



Your Company Name

Software Architecture Plan

Date

www.SDLCforms.com



Revision History

Date	Version	Author	Change

www.SDLCforms.com

COPYRIGHT NOTICE

Confidential – ©2015 Documentation Consultants

All rights reserved. These materials are for internal use only. No part of these materials may be reproduced, published in any form or by any means, electronic or mechanical, including photocopy or any information storage or retrieval system, nor may the materials be disclosed to third parties without the written authorization of (Your Company Name).



Table of Contents

1	Purpose	4
1.1	Scope	4
1.2	Definitions, Acronyms, and Abbreviations	4
1.3	References	4
1.4	Overview.....	4
2	Architectural Representation	5
3	Architectural Goals and Constraints	5
4	Use-Case View	5
5	Logical View	5
5.1	Overview.....	5
5.2	Architecturally Significant Design Packages	6
6	Process View	6
7	Deployment View	6
8	Implementation View	6
8.1	Overview.....	6
8.2	Layers.....	7
9	Data View (optional)	7
10	Size and Performance	7
11	Quality	7



This document provides a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system.

Note: Text displayed in blue italics is included to provide guidance to the author and should be deleted before publishing the document. In any table, select and delete any blue line text; then click Home→Styles and select “Table Text” to restore the cells to the default value.

1 Purpose

The purpose defines the role or purpose of the Software Architecture Document in the overall project documentation, and briefly describes the structure of the document. The specific audience for the document is identified, with an indication of how they are expected to use the document.

1.1 Scope

This is a brief description of what the Software Architecture Document applies to; what is affected or influenced by this document.

1.2 Definitions, Acronyms, and Abbreviations

Provides the definitions of all terms, acronyms, and abbreviations required to properly interpret the Software Architecture Document. This information may be provided by reference to the project’s Glossary.

1.3 References

Provides a complete list of all documents referenced elsewhere in the Software Architecture Document. Identify each document by title, report number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.

1.4 Overview

Describes what the rest of the Software Architecture Document contains and explains how the Software Architecture Document is organized.



2 Architectural Representation

This section describes what software architecture is for the current system, and how it is represented. Of the Use-Case, Logical, Process, Deployment, and Implementation Views, it enumerates the views that are necessary, and for each view, explains what types of model elements it contains.

3 Architectural Goals and Constraints

This section describes the software requirements and objectives that have some significant impact on the architecture; for example, safety, security, privacy, use of an off-the-shelf product, portability, distribution, and reuse. It also captures the special constraints that may apply: design and implementation strategy, development tools, team structure, schedule, legacy code, and so on.

4 Use-Case View

This section lists use cases or scenarios from the use-case model if they represent some significant, central functionality of the final system, or if they have a large architectural coverage—they exercise many architectural elements or if they stress or illustrate a specific, delicate point of the architecture.

5 Logical View

This section describes the architecturally significant parts of the design model, such as its decomposition into subsystems and packages; and for each significant package, its decomposition into classes and class utilities. You should introduce architecturally significant classes and describe their responsibilities, as well as a few very important relationships, operations, and attributes.

5.1 Overview

Describes the overall decomposition of the design model in terms of its package hierarchy and layers.



5.2 Architecturally Significant Design Packages

For each significant package, include a subsection with its name, its brief description, and a diagram with all significant classes and packages contained within the package.

For each significant class in the package, include its name, brief description, and, optionally, a description of some of its major responsibilities, operations, and attributes.

6 Process View

This section describes the system's decomposition into lightweight processes (single threads of control) and heavyweight processes (groupings of lightweight processes). Organize the section by groups of processes that communicate or interact. Describe the main modes of communication between processes, such as message passing, interrupts, and rendezvous.

7 Deployment View

This section describes one or more physical network (hardware) configurations on which the software is deployed and run. It is a view of the Deployment Model. At a minimum for each configuration it should indicate the physical nodes (computers, CPUs) that execute the software and their interconnections (bus, LAN, point-to-point, and so on.) Also include a mapping of the processes of the Process View onto the physical nodes.

8 Implementation View

This section describes the overall structure of the implementation model, the decomposition of the software into layers and subsystems in the implementation model, and any architecturally significant components.

8.1 Overview

Names and defines the various layers and their contents, the rules that govern the inclusion to a given layer, and the boundaries between layers. Include a component diagram that shows the relations between layers.



8.2 Layers

For each layer, include a subsection with its name, an enumeration of the subsystems located in the layer, and a component diagram.

9 Data View (optional)

This section contains a description of the persistent data storage perspective of the system. This section is optional if there is little or no persistent data, or the translation between the Design Model and the Data Model is trivial.

10 Size and Performance

This section contains a description of the major dimensioning characteristics of the software that impact the architecture, as well as the target performance constraints.

11 Quality

This section describes how the software architecture contributes to all capabilities (other than functionality) of the system: extensibility, reliability, portability, and so on. If these characteristics have special significance, such as safety, security or privacy implications, they must be clearly delineated.