YOUR COMPANY LOGO

Your Company Name

A Simple Approach To

Requirements and Systems Analysis

Date

REVISION HISTORY

Date	Version	Author	Change

TABLE OF CONTENTS

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

Purpose describes the intent of the document, which is to provide requirements and systems analysis of the project.

- *Problems or issues to resolve.*
- *Objectives or goals met with this solution.*
- *Solution to be implemented.*
- *Why the solution is being implemented.*

Briefly state the system, infrastructure, and software to which this document applies.

- *Summarize the history of system development, operations, and maintenance.*
- *Identify the project sponsor, user, developer, and support groups.*
- *Identify current and planned operating sites.*
- *High-level overview and technical summary of implementing the business requirements.*

List other relevant documents.

2   Project Information

This section of the document should contain project background information including the project's purpose, objectives, and major system functionality.

2.1   Project Description

The Project Description section provides a high-level general view of the project; background, vision, approach, timeframe, etc.

2.2   Project Approach

The Project Approach statement provides a complete description of the approach to be taken for the delivery of the project. Outline the phases of the project including the high-level activities. If applicable, describe how a specific phase relates to the overall project.

2.3   Goals, Objectives, and Scope

The objectives statement describes the goals of the project. The Project Scope statement defines the scope of the project.

List the high-level project deliverables and specify whether they are included or excluded from the project.

ID	Description	Included / Excluded

3   System Development Life Cycle (SDLC)

The System Development Life Cycle (SDLC) is used in project management to develop information systems or applications. Its purpose is to provide Information Technology (IT) project managers with the tools to help ensure successful implementation of systems or applications that satisfy strategic and business objectives. The documentation provides a mechanism to ensure that executive leadership, functional managers, and users sign-off on the requirements and implementation of the system. The process provides management with the capability to design, develop, and implement your intended system and ensure it's on time delivery and completed within budget. The development process includes the multiple stages from the initial feasibility of the concept through maintenance.

The goals of this SDLC approach are to:

- Deliver quality systems that meet or exceed customer expectations when promised and within cost estimates
- Provide a framework for developing quality systems using an identifiable, measurable, and repeatable process
- Establish a project management structure to ensure that each system development project is effectively managed throughout its life cycle.
- Identify and assign the roles and responsibilities of all involved parties, including functional and technical managers, throughout the system development life cycle.
- Ensure that system development requirements are well defined and subsequently satisfied.

The SDLC methodology will help to achieve these goals by:

- Establishing appropriate levels of management authority to provide timely direction, coordination, control, review, and approval of the system development project. Ensuring project management accountability.
- Documenting requirements and maintaining trace ability of those requirements throughout the development and implementation process.
- Ensuring that projects are developed within the current and planned information technology infrastructure.
- Identifying project risks early and manage them before they become problems.

3.1 SDLC Waterfall Model

The following is a summary of the stages used in the Documentation Consultants process:

1. Project Initiation
2. Project Planning
3. Requirements Definition
4. System Design & Development
5. Testing
6. Installation
7. Production Turnover
8. Post Production

