

# Design Review Checklist

Project Code

Version of the work product

Reviewer(s)

Review date

Work product' size

Effort spent on review (man-hour)

Question	Yes	No	N/A	Note
<b>DOCUMENT CONTROL</b>				
Verify whether document control procedures have been followed by checking for the following:				
Does the title page contain the document name, version number, release date and issued date.?				
Does the header and footer correctly specify the name, and version of the document?				
Does the page – numbering scheme indicate the total number of pages in the document?				
Is the history traceable?				
Does it include list of reference documents?				
Are all documents checked Spelling and Grammar in MS Word or similar tools?				
<b>ARCHITECTURE DESIGN</b>				
Is the system architecture, including the data flows, control flows, high-level elements and interface, clearly represented?				
Does the architecture cleanly decompose the top-level elements of the system?				
To the extent possible, is the architecture independent of the technology that will be used to implement it?				
Does the architecture take into account technology or other constraints that cannot be avoided? i.e., can the architecture be implemented for the target environment?				
Are external interfaces, including user interfaces, defined and justified?				
Is the architecture appropriately layered?				
Is an error-handling strategy described and justified?				
Is an I/O strategy described and justified?				
Is the architecture considered in selection process? i.e. has a brief explanation of advantages and disadvantages of the selected architecture in comparison with others				
<b>HIGH LEVEL DESIGN</b>				
Are all major components described and justified?				
Is there a sufficient number of views and/or models for all major aspects of design to be fully understood?				
Has the dataflow among all components been described?				
Are all major algorithms described and justified?				
Are component boundaries well defined, including functionality and interfaces?				
Does the design limit component connections and interaction?				
Have all shared data and resource between components been described?				

Question	Yes	No	N/A	Note
Is the problem domain model sufficient to capture all states and behavior?				
Are all significant system meta-states and events captured?				
Are all major data structures described and justified?				
If appropriate, are major data structures hidden by an abstraction layers?				
Is the conceptual view for all composite data elements and objects documented?				
Are inputs to routines necessary and sufficient to perform the required operation?				
Do routines clearly state how the output is derived from input or shared data?				
Are all the outputs produced by a routine used?				
Is a strategy described and justified for:				
Handling special states? (e.g. abnormal termination, error recovery, losing power)				
Handling failure of the system? (e.g. process termination, system recovery)				
Memory management?				
Does it included memory use estimates?				
Share resource management? Are the modules that use the shared resources indicated?				
Handling strings? Is localization or internationalization an issue?				
<b>DETAILED DESIGN</b>				
Does each unit have a unique identifier and follow style and naming convention?				
Are the proposed programming languages and technologies appropriate?				
Does the design for each unit contain appropriate views? (e.g., static structure, data definition, data flow, control flow, states, etc.)				
Has sufficient detail been included for someone else to develop the code?				
Has the overall function and intent of each unit been described?				
Are calls, events, and messages between units documented?				
Does the design reflect the actual operating environment? Hardware? Software?				
Does the design support the appropriate level modularity?				
Does the design support the appropriate level of design and code reuse?				
Have all the constraints, such as processing time, run time, memory usage, I/O, database access and response time, for this unit been taken into consideration?				
Are error, exception and other abnormal processing and states described in detail?				
Have data elements been described to a sufficiently low level of detail?				
Have valid value ranges and other data constraints been specified?				
Has the management and use of shared and stored data been clearly described?				
Are the internal and external interfaces clearly defined?				
Are the inputs and outputs for all the interfaces are sufficient and necessary?				
Does it describe in detail all the required modules as follows:				
Does it describe actors, use-cases of the system?				

Question	Yes	No	N/A	Note
Does it describe flow diagram of all public functions?				
Does it describe public member variables used in system?				
Does it describe format of external data used in the system?				
Does it describe GUI of the system as follows:				
Does it describe/layout position of all items in GUI?				
Does it describe processes or actions for GUI?				
Does it describe relationship between items in GUI and modules or public functions?				
Are the designed elements considered in selection process? i.e. has a brief explanation of advantages and disadvantages of the selected solution for designed elements in comparison with others				
<Add more rows if needed>				

**\* Comments**

**\* Suggestion**

- [    ] - Pass
- [    ] - Review again
- [    ] - Other