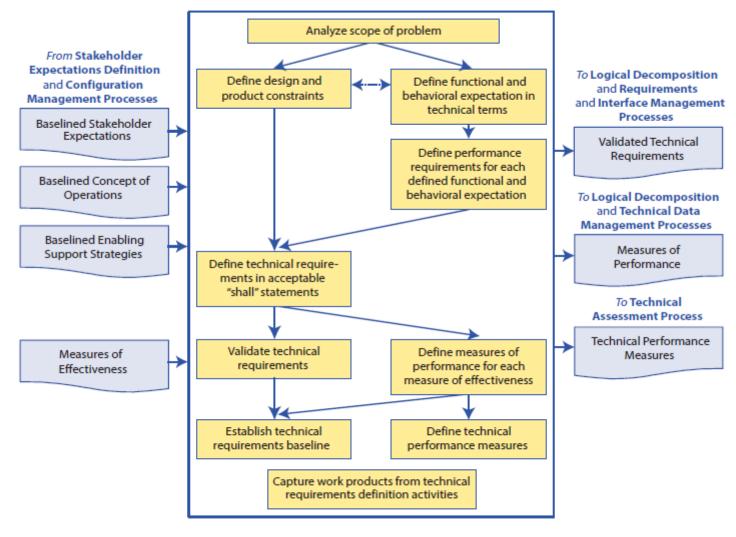


FIGURE 4.2-1 Technical Requirements Definition Process

Mission Authority Flowdown of Requirements Mission Objectives Programmatics: Mission Requirements Cost Schedule Constraints · Mission Classification Customer Implementing Organizations System Functional Requirements Environmental Institutional and Other Design Constraints Requirements and Guidelines Assumptions System Performance Requirements Subsystem A Subsystem Subsystem Subsystem X Functional and Functional and Performance Performance Requirements Requirements Allocated Derived Allocated Derived • • •

Requirements

FIGURE 4.2-3 The Flowdown of Requirements

Requirements

Requirements

Requirements

Requirements Metadata

TABLE 4.2-2 Requirements Metadata

| Item | Function |
|---------------------|---|
| Requirement ID | Provides a unique numbering system for sorting and tracking. |
| Rationale | Provides additional information to help clarify the intent of the requirements at the time they were written. (See "Rationale" box below on what should be captured.) |
| Traced from | Captures the bidirectional traceability between parent requirements and lower level (derived) requirements and the relationships between requirements. |
| Owner | Person or group responsible for writing, managing, and/or approving changes to this requirement. |
| Verification method | Captures the method of verification (test, inspection, analysis, demonstration) and should be determined as the requirements are developed. |
| Verification lead | Person or group assigned responsibility for verifying the requirement. |
| Verification level | Specifies the level in the hierarchy at which the requirements will be verified (e.g., system, subsystem, element). |

Propertybased requirements (PBRs)

- Property-based requirements (PBRs) are a type of system requirement that define desired properties or constraints the system must satisfy, without prescribing *exactly how* to achieve them.
- They are performance- and attribute-focused rather than specifying a fixed design solution, which makes them especially useful for complex, evolving, or innovative systems.

• Micouin, Patrice. Model Based Systems Engineering: Fundamentals and Methods (Focus) (6.4). Wiley press

Propertybased requirements (PBRs)

Define "what," not "how":

- They describe measurable system characteristics (e.g., range, durability, reliability, responsiveness).
- They avoid specifying implementation details.

Expressed in terms of attributes/properties: Examples:

- "The drone shall withstand wind speeds of up to 50 km/h."
- Not: "The drone shall use a carbon-fiber frame."

Allow design flexibility:

- Engineers can explore multiple design solutions that satisfy the same property.
- Supports innovation and trade-space exploration.

Often measurable and testable:

• Must have verification criteria (e.g., test, analysis, inspection).



Requirements Engineering

The process of establishing the services that the customer requires from a system and the constraints under which it operates and is developed.

The requirements themselves are the descriptions of the system services and constraints that are generated during the requirements engineering process.

What is a Requirement?

- It may range from a high-level abstract statement of a service or of a system constraint to a detailed mathematical functional specification.
- This is inevitable as requirements may serve a dual function
 - May be the basis for a bid for a contract therefore must be open to interpretation;
 - May be the basis for the contract itself therefore must be defined in detail;
 - Both these statements may be called requirements.



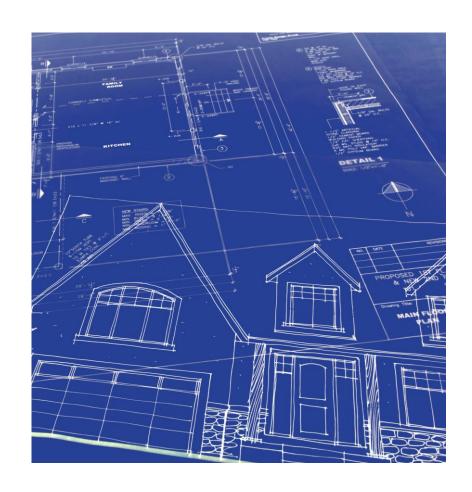


Requirements Development

- Collect top-level requirements from customers and stakeholders
- Develop an operations concept
- Derive lower-level requirements to support toplevel requirements and the operations concept
 - Writing requirements without a fully understood, agreed upon and documented operations concept will result in poor and misunderstood requirements, cost overruns and schedule slips.

Requirements Development

- Writing Good Requirements
- "Functional Requirements" What needs to be done
 - Functional Requirements are generally not Verified because
 Performance Requirements specify how good the function needs to be
 - Functional Requirements define a logical breakdown
- "Performance Requirements" How well it needs to be done
 - Performance Requirements are Verifiable
- Requirements Decomposition and Parent-Child Relationship
 - Requirements flow from higher to lower levels
 - Requirements at lower levels ("children") must trace from a higher-level requirement ("parent")
 - Eliminate any "orphan requirements" (requirements without parents)
- Add rationale or comments to the requirements to help trace "why" that requirement was created



Requirements Development

- Decompose the requirements in a hierarchical, parent-child relationship
 - Requirements flow from higher to lower levels
 - Requirements at lower levels ("children") must trace from a higher-level requirement ("parent")
 - Eliminate any "orphan requirements" (requirements without parents)
- Add rationale or comments to the requirements to help trace why that requirement was created

AN APPROACH

Starting point:

- Define the operational concept for my system (OpsCon/CONOPS)
- Use text and diagram-based modeling techniques that identify stakeholders, what they relate to the system, and ideally, their rationale.
- Use text and diagram-based modeling techniques (as appropriate) to describe the environment(s) in which the system will operate
- Use text and diagram-based modeling techniques (as appropriate) to describe the conditions under which the system will operate

SOLUTION SPACE ANALYSIS -ARTIFACTS

- 1. Articulation of Stakeholder Requirements (SHR)
 - Identification of stakeholders present at all phases of a solution's life
 - Definition of stakeholder needs, with respect to the solution, during all phases of a solution's life
- 2. Documented SHR validation criteria, supported by:
 - Understanding of what success means to your stakeholder(s). May include MOEs and MOSs.

