The designed checklist

|  |  |  |  |
| --- | --- | --- | --- |
| Item no | Where to look? | How to detect? |  |
| 1 | ***Correctness and completeness*** | Whether all functional capabilities are documented | □ |
| 2 |  | Whether reuse of existing software or use of COTS software is fully described | □ |
| 3 |  | Whether adaptation requirements (e.g., geographic parameters, platform variations) are identified | □ |
| 4 |  | Whether applicable timing, resource usage (e.g., CPU, memory, bandwidth), and associated system load requirements are identified | □ |
| 5 |  | Whether the effect of the operating system, executive, or COTS has been factored into the performance requirements and resource budgets | □ |
| 6 |  | Whether the applicable safety & security requirements are identified | □ |
| 7 |  | Whether applicable design constraint requirements (e.g., object-oriented design, language, support environment) are identified | □ |
| 8 |  | whether the applicable software quality attributes (e.g., reliability, maintainability, testability) are identified. | □ |
| 9 |  | Whether the applicable human performance/human engineering requirements are identified | □ |
| 10 |  | Whether applicable acceptance criteria (e.g., test, inspection, demonstration) are identified | □ |
| 11 |  | Whether requirements are traceable to requirements allocated to software | □ |
| 12 |  | Whether all functional data flows are specified, including sources and destinations | □ |
| 13 |  | Whether inputs and outputs of each requirement are necessary and sufficient for the specified processing | □ |
| 14 |  | Whether accuracy and precision requirements are defined | □ |
| 15 |  | Whether all software functions are considered (e.g., startup, restart, modes of operation, shutdown, normal terminations, abnormal conditions, performance monitoring and tuning). | □ |
| 16 |  | Whether operator interactions are considered | □ |
| 17 |  | Whether all functional processing requirements are specified for recognized error conditions (e.g., hardware faults, I/O errors, computational errors, processing overload, buffer overflow, events failing to occur, out-of-sequence events) | □ |
| 18 |  | Whether all test requirements are defined (e.g., test levels; provisions to inject test data, to adjust parameters, to control or trace the execution of test runs, to extract and reduce test results) | □ |
| 1 | ***Consistency*** | Whether all software requirements are derived from the system specification | □ |
| 2 |  | Whether each object is referred to by a unique name | □ |
| 3 |  | whether each object is defined by a set of non-conflicting characteristics | □ |
| 4 |  | Whether all requirements are free of logical/timing conflicts | □ |
| 5 |  | Whether requirements do not conflict with each other | □ |
| 6 |  | Whether all data/messages/requirements are specified only once | □ |
| 7 |  | Whether data flows are consistent with the specified inputs and outputs of the relevant requirements | □ |
| 8 |  | Whether data flow notations are used consistently | □ |
| 9 |  | Whether message data attributes are consistent with the inputs and outputs of relevant requirements | □ |
| 10 |  | Whether loads used to allocate resource budgets are consistently specified for all functions | □ |
| 11 |  | Whether function names used in diagrams are consistent with the requirements text | □ |
| 12 |  | Whether requirements are consistent with the operational context | □ |
| 1 | ***Feasibility*** | Whether data expected from external sources exist at those sources | □ |
| 2 |  | Whether data sent to external destinations are expected at those destinations | □ |
| 3 |  | Whether requirements are achievable with available technology and necessary implementation tools are available | □ |
| 4 |  | Whether the scope of requirements is realistic, considering software estimates, schedules, and support facility plans | □ |
| 5 |  | Whether the performance requirements are realistic based upon available facts or modeling information, (e.g., response times, accuracies, processing capacities) | □ |
| 6 |  | Whether resource budgets are realistic (e.g., CPU time, I/O utilization, memory, worst-case loads, data storage) | □ |
| 1 | ***Testability*** | Whether all requirements are specified against the software (i.e., not against the hardware or the operator). | □ |
| 2 |  | Whether all requirements can be verified by some (implicit, explicit, analytical, empirical) means. | □ |
| 3 |  | Whether test results can be evaluated against predetermined acceptance criteria. | □ |
| 1 | ***Meets Standards*** | Whether major software functions are described in relation to system operation | □ |
| 2 |  | Whether requirements are clearly stated and unambiguous | □ |
| 3 |  | Whether terminology is understandable and consistent | □ |
| 4 |  | Whether all notation and naming conventions are defined | □ |
| 5 |  | Whether glossary is adequate | □ |
| 6 |  | Whether requirements are clearly numbered or otherwise marked | □ |
| 7 |  | Whether requirements terminology (i.e., “shall”, “will”, “may”, etc.) is used correctly | □ |