Office of Information Systems, Business Systems Support Office

SYSTEM DESIGN

Template

Version 1.1 ● June 30, 2012

NOTE: Please remove this page when creating a System Design deliverable.

Using This Template

To create a deliverable from this template:

1. Delete the template title page (previous page) and this page.
2. Replace [bracketed text] on the cover page (next page) with your project and agency information.
3. Replace [bracketed text] in the tool header area at the top of page i (Contents page) with the same project and agency information as on the cover page.

   Note: Please do not remove or modify content in the footer area.
4. Complete the entire template. Each section contains abbreviated instructions, shown in italics, and a content area. The content area is marked with a placeholder symbol (⇒) or with a table. Relevant text from other project deliverables may be pasted into content areas.

   Note: Please do not remove the italicized instructions.
5. Update the table of contents by right-clicking and selecting “Update Field,” then “Update entire table.”

Template Revision History

Version	Date	Name	Description
1.0	2/5/2012	PDM Project Team	Original Creation
1.1	6/28/2012	Donna Young/David Davis	Major content revision

NOTE: Please remove this page when creating a System Design deliverable.

PROJECT DELIVERY METHODOLOGY

SYSTEM DESIGN

[Agency/Organization Name]

[PROJECT NAME]

VERSION: [Version Number] REVISION DATE: [Date]

Approval of the System Design indicates an understanding of the purpose and content described in this deliverable. By signing below, each individual agrees with the content contained in this deliverable.

Approver Name	Title	Signature	Date

Contents

Section 1 Introduction

1.1 Purpose

The purpose of System Design document is to translate the business requirements and business processes into a technical design that will be used to develop the application.

Section 2 Data Model Design

Provide a reference to the location of the Data Model.

Data Model Location	

Section 3 Process Model Design

This documentation translates the business process model defined in requirements into a technical application/system process model. The Process model will include such items as:

i. *Process Definitions – Identify the process name and purpose. Include the process decomposition by providing the inputs, activities and output for each process.*

ii. *Process Flow - Show how each processes flows from one process to the next.*

Provide a reference to the location of the actual Process Model that contains the process definitions and process flows.

Process Model Design location	

⇒

Section 4 Create/Read/Update/Create (CRUD) Matrix (Interaction Analysis)

This matrix illustrates the interaction between the Process Model and the Data Model. It helps ensure that each process is directly affecting the correct entity as required and that each entity has associated processes. Provide a reference to the location of or provide the actual CRUD matrix.

CRUD Matrix Location	

⇒

Section 5 Detailed Design

This documentation represents the visual solution for the application look and feel from a User Interface, Batch and Reporting perspective. It will establish the detail needed to completely define the various technical modules that must be produced to create the new system. Translate all requirements into a set of documentation forming the framework which will be used in application development, testing, and implementation activities.

5.1 User/Batch/Report Interface Design Overview

Provide a high-level description of the user interface design (user/report and or batch) for this system. Describe any systems requirements (e.g., performance or usability) associated with all of the user interfaces.

⇒

5.2 User Interface Navigation Hierarchy

Provide and describe a diagram of the navigation hierarchy that illustrates how a user moves through the user interface.

⇒

5.3 User Interface Functional Screen/Report and/or Batch process

Describe User Interface components in terms of functions, where, each function represents one or more screens, reports, or batch processes that encapsulates the function to be performed by the user.

7.3. *X.y* Function *X Screen/Report Format/Batch Process/Other User Interface XX*

Provide a description, and if appropriate, an image or mockup of each screen, report, batch function or other user interfaces within this function.

⇒

7.3. *X.y.1* Function *X Screen/Report Format/Batch Process/Other User Interface XX* - Fields

Provide a Screen/Other User Interface Fields Table that includes descriptive information for each field that appears on the screen or other user interface within the function or use case.

In addition, if the data is selected from a pick list, include the list of possible values or their description. If the content of a field is derived from client side calculations using other fields or values, then specify the algorithm for the calculation in a descriptive footnote to the table. If the content of a field is derived from server side calculations or lookups, then specify the source of that calculation (e.g., the class or stored procedure where the calculation occurs).

Also, specify the error messages to be displayed when the input does not meet requirements for the field.

⇒

5.4 Other Interfaces

Customize the following subsections to accurately and comprehensively document the design of any additional interfaces not described in the previous sections. Identify the technology that will be used to enable the interaction. Label each subsection appropriately and title each subsection descriptively to indicate the interface being documented.

5.*4* Interface *X*

Describe the interface design including technology, the protocol, any specific message formats, error conditions, handshakes, initiation and closure, and other features that define the design of the interface.

⇒

Section 6 Technical Architecture

This section of the document describes the System Architecture in terms of hardware and the Technical Architecture direction and overall technical solution in terms of class diagrams, objects and components.

6.1 System Architecture

Describe the technical direction, overall technical solution in terms of platform(s), operating system(s), network, language(s), hardware, supporting software and tools to be used in the creation and maintenance of the new application.

⇒

6.2 Technical Architecture Design

6.2.1 CLASS DIAGRAM

Provide a Unified Modeling Language (UML) based type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes.

⇒

6.2.2 OBJECT MAPPING

Provide the process of unifying data entities & classes. This creates, in effect, a "virtual object database" that can be accessed in the application. For this deliverable, typically the core and any special objects are mapped for future use.

⇒

6.2.3 COMPONENT DIAGRAM

Describe *how components are connected together to form the application. Example components could include:*

i. EBOs
ii. Services
iii. Dependencies

⇒

6.2.4 INTERFACE CONTROL DOCUMENT-

Describe how the application will communicate with external interfaces. Include configuration, inputs & outputs, exception handling, and other relevant notes.

⇒

6.2.5 SEQUENCE DIAGRAMS,

Show how processes operate with one another and in what order. Depict the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

⇒

6.3 System Development/Testing Environment

*For each environment necessary for system construction (**Development, QA** and **Acceptance**), provide detailed specifications for the **Workstation** and **Supporting Infrastructure** that will be used (including hardware, network and operating system requirements, all necessary installed packages and tools, and needed directory structures that will be utilized to store all construction components).*

6.3.1 Development Environment

a. Developer Workstation Configuration

b. Supporting Development Infrastructure Configuration

6.3.2 *QA* Testing Environment

a. QA Workstation Configuration

b. Supporting QA Infrastructure Configuration

6.3.3 User Acceptance Environment

a. User Acceptance Workstation Configuration

b. User Acceptance Infrastructure Configuration

Section 7 Requirements Traceability Matrix

Provide reference to the location of the Requirements Traceability Matrix that indicates traceability from the requirements documented in the Software Requirements Specifications to the design elements documented in the System Design Description.

Requirements Traceability Matrix Location	

Section 8 References

Provide a list of all documents and other sources of information referenced in the System Design document and utilized in developing the System Design document. Include for each the document number, title, date, and author.

Document No.	Document Title	Date	Author

Section 9 Glossary

Define of all terms and acronyms required to interpret the System Design document properly.

⇒

Section 10 Document Revision History

Identify changes to this document

Version	Date	Name	Description

Section 11 Appendices

Include any relevant appendices.

⇒