SOLUTION SPACE ANALYSIS -ARTIFACTS

- 3. Documented SHR validation environment(s), supported by:
 - Understanding the environment(s) in which each stakeholder experiences the needs you're filling.
- 4. Documented SHR validation condition(s)
 - Understanding the condition(s) under which your stakeholders experiences the needs you're filling.
- 5. Documented interfaces representing interactions that the system/solution will have with external entities (e.g., other systems, people, externally-imposed requirements)

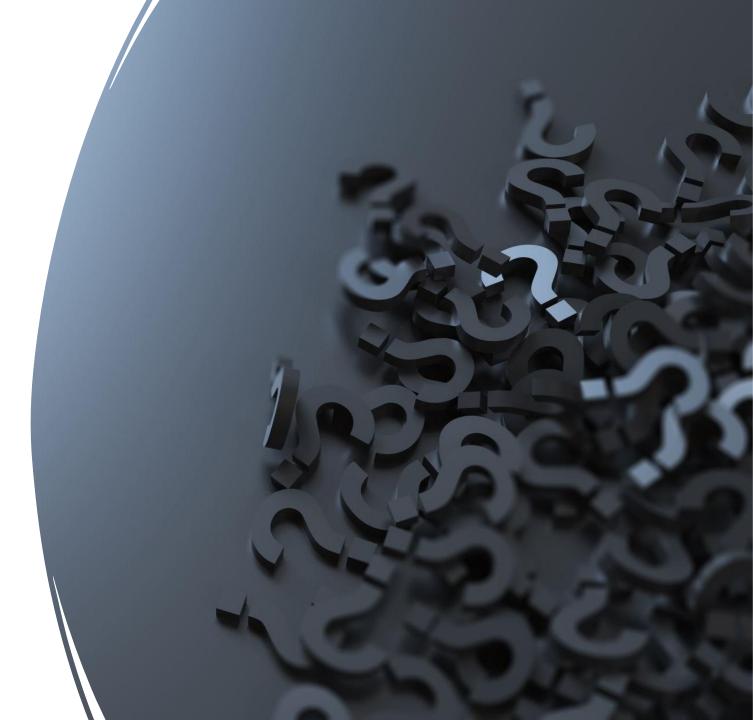
Requirements Management (Continued)

| ID | DESCRIPTION | REQUIREMENT | TRACED FROM | PERFORMANCE | MARGIN | Comments | REF |
|-----|--------------------------------|--|----------------|-------------|----------|---|-----------------|
| M1 | Mission Orbit | 575 +/-15 km Sun-synchronous dawn-dusk orbit | S3, S11, P3 | Complies | NA | Pegasus XL with HAPS provides required launch injection dispersion accuracy | F.2.c |
| M2 | Launch Vehicle | Pegasus XL with HAPS | P2, P4 | Complies | NA | | F.2.c |
| M3 | Observatory Mass | The NEXUS Observatory total mass shall not exceed 241 kg. | M1, M2 | 192.5 kg | 25.20% | | F.5.b |
| M4 | Data Acquisition Quality | The NEXUS mission shall deliver 95% data with better than 1 in 100,000 BER. | P1 | Complies | NA | Standard margins and systems baselined, formal system analysis to be completed by PDR | F.7 |
| M5 | Communication Band | The mission shall use S-band SQPSK at 5 Mbps for spacecraft downlink and 2 kbps uplink. | S12, P4 | Complies | NA | See SC27, SC28 and G1, G2 | F.3.f, F.7 |
| M7 | Tracking | MOC shall use NORAD two line elements for observatory tracking | P4 | Complies | NA | | F.7 |
| M8 | Data Latency | Data Latency shall be less than 72 hours | P12 | Complies | NA | | F.7 |
| M9 | Daily Data Volume | Accommodate average daily raw science data volume of 10.8 Gbits | P1, S12 | Complies | 12% | Marign based on funded ground contacts | F.3.e, F.7 |
| M10 | Ground Station | The Mission Shall be Compatible With the Rutherford Appleton Laboratory Ground Station and the Poker Flat Ground Station | P1 | Complies | NA | | F.7 |
| M11 | Orbital Debris (Casualty Area) | Design NEXUS observatory for demise upon reentry with < 1/10,000 probability of injury | P3 | 1/51,000 | 400% | See Orbital Debris Analysis in Appendix M-6 | F.2.e, App.6 |
| M12 | Orbital Debris (Lifetime) | Design NEXUS observatory for re-entry <25 years after end of mission | P3 | < 10 years | 15 years | See Orbital Debris Analysis in Appendix M-6 | F.2.e, App.6 |

Table 13

Different levels of abstraction

- User requirements
 - Usually, the first attempt for the description of the requirements
 - Services and constraints of the system
 - In natural language or diagrams
 - Readable by everybody
 - Serve operational objectives
- System requirements
 - Services and constraints of the system in detail
 - Useful for the design and development
 - Precise and cover all cases
 - Structured presentation



Stakeholder Needs and Requirements Definition process

- The purpose of the Stakeholder Needs and Requirements Definition process is to define the stakeholder requirements for a system that can provide the capabilities needed by users and other stakeholders in a defined environment.
- ISO/IEEE 15288

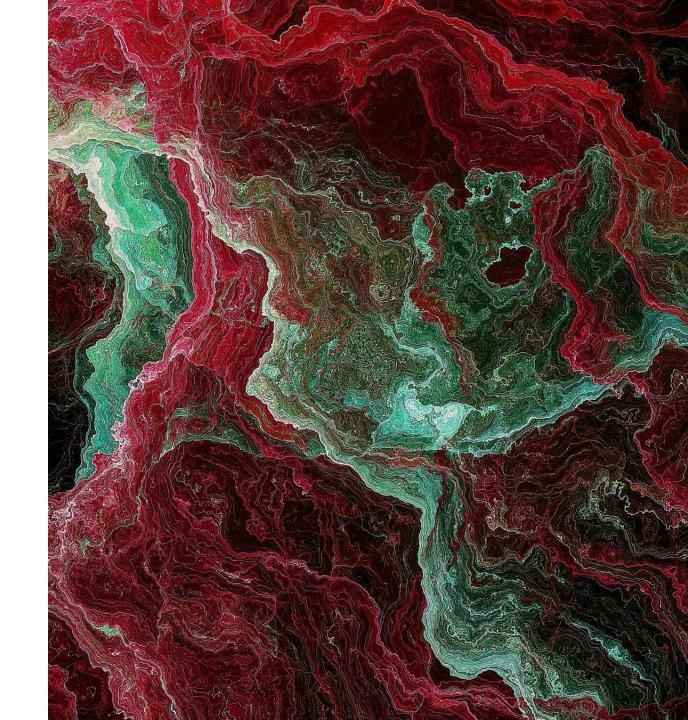
Stakeholder Needs and Requirements Definition process

- The purpose of the Stakeholder Needs and Requirements Definition process is to define the stakeholder requirements for a system that can provide the capabilities needed by users and other stakeholders in a defined environment.
- ISO/IEEE 15288



System Requirements Definition process

- The purpose of the System Requirements Definition process is to transform the stakeholder, user-oriented view of desired capabilities into a technical view of a solution that meets the operational needs of the user.
- ISO/IEEE 15288





