Date of Assessment: Project: **Volunteers’ Management System**

Assessor(s):

Process Assessed:

|  |  | Y, N, NA | F, O | Comments |
| --- | --- | --- | --- | --- |
| General Confederations | | | | |
| 1 | The complexity of the system matches the functionality it provides. |  |  |  |
| 2 | The system has a single consistent, coherent architecture. |  |  |  |
| 3 | The number and types of component is reasonable. |  |  |  |
| 4 | The system has a consistent system-wide security facility. All the security components work together to safeguard the system. |  |  |  |
| 5 | The system will meet its availability targets. |  |  |  |
| 6 | The architecture will permit the system to be recovered in the event of a failure within the required amount of time. |  |  |  |
| 7 | The architecture provides defines clear interfaces to enable partitioning for parallel team development. |  |  |  |
| 8 | The designer of a model element can understand enough from the architecture to successfully design and develop the model element. |  |  |  |
| 9 | The proposed solution can be easily understood by someone generally knowledgeable in the problem domain. |  |  |  |
| 10 | The Design Guidelines have been followed. |  |  |  |
| **Architectural Analysis Considerations** | | | | |
| 11 | Subsystem and package partitioning and layering is logically consistent. |  |  |  |
| 12 | All analysis mechanisms have been identified and described. |  |  |  |
| 13 | The services (interfaces) of subsystems in upper-level layers have been defined. |  |  |  |
| **General Model Considerations** | | | | |
| 14 | For the business model, requirements model or the design model during the elaboration phase, there is not an over-emphasis on implementation issues. |  |  |  |
| 15 | For the design model in the construction phase, there is a good balance of functionality across the model elements, using composition of relatively simple elements to build a more complex design. |  |  |  |
| 16 | Concepts are modeled in the simplest way possible. |  |  |  |
| 17 | The model is easily evolved; expected changes can be easily accommodated. |  |  |  |
| 18 | The key assumptions behind the model are documented and visible to reviewers of the model. If the assumptions are applicable to a given iteration, then the model should be able to be evolved within those assumptions, but not necessarily outside of those assumptions. |  |  |  |
| Diagrams | | | | |
| 19 | The purpose of the diagram is clearly stated and easily understood. |  |  |  |
| 20 | The graphical layout is clean and clearly conveys the intended information. |  |  |  |
| 21 | Labeling of model elements contributes to understanding. |  |  |  |
| **Error recovery** | | | | |
| 22 | For each error or exception, a policy defines how the system is restored to a "normal" state. |  |  |  |
| 23 | For each possible type of input error from the user or wrong data from external systems, a policy defines how the system is restored to a "normal" state. |  |  |  |
| **Administration** | | | | |
| 24 | Disk space can be reorganized or recovered while the system is running. |  |  |  |
| 25 | The responsibilities and procedures for system configuration have been identified and documented. |  |  |  |
| 26 | Access to the operating system or administration functions is restricted. |  |  |  |
| 27 | Licensing requirements are satisfied. |  |  |  |
| 28 | It is not possible for a malicious user to enter the system, or destroy critical data, or consume all resources. |  |  |  |
| **Performance** | | | | |
| 29 | Performance requirements are reasonable and reflect real constraints in the problem domain; their specification is not arbitrary. |  |  |  |
| **Reliability** | | | | |
| 30 | The architecture provides for recovery in the event of disaster or system failure |  |  |  |
| **Security** | | | | |
| 31 | Security requirements have been met. |  |  |  |
| **The Logical View** | | | | |
| 32 | Accurately and completely presents an overview of the architecturally significant elements of the design. |  |  |  |
| 33 | Presents the complete set of architectural mechanisms used in the design along with the rationale used in their selection. |  |  |  |
| 34 | Presents the layering of the design, along with the rationale used to partition the layers. |  |  |  |
| **The Process View** | | | | |
| 35 | The system tolerant of errors and exceptions, such that when an error or exception occurs, the system can revert to a consistent state. |  |  |  |
| 36 | Processes are sufficiently independent of one another that they can be distributed across processors or nodes when required. |  |  |  |
| **The Deployment View** | | | | |
| 37 | The throughput requirements have been satisfied by the distribution of processing across nodes, and potential performance bottlenecks have been addressed. |  |  |  |
| 38 | Requirements for reliable transport of messages, such that they exist, have been satisfied. |  |  |  |
| 39 | Requirements for secure transport of messages, such that they exist, have been satisfied. |  |  |  |
| 40 | System availability requirements, to the extent that they exist, have been satisfied. |  |  |  |
| 41 | All potential failure modes have been documented. |  |  |  |
| **Headers/Footers and Documentation format** | | | | |
| 42 | Revision History |  |  |  |
| 43 | Table of Contents |  |  |  |
| 44 | Headers/Footers |  |  |  |
| 45 | Format of the document |  |  |  |

# COMMENTS PAGE

|  |  |
| --- | --- |
| **#** | **Comments from assessment** |
| **1.** |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |