

System Design Document

[Jump to navigation](#) [Jump to search](#)

Original template borrowed from:

<https://www.cs.fsu.edu/~lacher/courses/COP3331/sdd.html>



Contents

- 1 [Purpose](#)
- 2 [Audience](#)
- 3 [Template](#)
 - 3.1 [Section/Topic](#)
 - 3.1.1 [Introduction](#)
 - 3.1.1.1 [Purpose of the System](#)
 - 3.1.1.2 [Design Goals](#)
 - 3.1.1.3 [Definitions, Acronyms, Abbreviations](#)
 - 3.1.1.4 [References](#)
 - 3.1.2 [Current Software Architecture](#)
 - 3.1.3 [Proposed Software Architecture](#)
 - 3.1.3.1 [Overview](#)
 - 3.1.3.2 [Subsystem Decomposition](#)
 - 3.1.3.3 [Software / Hardware Mapping](#)
 - 3.1.3.4 [Persistent Data Management](#)
 - 3.1.3.5 [Access Control and Security](#)
 - 3.1.3.6 [Global Software Control](#)
 - 3.1.3.7 [Boundary Conditions](#)
 - 3.1.4 [Subsystem Architectures](#)
 - 3.1.5 [Glossary](#)

Purpose

The results of the system design process are recorded in the System Design Document (SDD). This document completely describes the system at the architecture level, including subsystems and their services, hardware mapping, data management, access control, global software control structure, and boundary conditions. The SDD should define a virtual machine that implements all requirements in the RAD, and it should

provide a foundational guide for further implementation details all the way to an executable solution.

Note that the SDD is a "live" document that should be incrementally expanded and refined during review cycles.

Audience

The audience for the SDD includes the software architect and lead members (liaisons) from each subsystem development team.

Template

Section/Topic

Introduction

The first section of the SDD is an Introduction. Its purpose is to provide a brief overview of the function of the system and the reasons for its development, its scope, and references to the development context (e.g., reference to the problem statement written by the client, references to existing systems, feasibility studies, and the RAD).

Purpose of the System

Problem statement, business context, or purpose of the system. This should also include any relevant context that originated in its development.

Design Goals

Definitions, Acronyms, Abbreviations

References

Current Software Architecture

This section describes the current state of affairs, from a system insiders perspective. Particular attention and detail should be given to any portions of the current system that may be incorporated into the new one ("legacy" code).

Proposed Software Architecture

This section describes the top level software architecture for the system under development.

Overview

The overview presents an architectural overview of the system, the "10,000 foot" perspective. AKA: the executive summary.

Subsystem Decomposition

Name the top-level subsystems comprising the system, along with a

description of the responsibilities of each subsystem.

Software / Hardware Mapping

Describe how subsystems are assigned to hardware and off-the-shelf components. Also list and explain issues introduced by multiple nodes and software re-use.

Persistent Data Management

Describe the persistent data stored by the system and the data management infrastructure required for it. This section should include the description of data schemes, the selection of a database, and a description of the encapsulation of the database.

Access Control and Security

Describe the user model of the system in terms of an access matrix. This section also describes security issues, such as the selection of an authentication mechanism, the use of encryption, and management of keys.

Global Software Control

Describe how the global software control is implemented. In particular, this section should describe how requests are initiated and how subsystems synchronize. This section should list and address synchronization and concurrency issues.

Boundary Conditions

List and describe the boundary conditions: startup, shutdown, and error behaviour of the system.

Subsystem Architectures

For each subsystem identified in section 3.2, give the (1) subsystem architecture: its design, either as a known pattern or a new design; (2) the subsystem interface or API; and (3) a brief description of the implementation plan for the subsystem.

Glossary

A glossary of all entities defined and used in the software, to ensure consistency in the design and implementation. These definitions should be precise and expressed in the language of software developers. Where terms originate from the RAD Glossary, make appropriate correlating references. Aka the Data Dictionary

Retrieved from

"https://wiki.silogroup.org/index.php?title=System_Design_Document&oldid=186"

This page was last edited on 23 May 2020, at 23:42.

Content is available under Attribution-NonCommercial-NoDerivatives 4.0 International unless otherwise noted.