Functional and Non-functional requirements

- Functional requirements
 - Statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations.
 - May state what the system should not do.
- Non-functional requirements
 - Constraints on the services or functions offered by the system such as timing constraints, constraints on the development process, standards, etc.
 - Often apply to the system as a whole rather than individual features or services.
- Domain requirements
 - Constraints on the system from the domain of operation



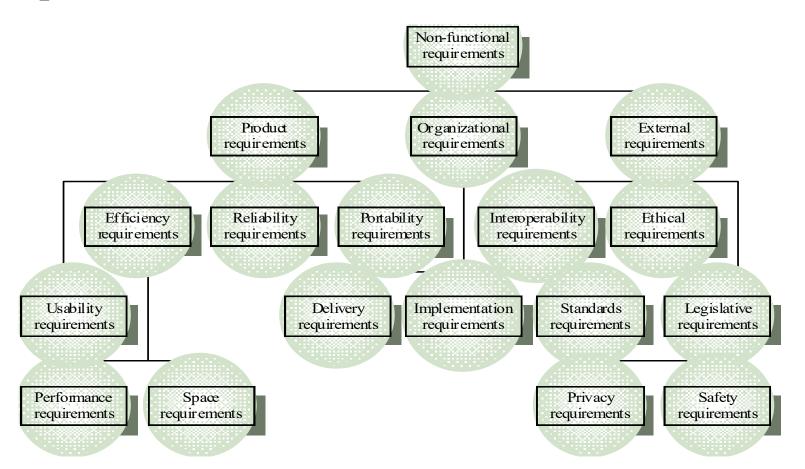
Functional Requirements

- Describe functionality or system services.
- Depend on the type of software, expected users and the type of system where the software is used.
- Functional user requirements may be high-level statements of what the system should do.
- Functional system requirements should describe the system services in detail.
- Essentially, these are the 'what's' of the system that we often refer to. These are not 'all that there is,' but these should describe the overall functionality of the system.

Non-functional requirements

- They can be further categorized into:
 - Product requirements
 - Product behavior
 - Ex: Timing, performance, memory, reliability, portability, usability
 - Organizational requirements
 - Policies and procedures in the customer's and developer's organizations
 - Example: Process requirements, implementation requirements, delivery requirements
 - External requirements
 - Factors externals to the system and the development process
 - Example: Interoperability, legislation, ethics
- Non-functional requirements may be more critical than functional requirements.

NON-FUNCTIONAL REQUIREMENTS



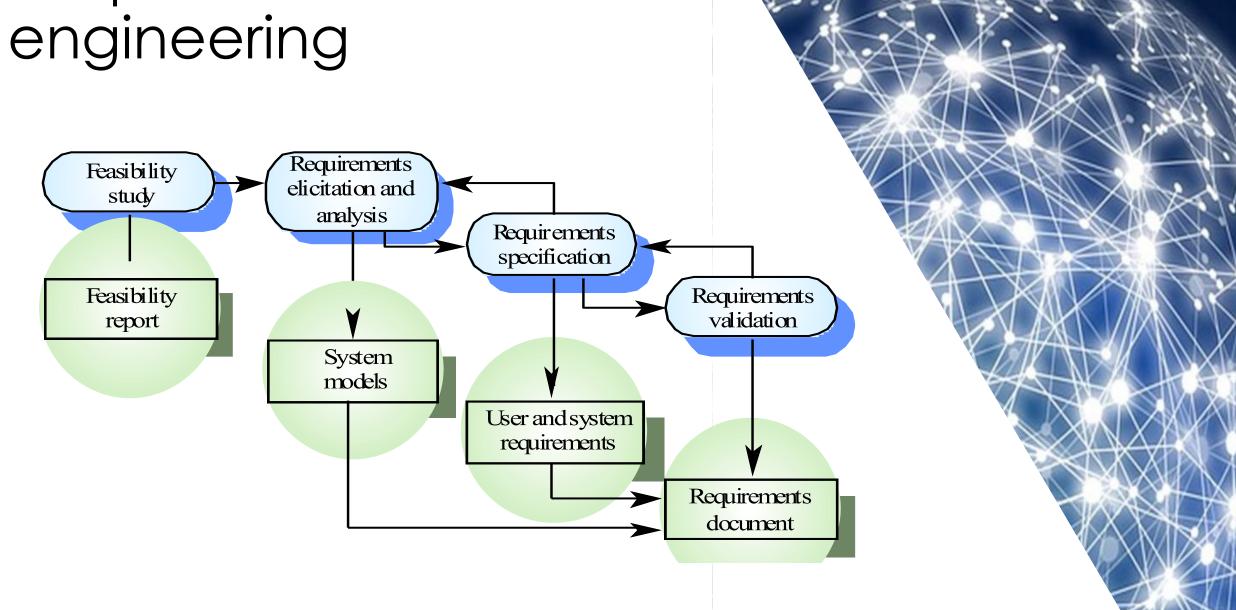
METRICS FOR SPECIFYING NONFUNCTIONAL REQUIREMENTS

| Property | Measure |
|-------------|---|
| Speed | Processed transactions/second User/event response time Screen refresh time |
| Size | Mbytes Number of ROM chips |
| Ease of use | Training time Number of help frames |
| Reliability | Mean time to failure (MTTF) Probability of unavailability Rate of failure occurrence Availability |
| Robustness | Time to restart after failure (MTTR) Percentage of events causing failure Probability of data corruption on failure |
| Portability | Percentage of target dependent statements Number of target systems |

Requirement engineering

- 5 important activities:
 - Feasibility study
 - Requirement's elicitation and analysis
 - Requirement's documentation
 - Requirement's validation
 - Requirements management

Requirement



Feasibility studies

- A feasibility study decides whether or not the proposed system is worthwhile
- A short-focused study that checks
 - If the system contributes to organisational objectives
 - If the system can be engineered using current technology and within budget
 - If the system can be integrated with other systems that are used

Elicitation and analysis

- Sometimes called requirements elicitation or requirements discovery
- Involves technical staff working with customers to find out about the application domain, the services that the system should provide and the system's operational constraints
- May involve end-users, managers, engineers involved in maintenance, domain experts, trade unions, etc. These are called stakeholders

Requirements Elicitation

- Problems of Requirement Elicitation
 - **Problems of scope:** The boundary of system is ill-defined. Or unnecessary details are provided.
 - Problems of understanding: The users are not sure of what they need, and don't have full understanding of the problem domain.
 - Problems of volatility: the requirements change overtime.

