

What is System Design?

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization.

What is SDD?

The System Design Document (SDD) is a documentation that describes how the functional and nonfunctional requirements recorded in the Requirements Document; transform into more technical system design specifications from which the system is built. The SDD documents the high-level system design and the low-level detailed design specifications.

The SDD describes design goals and considerations, provides a high-level overview of the system architecture, and describes the data design associated with the system, as well as the human-machine interface and operational scenarios.

The high-level system design is further decomposed into low-level detailed design specifications for each system component, including hardware, internal communications, software, system integrity controls, and external interfaces. The high-level system design serves as primary input to the Preliminary Design Review (PDR). The low-level detailed design serves as input to the Detailed Design Review (DDR).

SOFTWARE DESIGN SPECIFICATION TEMPLATE

Note: - You should begin to develop the software design specification after completion of the software requirements specification. The software requirements feed the design task.

The software design specification focuses on how the system will be constructed. It includes four models: data design (section 2.0), architectural design (section 3.0), interface design (section 4.0) and component-level design (section 3.0).

(Section 1.0 is repeated from the Software Requirements Specification. This section provides background information about the System.)

Note:-Page one of your submitted SDS documentation should be your cover page and page two the document Table of Contents (TOC).

Section One

1.0 Introduction

This section provides an overview of the entire requirement document. This document describes all data, functional and behavioral requirements for the proposed system.

An Example:

The purpose of this document is to describe in sufficient detail how the proposed system is to be constructed. The System Design Document translates the Requirement Specifications into a document from which the developers can create the actual system. It identifies the top-level system architecture, and identifies hardware, software, communication, and interface components.

1.1 Goals and objectives

Overall goals and proposed system objectives are described.

1.2 Statement of scope

Give a description of your proposed system major inputs, processing functionalities and outputs.

-Rank the major processing functionality from the developer's point of view. Use a simple ranking system such as: essential, desirable and future requirements. This should represent what you think you can accomplish in the time frame of a semester. The essential requirements, you are sure you can complete. The desirable requirements you hope to complete, but are not sure about. The future requirements, you have strong doubts about. Strive to balance the desires of your client with the reality of the time it takes to develop your proposed system.

1.3. Major constraints

Briefly explain any business or product line constraints that will impact the manner in which your proposed system is to be specified, designed, implemented or tested..

Section Two

2.0 Conceptual Data design

2.1. Project Conceptual Data Model

A project conceptual data model (CDM) is a high level representation of the data entities and their relationships. It normally includes the data elements that comprise each entity. It is a first step to developing the more detailed logical data model (LDM) which will be provided during the Logical Data Design.

Instructions:-Provide your Project system conceptual data model e.g. using an ERD diagram, database normalization etc.

Section Three

3.0 Architectural and component-level design

A description of the system architecture is presented in this section.

What is an Architectural design? Architectural design represents the structure of data and program components that are required to build a computer-based system. It considers the architectural style that the system will take, the structure and properties of the components that constitutes the system, and the interrelationship that occurs among all architectural components of a system.

What is Component level design? Component level design is the definition and design of components and modules after the architectural design phase. Component-level design defines

the data structures, algorithms, interface characteristics, and communication mechanisms allocated to each component for the system development.

3.1 System Structure

A detailed description of the proposed system structure chosen is presented in this section.

3.1.1 Architecture diagram

Document a pictorial representation of your proposed architecture e.g. using a UML diagram.

3.2 Description for system Components

A detailed description of each software component and modules contained within the architecture is presented in this section.

-Note: - section 3.2 is repeated for each system component.

3.2.1 Processing narrative for **each** software component (e.g. *component n*)

A processing narrative for each component is presented. It should describe the responsibilities of the component.

3.2.2 Component n interface description.

A detailed description of the input and output interfaces for the component is presented.

3.2.3 Component n processing detail

A detailed algorithmic description for each component is presented.

3.3.1 Interaction Diagrams

A sequence diagram, for each use case the component realizes, is presented.

3.4 Database Modelling:

Instructions: Describe the design of your database management system (DBMS) files and non-DBMS files associated with the system. Provide a comprehensive data dictionary showing data element name, type, length, source, validation rules, maintenance (create, read, update, delete (CRUD) capability), data stores, outputs, aliases, and description.

Section Four

4.0 User interface design

A description of the user interface design of the software is presented.

4.1 Description of the user interface. A detailed description of user interface including screen images is presented.

4.1.1 Screen images (Note:-no system screen shots will be accepted. You are yet to develop your proposed system)

Instructions: - Give a Representation of the interface forms from the user's point of view such as data entry screens. In addition, Describe the system output design relative to the user. System outputs include reports, data display screens, query results, etc.

Identify the following, if appropriate: -

- ✓ Access restrictions or security considerations
- ✓ Description of the purpose of the output
- ✓ Report requirements, including frequency for periodic reports
- ✓ Screen contents (provide a graphic representation of each layout. Define all data elements associated with the layout)

4.1.2 Objects and actions

Give a brief explanation of your system screen objects and actions.

4.2 Interface design rules

Give the Conventions and standards you will use while designing/ implementing the user interface.

4.3 Components available

Give the GUI components of your proposed system to be integrated in your finally implemented system.

5.0 Restrictions, limitations, and constraints

Outline the Special design issues which are likely to impact the design or implementation of proposed system.

6.0. References

7.0. Appendix: - in this section, Attach any additional information that supplements the design specification.