Software Architecture Design Document

<Project Name>

<Department Name>

<Version Number>

Document history

Date	Version	Author	Reviewed by	Approved by	Description

Table of contents

1. Introduction

1.1. Project Background

<Mentioned background of the project. Highlight the main purpose of this application and also include different module names if any>

1.2. Project Objectives

<Brief Objectives of the Project>

1.3. Purpose of this Document

1.4. Structure of this Document

2. Architecture

2.1. Architecture overview

Give a general description of the system, from the point of view of the user:

- In what environment it works (home, near patient bed, operating room ...)
- Who the users are
- What it is for,
- The main functions,
- The main interfaces, inputs and outputs.

If your software is integrated in a larger system, you may reference a document that describes this system.

2.2. Physical architecture overview

Describe the hardware components on which software runs and their interactions/relationships

Use components diagrams, deployment diagrams, network diagrams, interface diagrams...

2.3. Logical architecture overview

Describe the top level software components and their interactions/relationships.

Use UML package diagrams and/or layer diagrams and/or interface diagrams.

Describe also the operating systems on which the software runs.

3. Architecture Components

3.1. Data model overview

<This section describes why this data model is required (i.e. what features are driving the need), and the overall feature/capability that the database(s) is a part of (relationship to components). It should mention if data is shared with other systems/components/applications.

The scope is all data within the solution including operational/configuration, metadata and logs, etc. that may be in flat files. These should be described for completeness and also because with a distributed environment often central servers are used for such data as well as directory services, push/pull, and syncing techniques. Understanding how it will be managed and accurately maintained is important.

Applications that may be part of the broad solution should have their own data model. The information that is captured here should compliment the other data model documents (not repeat them) and be sufficient to explain the implementation. The detail sections should support any development efforts.

If there are multiple components (databases) in this solution provide a diagram here showing the relationships of the various components. Also provide a description of how the components work together via use cases or other means of describing the concept of operation. This overview should be sufficient to provide a transition and introduction to the more detailed information that follows.>

3.1.1. Data architecture strategy

<Describe here the principles used in designing the data model: what were the important criteria and what is being done to meet that criteria within the design (e.g. minimise maintenance, support certain volumes, completely leverage a legacy DB without modification, replace a legacy DB, combine several existing DBs, reduce data redundancy, identifiable improvements from the embedded environment)? This may be a table or bullet list with the criteria described and then the strategy for meeting it. Basically this should provide the rationale for the overall design.

Describe the constraints, tradeoffs, alternatives that were considered, major decisions/selections made and the rationale for the decision/selection.>

3.1.2. Data model

<This section describes the data model from a business point of view. A graphic model showing the main relationships (including cardinality) between the fundamental elements in the solution should be included. Constraints in the model also should be established. Large-scale data-centric software may require the definition of a dedicated logical data model. If such a model already exists that document should be referenced.>

3.1.3. Enhanced/Modified data stores

<Provide a list and a brief introduction of any modified data stores (databases) that are within the architecture – why the DB is being modified, etc.>

3.1.4. Leveraged data stores

<Provide a list and a brief introduction of any data stores (databases) that are within the architecture – that exist and will not be modified.>

3.1.5. Data access

<Describe any common method, approach, data abstraction, object hierarchy, or class structure that will be developed/used for programmatic data access. If no such mechanisms will be used say so and describe the basic data access approach.>

3.1.6. Data dictionary

<Provide an overview of the data dictionary and reference the GEA 2.0 EDM section for taxonomy and nomenclature>

3.2. Detailed data model

3.2.1. Component-1 (Duplicate this section for each data component)

<Provide a brief overview of the data component (database), what its data domain is purpose in the architecture, scope, etc. Include in the description what the relationship of this component has with others. Use cases or other techniques to describe the concept of operation should be used in the discussion.>

3.2.1.1. Entity-relationship diagram

<Place an ER diagram here.>

3.2.1.2. Tables

3.2.1.2.1. Table 1 (Duplicate this Section for Each Table)

<For each table in the database there needs to be a detailed design describing its contents, columns, and relationship to other tables. Should information be migrated from another application, the Source column identifies the source. For databases that are part of an installed application, only those tables related to data that will be used in the solution should be listed, and only columns containing information used in the solution need to be included.>

Table 1:Description

Column. Name	Description	Data Type	Constraints	Mandatory	Unique	Source
		Number (10), Varchar2 (20) etc.	PK, FK	yes/no	yes/no	

3.2.1.3. Indexes

<This section lists the indexes defined in the database. If databases are part of an installed application, only those indexes related to share data as part of an integration should be listed.>