See sample diagram below.
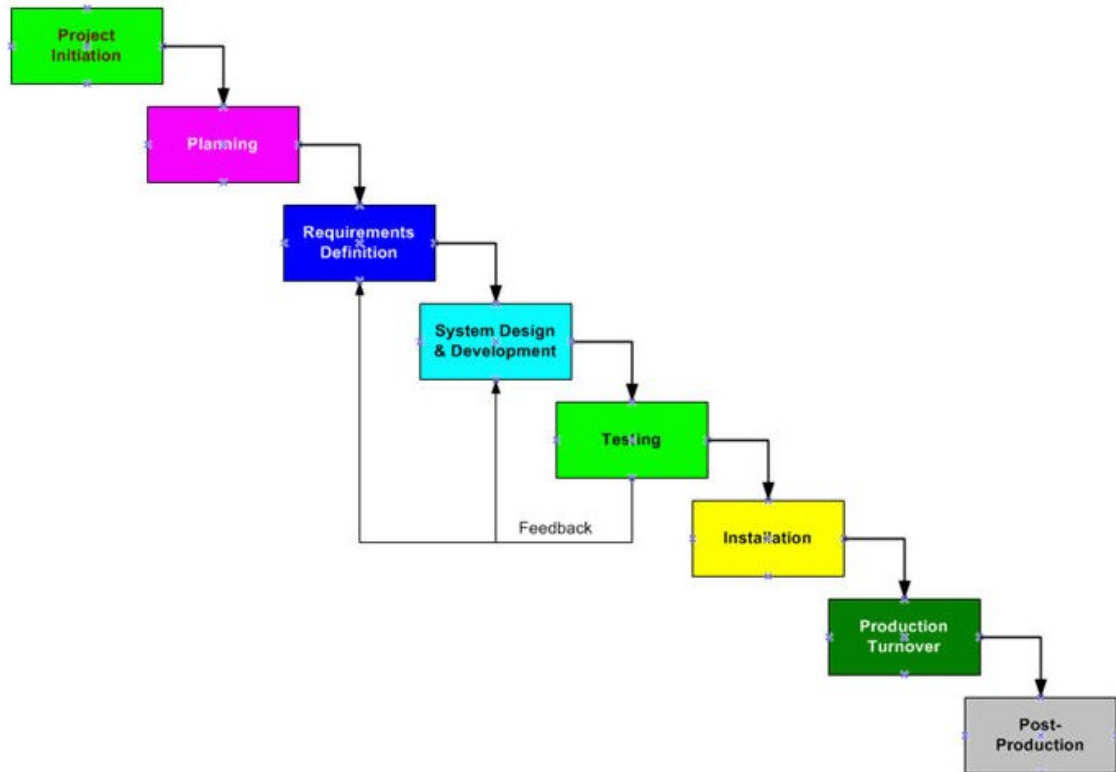
4 Project Specifications

4.1 Scope and Limitations

Topic	Content
General Area to be Studied	
Contact Name, Title, and Department	
Location and Phone	
Email Address	
Specific Functions to be Covered in Analysis	
Specific Project Constraints/Limitations	

4.2 Schedule and Location Considerations

Topic	Content
Schedule Factors:	
Project Start Considerations	
Project Completion Considerations	
Study Area Geographical Location(s)	
If Travel is Required, specify	
Will Client Provide Work Space, if yes, specify	

4.3 System Specifications

Topic	Content
Name of System	
List Current Problem(s)	
List Prime Objective(s)/Goal(s), see checklist	
Describe Specific System Constraints/Limitations	
Briefly Describe Existing Systems (e.g., programs, servers, databases, technology and other equipment)	

Topic	Content
Is the current system documented? If yes, is documentation available?	

4.4 Objective Considerations Checklist

Topic	Completed (Yes/No)
Efficiency-Reduce Waste:	
Decrease Workload	
Reduce Overtime	
Free Skilled People from Routine Work	
Reduce "Down/Off" Time	
Increase Off-line Operations	
Improve Economy-Reduce Cost:	
Reduce Number of Operations Performed	
Reduce Size or Quantity of Equipment Needed	
Reduce Manpower Requirements	
Reduce the Number of Reports Maintained	
More Accurate Information	
Increase Accuracy of Input/Output	
Select Better Reporting Elements	
Improve Distribution (right data to right people)	
Increase Capability and usefulness of equipment used	
Increase Flexibility of Processing Operations	
Improved Timeliness of Information	
Increase Throughput	
Reduce Process Turnaround Time	
Reduce Reproduction Service Time	
Reduce Distribution/Transmission Times	
Provide More Frequent Reports	
Increased Productivity with Current Resources	
Increase Hardware Utilization	
Enhance Software Utilization	
Provide Efficient Data Storage	

Topic	Completed (Yes/No)
Eliminate Redundant Operations	
Eliminate Unnecessary Operations	

4.5 Task Assignment

Task	Content
Task Number:	
Task Title:	
Task Description:	
Specific Output Required:	
Start of this Task Depends on:	
Subsequent Tasks Depend on the Completion of the following:	
Schedule:	
Start Date	
Completion date	
Resources:	
Workhours	
Materials/Services	
Other	
Assigned to:	
Assigned by:	

4.6 Resources

After key project activities and details are defined, consider the responsibilities for each individual or role in the project across every major functional department.