Requirements Elicitation

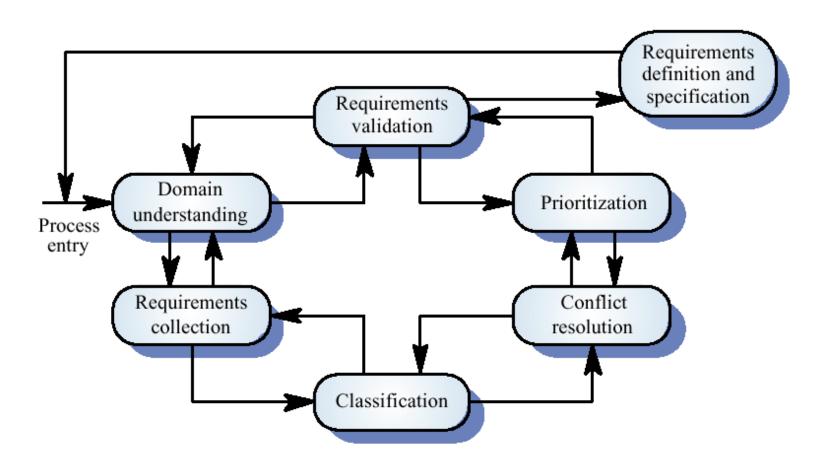
• Guidelines of Requirements Elicitation

- Assess the operational and technical feasibility for the proposed system
- Identify the people who will help specify requirements.
- Define the technical environment (e.g., computing architecture, operating system, telecommunication needs) into which the system or product will be placed
- Identify "domain constraints" (i.e., characteristics of the operational environment specific to the application domain) that limit the functionality or performance of the system or product to build
- Define one or more requirements elicitation methods (e.g., interviews, team meetings, ..etc)
- Solicit participation from many people so that requirements are defined from different point of views.
- Create usage scenarios of use cases to help customers/ users better identify key requirements.

Problems of requirements analysis

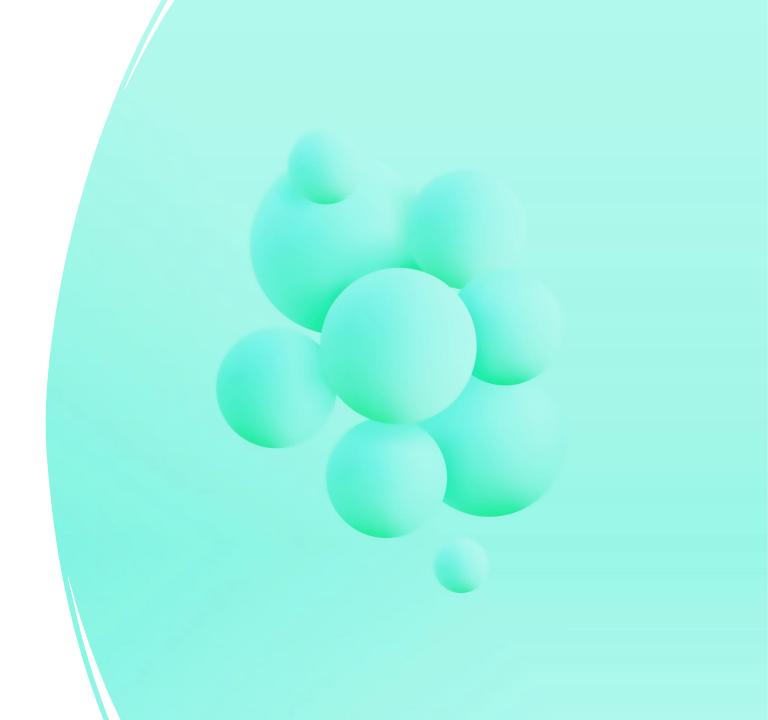
- Stakeholders don't know what they really want
- Stakeholders express requirements in their own terms
- Different stakeholders may have conflicting requirements
- Organisational and political factors may influence the system requirements
- The requirements change during the analysis process. New stakeholders may emerge and the operational environment change

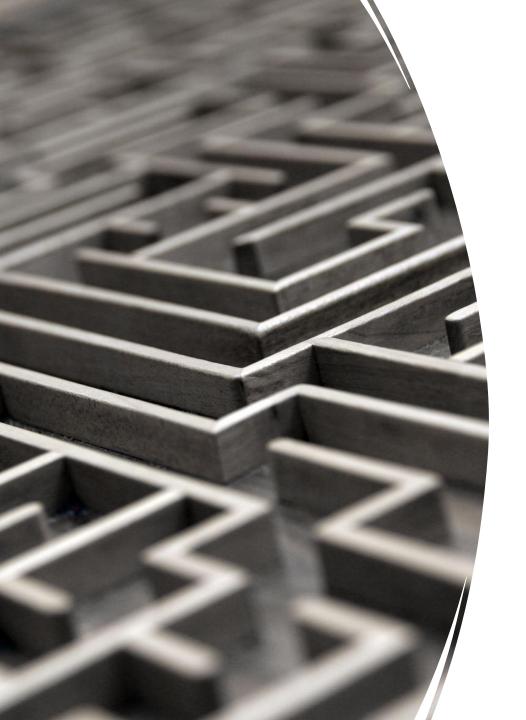
THE REQUIREMENTS ANALYSIS PROCESS



Process activities

- Domain understanding
- Requirements collection
- Classification
- Conflict resolution
- Prioritization
- Requirements checking





Viewpoint-oriented elicitation

- Stakeholders represent different ways of looking at a problem or problem viewpoints
- This multi-perspective analysis is important as there is no single correct way to analyse system requirements



Types of viewpoint

- Data sources or sinks
 - Viewpoints are responsible for producing or consuming data. Analysis involves checking that data is produced and consumed and that assumptions about the source and sink of data are valid
- Representation frameworks
 - Viewpoints represent particular types of system model. These may be compared to discover requirements that would be missed using a single representation.
 Particularly suitable for real-time systems
- Receivers of services
 - Viewpoints are external to the system and receive services from it. Most suited to interactive systems

External viewpoints



Natural to think of end-users as receivers of system services



Viewpoints are a natural way to structure requirements elicitation



It is relatively easy to decide if a viewpoint is valid



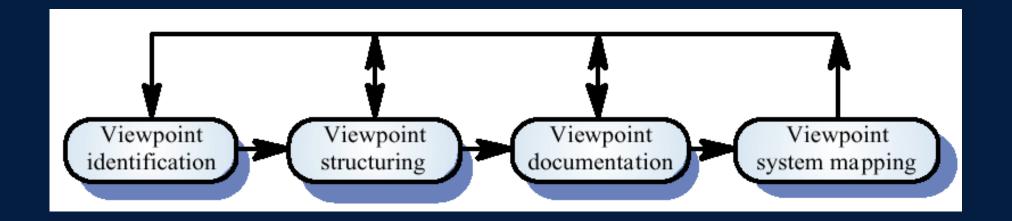
Viewpoints and services may be used to structure nonfunctional requirements

Methodbased analysis

- Widely used approach to requirements analysis.
 Depends on the application of a structured method to understand the system
- Methods have different emphases. Some are designed for requirements elicitation, others are close to design methods
- A viewpoint-oriented method (VORD) is used as an example here. It also illustrates the use of viewpoints