Indexes

Index name	Table	Column	Unique
			yes/no

3.2.1.4. Triggers

<This section describes defined triggers in the database. If databases are part of an installed application, only those triggers handling data as part of the integration should be listed.>

Triggers

Trigger Name	Table	Column	Event	Description

3.2.1.5. Stored procedures

<This section describes the stored procedures in the database that are used in the solution. If databases are part of an installed application, only those stored procedures handling data as part of the integration should be listed.>

Stored Procedures

Store Procedure Name	Arguments	Description

3.2.1.6. Security

<This section should describe any impacts to security or vulnerabilities that may exist. It should list any roles defined in the database and their permissions.>

3.2.1.7. Performance

<This section describes any impacts on performance that are anticipated. It includes a description of either design characteristics intended to improve, or minimize the impact on performance.>

3.2.1.8. Capacity

<This section describes any impacts on capacity. It includes a description of either design characteristics intended to improve, or minimise the impact on capacity.>

3.2.1.9. Data access

Describe the method, approach, data abstraction, object hierarchy, or class structure that will be developed/used for programmatic data access. >

3.2.1.10. Error handling

<This section needs to describe any error handling built into the component. If this component generates any error messages they should be listed with a description of their likely cause and a potential remedy if appropriate. The error message specification will be needed by testers as well as during the development of operational/user documentation.>

3.2.1.11. Installation and deployment strategy

<This section describes how this database needs to be installed including required software and configuration or setting of environment variables, etc. This section should be included only for new databases developed for the solution. Installation instructions for databases that are part of an already embedded application do not need to be included in this section unless the DB has been modified.>

3.2.1.12. Data initialisation

<This section describes how the data will be initially populated. For example, for new security tables user IDs need to be added to roles and, if a column was added to an existing table it will need to be populated. Many projects involve migration of data, in which cases the sections that follow come into play. If there is a migration planned this section should describe the strategy for doing so as an introduction to the sections that follow.>

3.2.1.12.1. Cleansing

<Describe any data cleansing that is planned or if none is planned say so>

3.2.1.12.2. Conversion

<Describe any data conversion that will be necessary to migrate the data – may include data types, mapping to completely new values, or adjusting other characteristics (i.e. length, etc.). Describe the tools to be used, processes, etc. If there is no conversion planned state there is none.>

3.2.1.12.3. Migration

<Describe the data migration that will be necessary – if data will be manually entered or typed in prior to going into production describe that as well. Describe the tools to be used, processes, etc.>

3.2.2. Data management

<Describe here what data management activities are recommended – backup, defrags, log maintenance, support of disaster recovery, etc.>

3.2.3. Documentation and training impacts

<This section should describe what should be included in the documentation and training as a result of this design.>

3.2.4. Component-2…

4. Infrastructure Architecture

4.1. Infrastructure Architecture principles

4.1.1. High availability

4.1.2. Reliability

4.1.3. Scalability

4.1.4. Serviceability

4.1.5. Manageability

4.1.6. Shareability

4.1.7. DR planning

4.2. Architectural Design

4.2.1. Network core

4.2.2. Aggregation & access layer

4.2.3. Service & server farms

4.2.4. Load balanced services

4.3. Server infrastructure

4.3.1. Deployment Architecture

4.3.2. Storage services

4.3.3. Application inventory

4.3.4. Database services

5. Security design processes

<The following sections describe activities that are performed, the key types of work products produced, and resources that can be used to support these activities.>

This...

5.1. Activities

<The following sections describe activities that are performed, the key types of work products produced, and resources that can be used to support these activities.>

5.1.1. Analysis activities

< The typical areas of analysis are:

- *Work (Business): Business functions, the work that is performed, the applications that support this work, and the technology in use;*
- *Application Security Assessment: User Satisfaction versus Strategic Value; Technical Quality versus Strategic value; Technical Quality versus Technical Evolution (ease to extend and change);*
- *Application to Location: Physical locations where applications or information is accessed;*
- *Application Security to Standards: Standards/Policy employed for application security functions, software, equipment, data stores, processes, transfers (network) or development activities;*
- *Summary Application Security Assessment: Overall summary of disposition of application security, such as Keep/Tune, Asset/Build upon, Replace/Discard, Renovate/Reengineer;*
- *Application to Security Level: Levels of security/integrity that the application supports, or the level of the environment in which it operates;*
- *Security Technology: (Technology in use or planned) (Firewalls, intrusion detections);*
- *Strengths and weaknesses of current infrastructure security elements; level of standardisation; technology obsolescence; quality of security infrastructure; security infrastructure services provided and service levels, cost breakdowns, etc.;*
- *Standards to Platform: Standards implemented (partial/full/proprietary) on the various platforms), include specific versions/dates if known;*
- *Platform to Security: The technology platforms, where they operate, and the classification/ trust/integrity levels at which they operate their environment (e.g., system-high security level).*

