RE Framework

Elicitation
Achieve progress in the **specification dimension** by eliciting new requirements as well as detailed information about existing requirements.

- Elicit all requirements at the level of detail for the system to be developed.
Table of Contents

• Eliciting Requirements - Examples
• Criteria for good requirements
Table of Contents

• Eliciting Requirements - Examples
• Criteria for good requirements
Requirement Examples

Various items that can comprise requirements:

1. External Interface Requirements
   Logical characteristics of each interface between the software product and the
   1. User Interfaces (required screen formats, page or window layouts, report contents, etc)
   2. Hardware components (number of ports, instruction sets, etc)
   3. Software Interfaces (using other required software products and interfaces with the system eg (e.g., data management system integration)
   4. Communication Interfaces
   ... of the system

2. Functional Requirements
   Actions that must take place in the software in accepting and processing the inputs and in processing and generating the outputs.
   Generally listed as “shall” statements starting with “The system shall…”

3. Performance Requirements
   Numerical requirements placed on the software or on human interaction with the software as a whole
   *Remember to state this in measurable terms*

4. Design Constraints
   Requirements derived from existing standards or hardware limitations

5. Software System Attributes
   1. Reliability: establish the required reliability of the software system
   2. Availability: guarantee a defined availability level for the entire system
   3. Security: protect the software from accidental or malicious access, use, modification, destruction, or disclosure
   4. Maintainability: ease of maintenance of the software itself
   5. Portability: ease of porting the software to other host machines and/or operating systems

Table of Contents

- Eliciting Requirements - Examples
- **Criteria for good requirements**
Typical mistakes

- **Noise**
  - text that carries no relevant information to any feature of the problem.

- **Silence**
  - a feature that is not covered by any text.

- **Over-specification**
  - text that describes a detailed design decision, rather than the problem.

- **Contradiction**
  - text that defines a single feature in a number of incompatible ways.

- **Ambiguity**
  - text that can be interpreted in at least two different ways.

- **Forward reference**
  - text that refers to a terms or features yet to be defined.

- **Wishful thinking**
  - text that defines a feature that cannot possibly be validated.

- **Requirements on users**
  - Cannot require users to do certain things, can only assume that they will

- **Jigsaw puzzles**
  - distributing key information across a document and then cross-referencing

- **Duckspeak requirements**
  - Requirements that are only there to conform to standards

- **Unnecessary invention of terminology**
  - E.g. ‘user input presentation function’

- **Inconsistent terminology**
  - Inventing and then changing terminology

- **Putting the onus on the developers**
  - i.e. making the reader work hard to decipher the intent

- **Writing for the hostile reader**
  - There are fewer of these than friendly readers
Do not write like this

● Ambiguity – or
  ○ *The same subsystem shall also be able to generate visible or audible caution or warning signal for the attention of security or business analyst*

● Multiple requirements – *and, or, with, also*
  ○ *The warning lamp shall light up when system intrusions is detected and the current workspace or input shall be saved*
Do not write like this

- Let-out clauses
  * if, when, except, unless, although, always

  - The fire alarm shall always be sounded when the smoke is detected, unless the alarm is being tested when the antivirus is deployed

- Long rambling sentences

  - Provided that the designated input signals from the specified devices are received in the correct order where the systems is able to differentiate the designators, the security solution should comply with the required framework of Section 3.1.5 to indicate the desired security states
Do not write like this

- **System design:**
  - **no names of components, materials, software objects/procedures, database fields**

  - *The antenna shall be capable of receiving FM signals, using a copper core with nylon armoring and a waterproof hardened rubber shield*

- **Mix of requirements and design:**
  - **no references to system, design, testing, or installation**

  - *The user shall be able to view the current selected channel number which shall be displayed in 14pt Swiss type on an LCD panel tested to standard 657-89 and mounted with shockproof rubber washers*
Do not write like this

- Speculation
  usually, generally, often normally, typically
  
  ○ *Users normally require early indication of intrusion into the system*

- Vague, undefinable terms
  user-friendly, versatile, approximately, as possible, efficient, improved, high-performance, modern
  
  ○ *Security-related messages should be versatile and user-friendly*
  ○ *The OK status indicator lamp shall be illuminated as soon as possible after system security self-check is completed*
Do not write like this

- Wishful thinking

  100% reliable/ safe/ secure. Handle all unexpected failures. Please all users. Run on all platforms. Never fail. Upgrade to all future situations.

- The gearbox shall be 100% secure in normal operation.
- The network shall handle all unexpected errors without crashing.
Good requirements

- Use simple direct sentences
  - Security analyst should be able to view system status

- Use a limited vocabulary
  - Security analyst should be able to change the infected component in less than 12 hours
  - Security analyst should be able to reconfigure the infected component in less than 12 hours
Good requirements

- Identify the type of user who wants each requirements
  - The navigator shall be able to...

- Focus on stating result
  - ... view storm clouds by radar ...

- Define verifiable criteria
  - ... at least 100 km ahead.