The responsibility matrix lays out the major activities in the project and precisely details the responsibilities of each stakeholder involved in a project. It is an important project communication tool because all stakeholders can see clearly who to contact for each activity.

4.7   Standard Roles and Responsibility Matrix

The purpose of the Roles and Responsibility Matrix is to provide a clear understanding and agreement on who does what on a project. It can be displayed with individuals or groups responsible for performing specific functions or tasks. The matrix should then be distributed for approval after its completion.

The Matrix is a method used to designate the responsibilities for each project role. It can be used to communicate Team Member roles, which help set expectations and ensure individuals know what is expected.

4.8   Sample Standard Roles and Responsibility Matrix

The following table illustrates a sample Roles and Responsibility matrix using an "X" in the department column intersecting with the respective project function.

Function / Responsibility ->	Department A	Department B	Department C
Business Case Study	X		
Feasibility Study	X		
Cost / Benefit Analysis	X		
Project Approval Document	X		
Project Charter	X		
Functional Requirements	X		
Technical Requirements	X		
Requirements Traceability Matrix		X	
Project Plan		X	
Training Plan		X	
System Design Document		X	
Process Guide		X	
Installation Guide		X	
Software User Guide		X	
System Administrators Guide		X	
Technical Test Plan			X
User Acceptance Test Plan			X
Product Acceptance Document			X
Production Turnover Maintenance			X
Modification Request			X
Product Retirement Plan			X

5 Estimate Resources

	IT LABOR ESTIMATE WORKSHEET - PROJECT		
WBS/Task #	**Task/Activity Name**	**Resource Class (BA / Dev / Arch / SA, etc)**	**Hours**
	Business Case Document		
	Obtain Approval		
	Develop Project Plan		
	Requirements Document		
	Hardware / Software Procurement		
	Design		
	Configure Development Systems		
	Database Development		
	Database Unit Testing		
	GUI Development		
	GUI Unit Testing		
	Technical Test Plan		
	Configure Test System		
	System Testing		
	User Acceptance Test Plan		
	User Acceptance Testing - Test Environment		
	User Acceptance Testing - Production Environment		
	Service Level Agreement		
	Security Plan		
	Production Readiness Document		
	Disaster Recovery Plan		
	BU Business Continuation Plan		
	Draft Implementation Plan		
	Production Turnover Mgmt. Document		
	Install / Configure Production System		
	Product Acceptance Document		
	Conduct Post-Project Review		
	Status Reporting & Reviews		
	Other Project Management		
	Develop Training Materials		
	Conduct User Training		

IT LABOR ESTIMATE WORKSHEET - PROJECT

WBS/Task #	Task/Activity Name	Resource Class (BA / Dev / Arch / SA, etc)	Hours
	Populate IT Inventory		
	Archive Test Cases in Test Case Tool		

Add up hours for each *Resource Class* in details above enter here and then transfer to Cost Estimate sheet