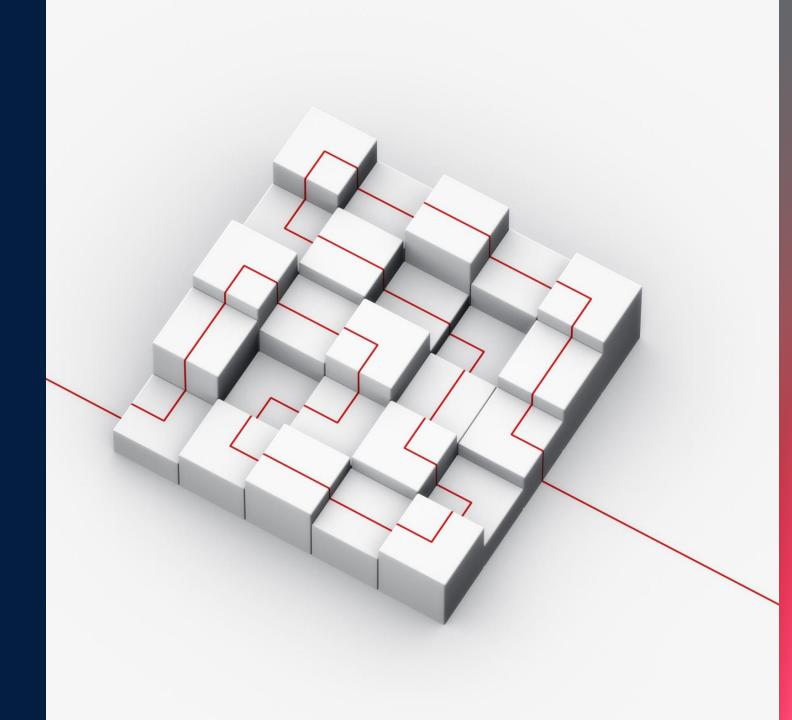
The VORD method

Viewpoint-Oriented Requirements Definition



VORD process model

- Viewpoint identification
 - Discover viewpoints which receive system services and identify the services provided to each viewpoint
- Viewpoint structuring
 - Group related viewpoints into a hierarchy. Common services are provided at higher-levels in the hierarchy
- Viewpoint documentation
 - Refine the description of the identified viewpoints and services
- Viewpoint-system mapping
 - Transform the analysis to an objectoriented design



Requirements validation techniques

- Requirements reviews
 - Systematic manual analysis of the requirements
- Prototyping
 - Using an executable model of the system to check requirements.
- Test-case generation
 - Developing tests for requirements to check testability
- Automated consistency analysis
 - Checking the consistency of a structured requirements description



Requirements reviews

- Regular reviews should be held while the requirements definition is being formulated
- Both client and contractor staff should be involved in reviews
- Reviews may be formal (with completed documents) or informal. Good communications between developers, customers and users can resolve problems at an early stage

Review checks

- Verifiability. Is the requirement realistically testable?
- Comprehensibility. Is the requirement properly understood?
- Traceability. Is the origin of the requirement clearly stated?
- Adaptability. Can the requirement be changed without a large impact on other requirements?



Classification of requirements

- Mutable requirements
 - Requirements that change due to the system's environment
- Emergent requirements
 - Requirements that emerge as understanding of the system develops
- Consequential requirements
 - Requirements that result from the introduction of the computer system
- Compatibility requirements
 - Requirements that depend on other systems or organisational processes



Requirements Traceability Matix (Missile tracking system)

| Req. | Requirement Description | Category | Source | Design Spec / Solution | Implementation (Module / Component) | Verification Method | Test Case ID | Status |
|-----------|--|----------------------|--------------------------|---|-------------------------------------|------------------------------------|--------------|--------|
| R- 001 | System shall detect incoming missiles within 500 km range. | Functional | DoD Spec MIL- STD-601 | Radar subsystem with phased array antennas | Tracking Radar Module | Simulation & Live Range Test | TC-001 | Open |
| R- 002 | System shall classify target type (missile, aircraft, drone). | Functional | Stakeholder Req | Signal processing & Al-based classifier | Target Classification Module | Algorithm Verification, Simulation | TC-002 | Open |
| R- 003 | System shall track up to 50 simultaneous targets. | Performance | DoD Spec | Multi-target tracking algorithm | Tracking Engine | Stress Test & Simulation | TC-003 | Open |
| R- 004 | Tracking accuracy shall be within ±5 meters. | Performance | Defense Contract Spec | Kalman Filter & Sensor Fusion | Guidance & Navigation Module | Accuracy Bench Test | TC-004 | Open |
| R- 005 | System shall provide real-time operator alerts within 1 second of detection. | Functional | Ops Requirement | Real-time comms interface | Command & Control UI | Response Time Testing | TC-005 | Open |
| R- 006 | System shall operate under electronic countermeasures (jamming). | Reliability | MIL-STD-464 | ECCM techniques (frequency hopping, adaptive filters) | Radar & Communication Modules | EMI/EMC Testing | TC-006 | Open |
| R- 007 | System shall log tracking data for 30 days. | Non- Functional | Ops Requirement | Encrypted storage module | Data Logging & Storage | Data Retention Verification | TC-007 | Open |
| R- 008 | System shall comply with MIL-STD-882E safety requirements. | Safety | Safety Spec | Safety certification protocols | All Subsystems | Compliance Audit | TC-008 | Open |
| R- 009 | System shall be interoperable with allied defense systems (NATO Link-16). | Interoperabili ty | NATO Spec | Link-16 communication interface | Comm Subsystem | Interoperability Testing | TC-009 | Open |
| R- 010 | System shall provide 99.9% uptime in mission-critical scenarios. | Reliability | Contract SLA | Redundant hardware/software | Full System | Reliability/Failover Testing | TC-010 | Open |

Requirements Traceability Matix (Missile tracking system)

| Req. ID | Requirement Description | Category | Source | | Implementation (Module / Component) | Verification Method | Test Case ID | Status |
|---------|---|----------------------|-----------------------|---|--|-----------------------------------|--------------|--------|
| 1 | System shall detect and engage aircraft targets within 25 km range. | Functional | Naval Defense Spec | | | Simulation & Live Firing Test | TC-001 | Open |
| 1 | System shall track and engage multiple targets simultaneously (min 10). | Performance | Naval Defense Spec | Multi-target tracking radar & algorithms | | Stress Test & Simulation | TC-002 | Open |
| | Gun shall have a firing rate of at least 200 rounds per minute. | Performance | Ops Requirement | High-speed naval gun mount | Gun Mount & Firing Mechanism | Firing Rate Test | TC-003 | Open |
| 1 | System shall achieve a target hit probability of 90% at 10 km range. | Performance | Ops Requirement | Fire control computer with ballistic calculations | Guidance & Fire Control Computer | · ' | TC-004 | Open |
| 1 | System shall provide automated target acquisition and tracking. | Functional | Stakeholder Req | l . | Autonomous Tracking Subsystem | Functional Demonstration | TC-005 | Open |
| 1 | System shall operate in adverse weather and sea state conditions. | Reliability | MIL-STD-810G | 00 | | Environmental Testing | TC-006 | Open |
| 1 | System shall integrate with ship's combat management system (CMS). | Interoperabilit y | · · | CMS interface with standardized protocols | Communication Interface | Interoperability Testing | TC-007 | Open |
| 1 | System shall comply with naval safety standards (NAVSEA, MIL-STD). | Safety | Safety Spec | Safety certification protocols | All Subsystems | Compliance Audit | TC-008 | Open |
| | System shall allow manual override by operator at any time. | 1 | Ops Requirement | Manual targeting & fire control station | | Operational Testing | TC-009 | Open |
| 1 | System shall ensure ammunition storage safety and blast protection. | Safety | Safety Spec | 1 | l | Safety Inspection & Blast Test | TC-010 | Open |
| | | | | | | | | |

Controls



Inputs

- · Organization strategic plan
- ConOps
- · Source documents
- · Life cycle constraints
- · Project constraints
- Stakeholder requirements traceability