  - Acceptance criterion: Aircraft flying at 800km/h at 10,000 meters towards a known storm cloud indicated by meteorology satellite report; storm cloud is detected at a range of at least 100 km.
Criteria for Writing Good Requirements

- What, not how (external observability)
  - Avoid premature design or implementation decisions
- Understandability, clarity (not ambiguous)
- Cohesiveness (one thing per requirement)
- Testability
  - Somehow possible to test or validate whether the requirement has been met, clear acceptance criteria
  - Often requires quantification, this is more difficult for security than e.g. for performance
    - “The response time of function F should be max 2 seconds”
    - “The security of function F should be at least 99.9 %”
Documenting requirements artefacts

Diagram:

- Identifier
- Description
  - has
- Requirements artefact
  - has
  - Criticality
  - Priority
  - Risk
  - 0..* has
  - Goal
  - 1..* 1..*
  - example of satisfaction
  - Scenario
    - contributes to realisation of
    - derived from
  - Solution-oriented requirement
    - derived from
    - 1..* 0..*
  - is realised by
Requirement Shell
Volere template, 2010

Requirement # : Unique id           Requirement Type :       Event/use case # :

Description : *A one sentence statement of the intention of the requirement*

Rationale : *A justification of the requirement*

Originator : *Who raised this requirement?*

Fit Criterion : *A measurement of the requirement such that it is possible to test if the solution matches the original requirement*

Customer Satisfaction : Customer Disatisfaction :

Priority : *The relative urgency of this requirement*

Supporting Materials : *Pointer to documents that illustrate and explain this requirement*

History : *Creation, changes, deletions, etc.*

The type from the template

List of events/use cases that need this requirement

Degree of stakeholder happiness if this requirement is successfully implemented. Scale from 1 = uninterested to 5 = extremely pleased.

Measure of stakeholder unhappiness if this requirement is not part of the final product. Scale from 1 = hardly matters to 5 = extremely displeased.
Take Home!

• Requirements specification

• Criteria for good requirements
Exercise
Do these security requirements correspond to the “criteria of good requirements”? If not, please refine them so that they would correspond to the “criteria of good requirements”.

SecReq.1: Automation of the receipt generation or checking fuel capacity and sold amount of fuel as often as possible.

SecReq.2: Cash register computer must compare data about company car / employee ID to ERP and give warning if there is a mismatch.

SecReq.3: IT department must follow security advisories and patch the vulnerable systems, given that solutions are available.

SecReq.4: Implementing at least two workplaces for serving customers in filling station.

SecReq.5: Appropriate trainings shall be provided to all employees of PowerAB with regards to information security awareness.

SecReq.6: Database backup server accounts should be sufficiently protected against unauthorized access.

SecReq.7: IT managers should plan the placement of card readers near 24/7 cameras, which are monitored from a security officer.

SecReq.8: Have a backup server for redundancy and push uptime to near 100%.

SecReq.9: Using better security applications.

SecReq.10: The firewall should continuously monitor the communication channel and block suspicious software while transmitting message from account department to personnel department.
SecReq.1: Automation of the receipt generation or checking fuel capacity and sold amount of fuel as often as possible.

SecReq.2: Cash register computer must compare data about company car / employee ID to ERP and give warning if there is a mismatch.

SecReq.3: IT department must follow security advisories and patch the vulnerable systems, given that solutions are available.
SecReq.1: **Automation** of the receipt generation or checking fuel capacity and sold amount of fuel as often as possible.

SecReq.2: Cash register computer must compare data about company car / employee ID to ERP and give warning if there is a mismatch.

SecReq.3: IT department must follow security advisories and patch the vulnerable systems, given that solutions are available.

SecReq.1.1: Fuel system should check fuel capacity

SecReq.1.2: Receipt system should generate receipts

SecReq.2: Cash register computer should notify attendant about mismatch between car number and employee ID

SecReq.3: IT department should patch PowerAB software systems
SecReq.4: Implementing at least two workplaces for serving customers in filling station.

SecReq.5: Appropriate trainings shall be provided to all employees of PowerAB with regards to information security awareness.

SecReq.6: Database backup server accounts should be sufficiently protected against unauthorized access.
SecReq.4: Implementing at least two workplaces for serving customers in filling station.

SecReq.5: Appropriate trainings shall be provided to all employees of PowerAB with regards to information security awareness.

SecReq.6: Database backup server accounts should be sufficiently protected against unauthorized access.
SecReq.7: IT managers should plan the placement of card readers near 24/7 cameras, which are monitored from a security officer.

SecReq.8: Have a backup server for redundancy and push uptime to near 100%.
SecReq.7: IT managers should plan the placement of card readers near 24/7 cameras, which are monitored from a security officer.

SecReq.8: Have a backup server for redundancy and push uptime to near 100%.

SecReq.7: Security officer should monitor the card readers.

SecReq.8: The backup server should ensure the uptime of 99.1%.
SecReq.9: Using better security applications.

SecReq.10: The firewall should continuously monitor the communication channel and block suspicious software while transmitting message from account department to personnel department.
SecReq.9: Using better security applications.

SecReq.9: “I do not know how to fix it !!!”

SecReq.10: The firewall should continuously monitor the communication channel and block suspicious software while transmitting message from account department to personnel department.

SecReq.10.1: The firewall should monitor the communication channel.

SecReq.10.2: The firewall should block the untrusted software.

Explain “untrusted”!
Exercise

- The application shall verify the identity of all of its users before allowing them to use all its capabilities

- The system shall let users to log in with passwords of at least 8 characters, containing both small and capital letters, numbers and special signs

- The system shall use Norton antivirus protection

- The application shall disinfect any file found to contain a harmful program if disinfection is possible

- The system shall encrypt all confidential data using the RSA algorithm
Exercise

- The application shall verify the identity of all of its users before allowing them to use all its capabilities.

- The system shall let users to log in with passwords of at least 8 characters, containing both small and capital letters, numbers and special signs.

- The system shall use Norton antivirus protection.

- The application shall disinfect any file found to contain a harmful program if disinfection is possible.

- The system shall encrypt all confidential data using the RSA algorithm.