Resource Class (BA / Dev / Arch / SA, etc)	Total Hours

5.1 Sample Staffing Needs and Costs

Fill out the staffing Low/High Days for each resource needed - Totals will assist with calculating staffing needs.							
Recommended Staffing	**Qty**	**Low**	**High**	**Rate**	**Low Cost**	**High Cost**	**Staff**
Technical Writer 1	1	0	0	$75	$0	$0	
Technical Writer 2	1	0	0	$100	$0	$0	
UI/UX Developer	1	0	0	$125	$0	$0	
Integration Developer	1	0	0	$125	$0	$0	
Backfill General Developer	1	0	0	$125	$0	$0	
General Developer	1	0	0	$100	$0	$0	
Vendor Consultant	1	0	0	$125	$0	$0	
Vendor Consultant	1	0	0	$125	$0	$0	
Quality Assurance Analyst	1	0	0	$125	$0	$0	
Systems Administrator	1	0	0	$125	$0	$0	
Implementation Contractor	1	0	0	$125	$0	$0	
Report Developer	1	0	0	$125	$0	$0	
Labor Totals:	**12**	**0**	**0**		**$ -**	**$ -**	

5.2 Personnel and Headcount

Name	Phone	Email	Department	Title	Location

6 Processes

6.1 Current Processes

Provide Business Process Diagrams for any complex processes or steps that are used to do a specific job. Consider systems involved, users, when tasks are performed, and what the results are. Describe the process using text or a graphic process flow.

Any business rules, such as calculations, decisions / if-then, algorithms, or procedures should be clearly defined and broken down to their individual steps. Include the following information in the process diagrams:

- o *Inputs, processes, and outputs*
- o *Business workflow*
- o *Business rules, edits, validations, and/or formulas*
- o *Exception handling and processing.*

Provide detailed step information about current processes in the following table, if applicable.

	Description	User	Issues
1.			
2.			
3.			
4.			
5.			

6.2 New Processes or Future Enhancements

Provide Business Process Diagrams for any new complex processes or steps that must be performed to do a specific job. Consider systems involved, users, when tasks are performed, and what the results are. Describe the process using text or a graphic process flow.

Any business rules, such as calculations, decisions / if-then, algorithms, or procedures should be clearly defined and broken down to the individual steps. Include the following information in the process diagrams:

- *Inputs, processes, and outputs*
- *Business workflow*
- *Business rules, edits, validations, and/or formulas*
- *Exception handling and processing.*

6.2.1 New Process Flow

Provide detailed information about the new processes, including detailed step information about the new processes in the following table, if applicable.

	Description	User	Issues
1.			
2.			
3.			
4.			
5.			

7 Requirements Information

7.1 High-Level Business Requirements

This section includes details about the business requirements. Business requirements are those items needed to support user goals, tasks, and activities. It may include high-level modifications or enhancements. They can be either mandatory or desirable.

- *Mandatory: "Must Have" represents the core functionality and must be included in the solution.*
- *Desirable: "Nice to Have" requirements can be implemented after all mandatory requirements are fulfilled and there are sufficient resources available.*

This section should also include business rules that impact the project. A Business Rule defines or constrains some aspect of the business, e.g., products sold with a 50% or more discount must include an authorized approval code.

#	Function	Description	Priority	Comments

Note: Priority may be defined as Mandatory or Desirable.

7.2 System Interfaces

Provide information about the systems or applications that the solution must interface with, e.g.,

- *Communication hardware, software and their requirements, methods, and functionality.*
- *Data, formats, messages, and transfer schedules.*
- *Performance and capacity.*
- *Security designs and considerations.*
- *Names of reference manuals and other documentation along with their location.*

7.3 Infrastructure Requirements

Provide infrastructure information when there are equipment / network additions or changes, e.g., servers, printers, network devices, network configuration or management changes, new or upgraded middleware or operating system software, or changes to data centers.

8   Other Requirements Information

8.1   Product, Organizational, System, and External Requirements

This section includes details about other requirements that can cover products, organizations, systems (e.g., performance, operations), and external requirements (e.g., security).