Activities

- Prepare for business or mission analysis
- Define the problem or opportunity space
- Characterize the solution space
- Evaluate alternative solution classes
- Manage the business or mission analysis

Outputs

- Business or mission analysis strategy
- Major stakeholder identification
- Preliminary life cycle concepts
- Problem or opportunity statement
- · Business requirements
- · Alternative solution classes
- Preliminary validation criteria
- · Preliminary MOE needs
- Preliminary MOE data
- Business requirements traceability
- Business or mission analysis record

MISSION ANALYSIS PROCESS



Enablers

Stakeholder needs and requirements definition process

Controls



Inputs

- · Source documents
- · Project constraints
- Major stakeholders identification
- Preliminary life cycle concepts
- Problem or opportunity statement
- · Business requirements
- Alternative solution classes
- Preliminary validation criteria
- Validated requirements
- · Preliminary MOE needs
- · Preliminary MOE data
- Business requirements traceability
- · Life cycle constraints
- · Stakeholder needs
- System requirements traceability

Activities

- Prepare for stakeholder needs and requirements definition
- · Define stakeholder needs
- Develop the operational concept and other life cycle concepts
- Transform stakeholder needs into stakeholder requirements
- Analyze stakeholder requirements
- Manage the stakeholder needs and requirements definition

Outputs

- Stakeholder needs and requirements definition strategy
- · Life cycle concepts
- System function identification
- · Stakeholder requirements
- · Validation criteria
- · MOE needs
- MOE data
- Stakeholder requirements traceability
- · Initial RVTM
- Stakeholder needs and requirements definition record



Enablers

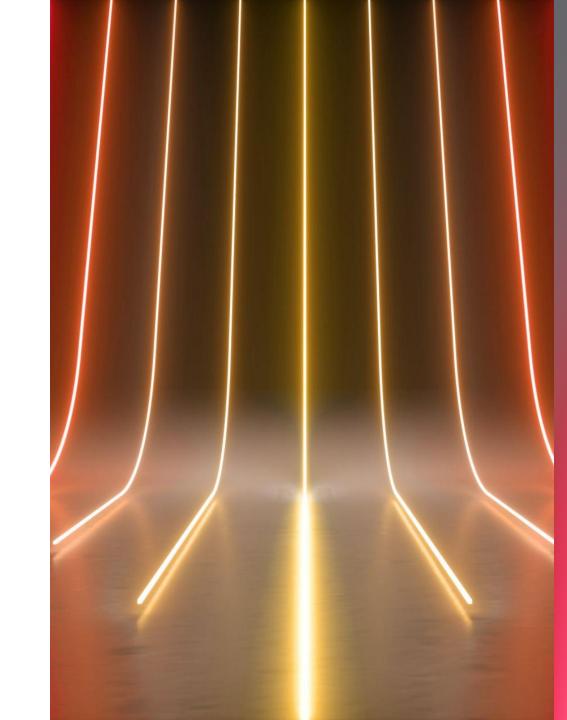
IMPORTANT SYSTEMS ENGINEERING DOCUMENTS

- Stakeholder requirements specification (StRS)
- Mission Element Needs Statement (MENS)
- Systems Engineering Management Plan (SEMP)
- System Requirements Document (SRD)
- System Requirements Validation Document (SRVD)
- System Description Document (SDD)
- A System Requirements
 Specification (SyRS)



NEEDS

- https://www.sebokwiki.org/wiki/Stakeholder Needs and Requirements
- Real needs are those that lie behind any perceived needs (see below); they are conditioned by the context in which people live. As an example, a generic need could be the ability to identify infectious diseases easily. Often, real needs appear to be simple tasks.
- Perceived needs are based on a person's awareness that something is wrong, that something is lacking, that improvements could be made, or that there is an operational need not being met. Perceived needs are often presented as a list of organized expectations resulting from an analysis of the usage conditions for the considered action. Following from the infectious disease example above, the real need might be perceived as a need to carry out medical tests in particular circumstances (laboratories, points of care, hospitals, and/or human dispensaries). Since the real need is seldom clearly expressed, richness of the knowledge of the perceived needs is used as a basis for potential solutions. This step has to be as complete as possible to cover all the contexts of use.



NEEDS

- Expressed needs originate from perceived needs in the form of generic actions or constraints and are typically prioritized. In the example, if safety is the primary concern, the expressed need to protect the operator against contamination may take priority over other expressed needs such as assist in the execution of tests. When determining the expressed needs, the analysis of the expected mission or services in terms of operational scenarios takes place.
- **Retained needs** are selected from the expressed needs. The selection process uses the prioritization of expressed needs to achieve a solution or to make attaining solutions feasible. The retained needs allow the consideration of potential solutions for a SoI. These retained stakeholder intentions do not serve as stakeholder requirements, since they often lack definition, analysis, and possibly consistency and feasibility. Using the concept of operations to aid the understanding of the stakeholder intentions at the organizational level and the system operational concept from the system perspective, requirements engineering leads stakeholders from those initial intentions to structured and more formal stakeholder requirement statements, ISO/IEC/IEEE 29148 Systems and software engineering - Requirements engineering (ISO 2011). Characteristics of good requirements can be found in (ISO 2011). Exploration of potential solutions must start from this step. The various solutions suggested at this step are not yet products but describe means of satisfying the stakeholder requirements. Each potential solution imposes constraints on the potential future Sol.



NEEDS

Specified needs, are the translation of the stakeholder needs to represent the views of the supplier, keeping in mind the potential, preferred, and feasible solutions. Specified needs are translated into system requirements. Consistent practice has shown this process requires iterative and recursive steps in parallel with other life cycle processes through the system design hierarchy (ISO 2011).

Realized needs are the product, service, or enterprise realized, taking into account every specified need (and hence, the retained needs).

Validating Requirements

- Is each requirement consistent with the overall objective for the system/product?
- Have all requirements been specified at the proper level of abstraction? That is, do some requirements provide a level of technical detail that is inappropriate at this stage?
- Is the requirement really necessary or does it represent an add-on feature that may not be essential to the objective of the system?
- Is each requirement bounded and unambiguous?
- Does each requirement have attribution? That is, is a source (generally, a specific individual)
 noted for each requirement?
- Do any requirements *conflict* with other requirements?
- Is each requirement achievable in the technical environment that will house the system or product?
- Is each requirement testable, once implemented?
- Does the requirements model properly reflect the information, function and behavior of the system to be built.

Requirements Specifications

- A Software Requirements Specification (SRS)
 - The software requirement specification document enlists all necessary requirements for project development. To derive the requirements, we need to have clear and thorough understanding of the products to be developed.
 - A general organization of an SRS is as follows:
 - Introduction
 - Purpose, Scope, Definitions, System Overview, References
 - Overall Description
 - Product Perspective, Product functions, User characteristics, constraints, assumptions and dependencies.
 - Specific Requirements
 - External Interface requirements, functional requirements, performance requirements, design constraints, logical database requirement, software system attributes.