The specific analysis process may be augmented and specialised with items specific to the scope of the project. This may include those things particular to the enterprise's business and its technological environment (e.g. centralised, distributed, global-international). The analysis team should strive to keep their terminology consistent across the enterprise. A list of references of all data sources, such as configuration management reports, inventory lists, interviews, email, and other sources, should be maintained. This will help to assess any uncertainty in the accuracy of the characterisation and help others to tap these sources again if necessary.>

6. Security infrastructure architecture design

<This section would typically be included as part of other activities in the Design stage, as it will document infrastructure security architecture component details that comprise the Department's production and non-production technical infrastructure environment. Based on the high-level information gathered to document the Current State Infrastructure Security Strategy, this section may be used to drill down into components/elements of the Department's infrastructure security architecture strategy. Document strategic alternatives and design considerations that support improved infrastructure security architecture within the Department's environment, and specify if appropriate the physical and logical aspects of corporate security standards. Considerations should support simplicity within the Department's environment, and how security administration, management, maintenance and user interfacing complies with standard service levels. Some areas that could be addressed are included below.>

This...

6.1. Security policy

<Document design considerations and policy enhancements recommendations that support best practices within the Department's environment relative to this project, and if applicable, to support the appropriate security maturity model. For Example, NIST, ISO, BS, NSA, IAM, CIAC, etc.>

This...

6.2. Security threats

<Document design considerations that will reasonably address identified threats or vulnerabilities.

For example, Single Points of Failure protection in the Access and Core Network, personnel single points of failure, Denial of Service attacks.>

This...

6.3. Network security

<Document network security design considerations relative to this project that support network security improvements within the Department's environment, and any known limitations based on current technology and infrastructure architecture. For example, all components should be maintainable remotely (e.g. from NOC). Each component of the solution must be independently serviceable, Authentication – Secure Sockets Layer (SSL), Local/Remote Access Control, Firewall protection, DMZ issues, Security Administration.>

This...

6.4. Software/Application security

<Document design considerations that will address software/application security improvements within the Department's environment, and any known limitations based on current technology and infrastructure architecture relative to this project.

For example, Application Access Control, Authentication, Security Administration.>

This...

6.5. Facilities security

<Document design considerations that support physical security improvements for buildings, computer rooms, access systems and the type of access controls used in the facilities relative to this project within the Department's environment, and any known limitations based on current technology and infrastructure architecture. In addition to how security administration, management, maintenance and user interfacing complies with standard service levels.>

This...

7. Infrastructure security architecture design

<In this section, please provide a listing of any assumptions the project team has used in providing the above information. This can be offered in text format, or provided in a table pulled from a spreadsheet. Some sample assumptions could include:

- *Technology refresh process has been outsourced, and contract review was not in-scope;*
- *Legacy application consolidation project is currently in progress;*
- *Currently there is no security policy in place>.*

This…

<The following section contains possible additional paragraphs you may wish to include in your document. Use any or all of these paragraphs, or if none, then delete the entire section.

Any of the information below may be provided in a list in the body of this document, as a reference to an appendix, or as a reference to another document.>

8. Justification of architecture

8.1. System architecture capabilities

Describe here the rationale of the hardware / software architecture in terms of capabilities:

- Performances (for example response time, user mobility, data storage, or any functional performance which has an impact on architecture)
- User / patient safety (see §4.3 and §4.4)
- Protection against misuse (see 4.4)
- Maintenance (cold maintenance or hot maintenance),
- Adaptability, flexibility
- Scalability, availability
- Backup and restore
- Hardware and Software security : fault tolerance, redundancy, emergency stop, recovery after crash ...
- Administration,
- Monitoring, audit
- Internationalization

8.2. Network architecture capabilities

- Bandwidth
- Network failures
- Loss of data
- Inconsistent data
- Inconsistent timing of data
- Cyber security

8.3. Risk analysis outputs

If the results of risk analysis have an impact on the architecture, describe here for each risk analysis output what has been done to mitigate the risk in the architecture.

Use diagrams if necessary, like architecture before risk mitigation and architecture after risk mitigation, to explain the choices.

9. Technology Stack and Server Configuration

<List Down all the technology inclusive of hardware and software used . Also mention the list of available tools>

10. Standard and regulatory References

#	Document Identifier	Document Title
[STD1]		Add your documents references. One line per document

11. Annexure

11.1. Data Dictionary Template (Mandatory)

Data Dictionary - Template.xls