



WRITING GOOD REQUIREMENTS

DO

- + Start with entity: "The Navigation_Module shall..."
- + Use "shall" for obligations
- + Be specific: "within 100 +/- 10 ms"
- + One thought per requirement
- + Use active voice: "shall process"
- + End with a period
- + Add tolerances: "12 +/- 1 m"

DON'T

- Vague terms: fast, reliable, user-friendly
- Escape clauses: where possible, as appropriate
- Absolutes: always, never, 100%
- Pronouns: it, they, this, that
- Implementation: MySQL, Python, AWS
- Purpose phrases: in order to, so that
- Passive voice: shall be processed

REQUIREMENT PATTERNS

Basic:	The [entity] shall [action] [object].
Conditional:	When [condition], the [entity] shall [action].
Performance:	The [entity] shall [action] within [X +/- Y].
Interface:	The [entity] shall [send/receive] [data] via [interface].

THE 42 INCOSE RULES

Accuracy R1-R9

- R1 Structured statements
- R2 Active voice
- R3 Appropriate subject
- R4 Defined terms
- R5 Definite articles (the)
- R6 Common units
- R7 No vague terms
- R8 No escape clauses
- R9 No open-ended clauses

Concision R10-R11

- R10 No superfluous infinitives
- R11 Separate clauses

Non-Ambiguity R12-R17

- R12 Correct grammar
- R13 Correct spelling
- R14 Correct punctuation
- R15 Logical [X AND Y]
- R16 Avoid 'not'
- R17 Avoid '/'

Singularity R18-R23

- R18 Single thought
- R19 No combinators
- R20 No purpose phrases
- R21 No parenthetical text
- R22 Enumerate sets
- R23 Reference diagrams

Other Rules

- R24-25 Completeness (no pronouns)
- R26 Realism (no absolutes)
- R27-28 Explicit conditions
- R29-30 Unique expression
- R31 Solution-free
- R32 Use 'each' not 'all'
- R33 Define tolerances
- R34-35 Measurable criteria
- R36-40 Uniformity
- R41-42 Modularity

INTERFACE GUIDE

Navigation Tabs

Dashboard	Project overview & stats
Author	Write & edit requirements
Register	Table view with filters
Trace	Parent-child relationships
V&V	Verification planning
Rules	42 rules reference
Glossary	Project dictionary

QUALITY GRADES

A+ / A	90%+	Excellent - ready for review
B+ / B	80-89%	Good - minor improvements
C+ / C	70-79%	Acceptable - review rules
D	60-69%	Needs work - use AI
F	<60%	Poor - significant rewrite

KEY ACTIONS

+ New	Create new requirement
Save Project	Download JSON backup
Load	Restore saved project
Export > CSV	Spreadsheet format
Export > ReqIF	DOORS compatible
AI Improve	Auto-fix violations
Bulk Analysis	Fix all at once

RECOMMENDED WORKFLOW

1. Start with Stakeholder Needs
2. Derive System Requirements (link parents)
3. Decompose to Subsystem/Component
4. Run AI Improve on low scores
5. Build glossary for key terms
6. Check traceability for orphans
7. Export when all >= 80%