SOLUTION SPACE ANALYSIS



IDENTIFY WHO THE STAKEHOLDERS ARE



DETERMINE WHAT EACH STAKEHOLDER
NEEDS THE SOLUTION TO DO (OR NOT DO)
TO OPERATE HOW THEY WANT TO,
INFLUENCE AS THEY WANT OR NEED TO, OR
BE INFLUENCED AS THEY WANT TO NEED TO

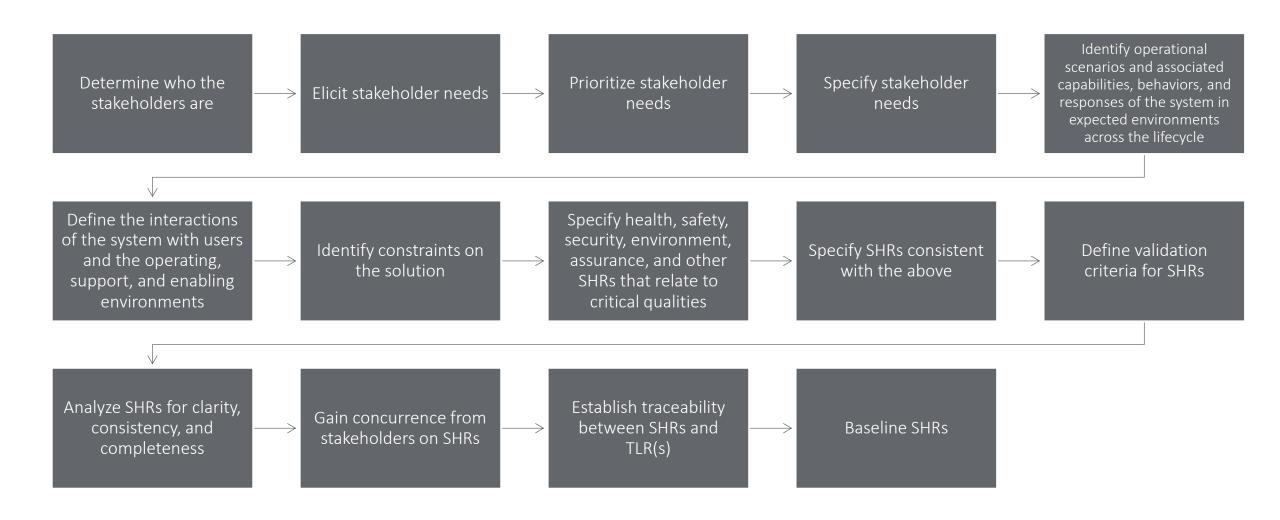


DETERMINE HOW OUR SOLUTION (AS A HOLISTIC ENTITY) INTERFACES AND INTERACTS WITH OTHER ENTITIES IN ITS OPERATING ENVIRONMENT(S)



ANALYZE DEFINED SOLUTION (TO THIS POINT) FOR DOWNSTREAM IMPACTS, RISKS, AND OPPORTUNITIES

INCOSE's Way



Anti-Aircraft Measures of Effectiveness

According to Boord, Warren J.; Hoffman, John B.. Air and Missile Defense Systems Engineering (p. 61). CRC Press

Reaction time—How

Firepower—How Defense penetration technique resistance—When Environmental resistance—When

Continuous availability—When

Contiguous coverage in theater—Where

Anti-Aircraft Measures of Effectiveness

- My list (using multiple sources)
- Probability of Intercept (Pi): The probability that the system successfully detects, tracks, engages, and intercepts the target.
- Time to Intercept (TTI):The time from detection or launch to the point of intercept.
- Coverage Area / Footprint: The geographic area the system can protect.
- Survivability / Redundancy: The system's ability to withstand or recover from attacks or failures.
- False Alarm Rate: The rate of false detections or unnecessary launches.
- Discrimination Capability: The ability to distinguish real warheads from decoys/chaff/flares.
- Mobility / Deployability: How quickly and flexibly a system can be moved and set up.
- Cost per Kill / Mission Cost: The average cost of intercepting a single threat, factoring interceptor use, operations, and logistics.
- System Availability / Uptime: The percentage of time the system is fully functional and ready.

IMPORTANT SYSTEMS ENGINEERING DOCUMENTS

Stakeholder requirements specification (StRS)

Mission Element Needs Statement (MENS)

Systems Engineering Management Plan (SEMP)

System Requirements Document (SRD)

System Requirements Validation Document (SRVD)

System Description Document (SDD)

A System Requirements Specification (SyRS)

DOCUMENTS

• Wiley provides a description of all documents https://onlinelibrary.wiley.com/doi/pdf/10.1002/97 80470413791.app1



STAKEHOLDER REQUIREMENTS SPECIFICATION (STRS)

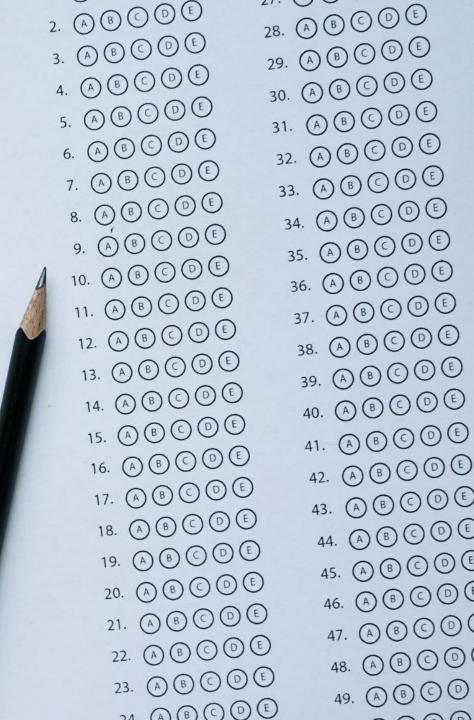
StRS Template Table of Content
Adapted from ISO/IEC/IEEE 29148:2011, Section 9.4

1. Introduction

- 1.1 Purpose
- 1.2 Document conventions
- 1.3 Intended audience
- 1.4 Additional information
- 1.5 Contact information/SyRS team members
- 1.6 References

2. System overview

- 2.1 System context
- 2.2 System functions
- 2.3 User characteristics



STAKEHOLDER REQUIREMENTS **SPECIFICATION (STRS)**

- 3. Functional Requirements
- 4. Usability Requirements
- 5. Performance Requirements
- 6. System Interfaces
- 7. System Operations
- 7.1 Human system integration requirements
 7.2 Maintainability
 7.3 Reliability

- 8. System modes and states

- 9. Physical characteristics9.1 Physical requirements9.2 Adaptability requirements



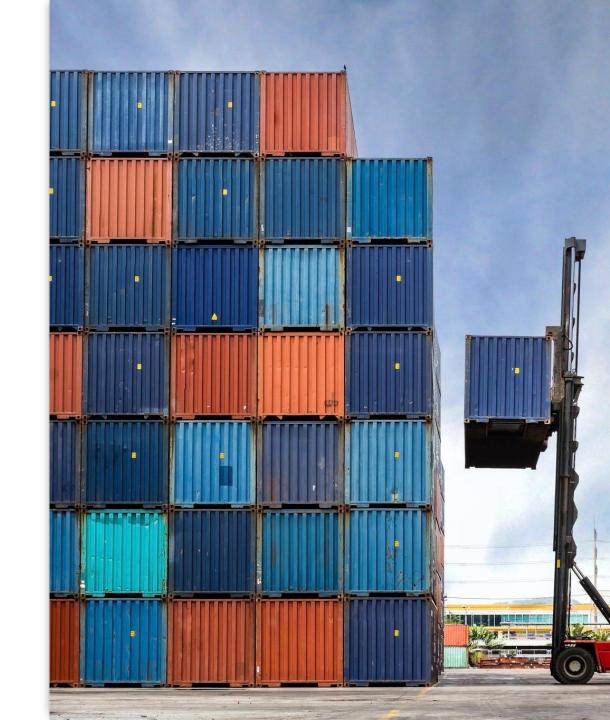
STAKEHOLDER REQUIREMENTS SPECIFICATION (STRS)

- 10. Environmental conditions
- 11. System security
- 12. Information management
- 13. Policies and regulations
- 14. System life cycle sustainment
- 15. Packaging, handling, shipping and transportation
- 16. Verification
- 17. Assumptions and dependencies
- 18. Other Requirements

Appendix A: Terminology/Glossary/Definitions list

Appendix B: Requirements Traceability Matrix

This matrix traces each requirement to its parent/source requirement or material.