Audience	*Indicate the users of the system or application.*
Security	*Provide information about the standard or unique security system or application situations, e.g., access / authentication / authorization processes; client, user or server certificates; encryption, password rules, and security procedures.*
Volume and Performance Metrics	*Indicate system or application volume and performance metrics to be performed, e.g., database records or transactions to be processed within a certain time period.*
Capacity and Scalability	*Indicate capacity and growth information or concerns that will have to be met over time.*
Support Considerations	*Indicate system or application support, e.g., who will support it, hardware / software to support it, service level agreements or contractual obligations.*
Audit	*Indicate any system or application audit requirements or constraints.*
Design Constraints	*Indicate system or application design constraints, e.g., software languages, developmental tools, architectural and design constraints.*
Output	*Indicate reports, files or other export output.*
Data Retention	*Indicate any special data retention requirements.*
Legal or Regulatory	*Indicate any legal or regulatory requirements that can impact the system or application.*

8.2   Usability, Performance, Operations, and Maintenance Requirements

Provide requirements information related to product usability, system performance, operations, maintenance or conditions that must be met by the proposed application or system. These requirements can have a more detailed title / description or be in a numbered list.

#	Function	Description	Priority	Comments

8.3   Content / Data / Sample Report Requirements

Include information about reports or data that must be provided or modified, e.g.,

- *Samples of all content and data.*
- *Examples of input / output files and their schema.*
- *List all necessary content with a brief description.*
- *Who will provide the information? List will provide the information and who is responsible for what and when, e.g., one-time only, other content must be maintained on a regular daily, weekly, or monthly basis.*
- *Who will maintain the information when the application is operational?*

8.4   Screen Requirements

Provide any business rules or information related to screen information that must be included.

Element	Description & Rules	Source	Req	Default Value

8.5　Training and Documentation Requirements

Provide training and documentation information that is required to support the application or system to be implemented, e.g., training plans, training materials, support documentation, and user documentation.

Training / Documentation	Resource	Schedule	Comments

9　Analysis

9.1　Design Assumptions and Support Considerations

Provide information about technology tools, environment conditions, and other support considerations that can affect project design, if applicable.

For example:

- *Behavioral design (how it will behave from a user's point of view in meeting its requirements, ignoring internal implementation).*
- *Constraints include issues such as throughput, response time, CPU utilization, etc. Use a separate sub-section for each constraint if necessary.*
- *Decisions affecting the selection and design of the system, application or software.*
- *Design decisions that respond to requirements designated critical, such as those for safety, security, or privacy.*
- *Design decisions depending on system states or modes.*
- *Design decisions regarding inputs it will accept and outputs it will produce, including interfaces with other systems.*
- *Design decisions on behavior in response to each input or condition, including actions it will perform, response times and other performance characteristics, description of physical systems modeled, selected equations / algorithms / rules, and handling of non-acceptable inputs or conditions.*
- *Design decisions on how databases / data files will appear to the user.*

9.2 Business Impact

Provide information about the impact to the business process.

9.3 Application Impact

Provide information about the impact to the application.

9.4 Current Application Behavior

Provide key information about current application behavior.

9.5 Current Software Architecture

9.5.1 Current Process Model

Provide information and a diagram or workflow of the existing process with sub-system decomposition.

9.5.2 Current User Interface

Include a copy of all wire frames and screen shots with before and after changes.

9.5.3 Current Data Model

Provide information about the current data model that will be affected by the requirements.

9.6 New Software Architecture

9.6.1 New Process Model

Provide information and a diagram or workflow of the new process with sub-system decomposition.

9.6.2 New User Interface

Include a copy of all wire frames and screen shots with before and after changes.

9.6.3 New Data Model

Provide information about the new data model. Include table information that were added, changed, or deleted with their hierarchy.

9.6.4 Procedures

Provide new and modified information, e.g.,

- *Procedures*
- *Functions*
- *Packages*
- *Sequences*
- *Database triggers.*

9.6.5 Data Changes

Provide the steps or scripts to make data related changes.