MISSION ELEMENT NEEDS STATEMENT (MENS)

Mission Needs Statement (MNS) is a U.S. Department of Defense type of document which identifies capability needs for a program to satisfy by a combination of solutions Directives from the DoD can be found at

https://www.directives.doe.gov/directives-documents/400-series/0413.3-EGuide-17/@@images/file



A SYSTEMS ENGINEERING MANAGEMENT PLAN (SEMP)

• A Systems Engineering Management Plan (SEMP) is a document that addresses a contractor's overall systems engineering management approach. It provides unique insight into the application of a contractor's standards, capability models, configuration management, and toolsets to their organization. This is different from a Systems Engineering Plan (SEP) which should address System Engineering (SE) aspects on a particular program or project.



A SYSTEMS ENGINEERING MANAGEMENT PLAN (SEMP)

• "The SEMP provides the specifics of the technical effort and describes what technical processes will be used, how the processes will be applied using appropriate activities, how the project will be organized to accomplish the activities, information flow within the organization, decision making structure, and the resources required for accomplishing the activities. The process activities are driven by the critical events during any phase of a life cycle (including operations) that set the objectives and work product outputs of the processes and how the processes are integrated. The SEMP provides the communication bridge between the project management team and the engineering discipline teams" -NASA https://www.nasa.gov/consortium/SystemsEngineeringMa nagementPlanTechnicalContent

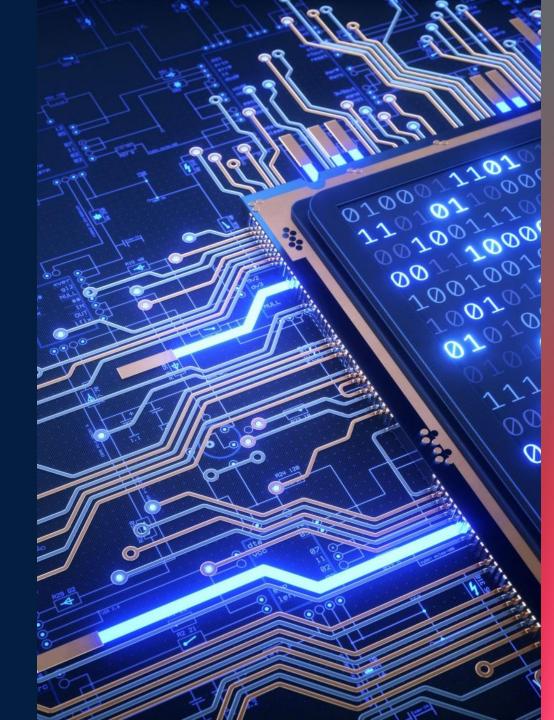


System Requirements Document (SRD)

Template found at http://bluehawk.monmouth.edu/rclayton/web-pages/s00-325/srs-template.html

"System Requirements Document is also known as System Requirements Specifications. System requirements document is a set of documentation that describes the behavior and features of a software or system. It comprises of various elements that attempt to characterize the functionality needed by the client to satisfy their users. In other words, the system requirements document (SRD) describes the system-level performance and functional requirements for a system."

- https://www.javatpoint.com/system-requirements-document



System Requirements Document (SRD)

- According to the IEEE standards, SRDs must cover the following important topics.
- Quality
- Security/Privacy
- Functional capabilities
- Safety
- Performance levels
- Constraints and Limitations
- Data Structure/Elements
- Reliability
- Interfaces



A System Requirements Specification (SyRS)

Template/Example
https://www.its.dot.gov/meetings/pdf/Cores
vstem_SF_SvRS_RevA%20(2011-06-13).pdf

"A System Requirements Specification (SRS) (also known as a Software Requirements Specification) is a document or set of documentation that describes the features and behavior of a system or software application. It includes a variety of elements (see below) that attempts to define the intended functionality required by the customer to satisfy their different users."

https://www.inflectra.com/ideas/topic/requi rements-definition.aspx 2. A B C D E 28. (A) (B) (C) (D) (E) 3. A B C D E 29. A B C O E 4. (A) (B) (C) (D) (E) 30. (A) (B) (C) (D) (E) 5. A B C D E 31. (A) (B) (C) (D) (E) 6. ABCDE 32. (A) (B) (C) (D) (E) 7. (A) (B) (C) (D) (E) 33. (A) (B) (C) (D) (E) 8. (A) (B) (C) (D) (E) 34. (A) (B) (C) (D) (E) 9. A B C O E 35. ABCDE 10. (A) (B) (C) (D) (E) 36. (A) (B) (C) (D) (E) 11. (A) (B) (C) (D) (E) 37. A B C D E 12. (A) (B) (C) (D) (E) 38. A B C D E 13. (A) (B) (C) (D) (E) 39. (A) (B) (C) (D) (E) 14. A B C D E 40. (A) (B) (C) (D) (E) 15. A B C D E 41. (A) (B) (C) (D) (E) 16. (A) (B) (C) (D) (E) 42. A B C D E 17. ABCDE 43. (A) (B) (C) (D) (E) 18. (A) (B) (C) (D) (E) 44. A B C D E 19. (A) (B) (C) (D) (E) 45. A B C D 20. (A) (B) (C) (D) (E) 46. A B C D 21. (A) (B) (C) (D) (E) 47. A B C 22. A B C D E 48. (A) (B) 23. (A) (B) (C) (D) (E) 21 ABCOE

RESOURCES

NASA SEMP Guideline

https://www.nasa.gov/consortium/SystemsEngineeringManagementPlanTechnicalContent

SEMP Template

https://www.fhwa.dot.gov/cadiv/segb/views/document/sections/section8/8 4 2.cfm

SeBOK StRS

https://www.sebokwiki.org/wiki/Stakeholder Needs and Requirements

SRD Example

https://wiki.services.eoportal.org/tiki-download wiki attachment.php?attId=634

SRD Example

https://man.fas.org/dod-101/sys/ac/equip/srd_an-alq-172-srda2.htm

SDD Template

https://www.cs.fsu.edu/~lacher/courses/COP3331/sdd.html

NASA SDD Template

https://www.its.dot.gov/research_archives/msaa/pdf/MSAA_SystemDesignFINAL.pdf