10 Architecture Design

Provide information about the architectural design of the system, application or software, e.g.,

- *Identify the software units that make up the application or system. A software unit is an element in the design; include components of major divisions and subdivisions, e.g., a class, object, module, function, routine, or database.*

- *Show the relationship(s) of the system, application or software.*

- *State the purpose of each system, application or software and identify the requirements and design decisions allocated to it.*

- *Identify each system, application or software development status / type (such as new development, existing design or software to be reused as is, existing design or software to be reengineered, software to be developed for reuse, etc.). For existing design or software, the description shall provide identifying information, such as name, version, documentation references, library, etc.*

- *Describe the application or system's planned utilization of computer hardware resources (e.g., processor capacity, memory capacity, input / output device capacity, auxiliary storage capacity, and communications / network equipment capacity). Include all computer hardware resources, in*

system-level resource allocations affecting the application or system, and resource utilization measurement planning in the development plan:

- *Requirements or system-level resource allocations being satisfied.*
- *The assumptions and conditions on which the utilization data are used (e.g., typical usage, worst-case usage, assumption of certain events).*
- *Any special considerations affecting utilization (e.g., use of virtual memory, multiprocessors or the impacts of operating system overhead)*
- *The units of measure used (such as percentage of processor capacity, cycles per second, bytes of memory, kilobytes per second).*

- *Identify program libraries and their location.*
- *Other information may include the following:*
  - *Source/Version Control*
  - *System Interaction Model*
  - *Business layer objects*
  - *Databases (type, environmental requirements, dataflow diagrams, Entity Relationship Diagram (ERD),*
  - *Tables/views*
  - *Stored procedures/macros*
  - *Functions*
  - *Integration*
  - *Presentation layer objects*
  - *Web Page Design*
  - *Reports*
- Infrastructure Information
  - *Servers (physical or virtual)*
  - *Server usage, function, model, memory, location(s), processors/type/speed*
  - *Operating systems (type, version, use)*
  - *Disk configuration (category, volume, size, raid level, location)*
  - *Backup configuration and frequency*
  - *Business continuance policies, procedures, and files*
  - *Reboot procedures*

10.1   Interface Design

Describe the interface characteristics of the system, application or software. Include both interfaces among the system, application or software and their interfaces with external entities such as systems, configuration items, and users.

(Use the Appendix Section at the end of the document if landscape paper is necessary).

10.2   Application Layer Information

Provide other relevant information that may not be included in the previous tables, e.g.,

- *Application objects, e.g., APIs, classes, reports, forms, XMLs.*
- *Service layers, e.g., business logic, data access, and navigation logic.*
- *Enterprise information system layer, e.g., external systems and databases that provide or store data.*
- *Configuration parameter changes to the configuration files.*
- *Persistent data management, e.g., persistent data stored by the system and the data management infrastructure required for it. Include the description of data schemes, the selection of a database, and the description of the encapsulation of the database.*
- *Global variables and international parameters used in the system.*
- *Boundary conditions*
- *Start-up, shut-down, error conditions, and printing conditions*
- *Historical Data Impact.*

10.3   Implementation and Software Execution

Provide information about the execution of the system, application or software. Include diagrams and descriptions showing the dynamic relationship of the system, application or software, e.g.,

- *Flow of execution control*
- *Data flow*
- *Dynamically controlled sequencing*
- *State transition diagrams*
- *Timing diagrams*
- *Priorities among system, application or software*
- *Handling of interrupts*
- *Timing / sequencing relationships*
- *Exception handing*
- *Concurrent execution*
- *Dynamic allocation*
- *Dynamic creation / deletion of objects, processes, tasks, and other dynamic behavior.*

- *Other information, e.g., steps to follow to implement the changes including code check-out, creation of the folders if any, changing configuration, compilation of the code, placing the compiled version in the appropriate folder, verification of the implementation steps such as testing the URL, executing the code etc.*

11   Future Improvements

Provide information about the following, if applicable:
- *Future improvements and areas likely to change.*
- *Information about missing useful or desirable features and functions.*
- *Functional improvements to the package and programs.*