Project Name:

Ambiguity Review Checklist

Project Name		Project Code		Author	

Checklist Purpose	Review a Functional Specification of structural ambiguity (not to be confused with content reviews). For all negative responses the QA Project Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email.

Context	Task			
		Yes	No	Comments
Complexity	Are the requirements overly complex?			
Dangling Else	Are there cases where the else part of a condition is missing?			
Ambiguity of Referenced	Are there cases where there are references, which are not clearly defined?			
Scope of Action	Are there cases where the scope of the action for a condition is not clearly defined?			
Omissions				
	Are there causes without effects?			
	Are there missing effects?			
	Are there effects without causes			
	Are there missing causes?			
Ambiguous Logical Operators				
	Are there compound usage of and, or that are not clear?			
	Are there implicit connectors?			
	Is the use of “or” correctly used?			
Negation				
	Are there cases of scope negation?			
	Are there cases of unnecessary negation?			
	Are there cases of double negation?			
Ambiguous Statements	Are there cases of ambiguous verbs?			

Project Name:

Context	Task			
		Yes	No	Comments
	Are there cases of ambiguous adverbs?			
	Are there cases of ambiguous adjectives?			
	Are there cases of ambiguous variables?			
	Are there cases of aliases?			
Random Organization				
	Are there cases of mixed causes and effects?			
	Are there cases of random case sequences?			
Built-in Assumptions	Are there cases of functional / environmental knowledge?			
Ambiguous Precedence Relationships	Are there cases where the sequences of events are not clear?			
Implicit Cases	Are there implicit cases?			
Etc.	Are there examples of Etc.?			
I.E. Versus E.G.	Is I.E. and E.G. used correctly?			
Temporal Ambiguity	Are there cases of timing ambiguities?			
Boundary Ambiguity	Are there boundary ambiguities?			

<Company Logo>

Approvals

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Approvals

Project Name:
Project Code:
Date:

Deliverable Approvals			
Test Deliverable Name	**Approval Status**	**Approver**	**Approved Date**

Architecture Review Checklist

Project Name		Project Code		Author	

Checklist Purpose	Review the architecture for completeness and clarity. For all negative responses the Test Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email.

Area	Task			
		Yes	No	Comments
	Has an overview description of the system been documented?			
	Has the 2 or 3 tier architecture been defined?			
	Has the database and access been defined?			
	Has servers been defined?			
	Have the protocols been defined, ex. HTTP, JSP, PeopleSoft, Tuxedo?			
	Is the vendor in-house or outsourced?			
	Has the point of contact to resolve technical architecture problems been defined?			
	Has the platform been defined?			
	Is there a network diagram?			
	Has the test equipment been defined?			
	Has load balancing been defined?			
	Have the business processes been defined?			
	Are there common tasks that may be performed more often than others?			
	Have peak volumns been defined?			
	Have the Web servers been identified?			

Archive Project Documents

Project Code		Project Name		Author	

Checklist Purpose	This Checklist is used to check the correct archival of project documents. For all negative responses the QA Project Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email, depending upon the severity.

S. No	Activity	Status		
		Yes	No	Remarks
1	Is the Business Requirement Document Archived?			
2	Is the functional specification document archived?			
3	Is the Logical Design Document Archived?			
4	Is the Prototype document Archived?			
5	Are the E-Mails and correspondence relating to the project archived?			
6	Are the Minutes of all the meetings archived?			
7	Is the Test Strategy Document Archived?			
8	Is the Automation/Performance Strategy Document Archived?			
9	Is the Test Plan Document Archived?			
10	Are the documents containing the test cases and conditions archived?			
11	Are the automation scripts archived?			
12	Are the Test data guidelines/Test Data document archived?			
13	Is the Run Plan document archived?			
14	Is the Traceability matrix document archived?			
15	Are the day wise/consolidated Defect logs (screen shots) documents archived?			
16	Are the day wise/consolidated Defect report documents archived?			
18	Is the project execution tracker document archived?			
19	Is the downtime log document archived?			
20	Are the consolidated summaries of all the passes of test documents archived?			
21	Is the functionality NOT tested document archived?			
22	Is the defect metrics document archived?			
23	Is the QA project metrics document archived?			
24	Is the Project Closure Report Document archived?			

<Company Logo>

Change Request Form

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Change Request Form
Report Number: _____ Change Request No: _____
System Affected:
Subsystem Affected:
Problem Statement:
Action Required:

<Company Logo>

Clarification Request

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Clarification Request

Project Name:				**Project Code:**			
Issue No.	**Document Reference**	**Application / Function**	**Date Raised**	**Clarification Requested**	**Raised by**	**Status**	**Response**

Data Design Review Checklist

Project Name		Project Code		Author	

Checklist Purpose	Review the logical and physical design for clarity and completeness. For all negative responses the QA Project Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email, depending upon the severity.

Context	Task	Status		
		Yes	No	Remarks
Logical Design				
	Has the data been inadequately defined?			
	Are the data entity definitions incomplete?			
	Are the cardinalities defined incorrectly?			
	Are the attributes defined adequately?			
	Are there normalization violations?			
	Are the primary keys defined incorrectly?			
	Are the foreign keys defined incorrectly?			
	Are the compound keys defined incorrectly?			
	Are the entity subtypes defined incorrectly?			
	Are the parent processes incomplete?			
	Are the child processes incomplete?			
	Are the process inputs and outputs interactions with the entities incomplete?			
	Are the elementary entities defined correctly?			
	Are there parallel linkage problems?			
	Are event-trigger processes designed incorrectly?			
	Are there entity/ process			

Context	Task	Status		
		Yes	No	Remarks
	associations incorrectly defined?			
	Are there entity/process read associations incorrectly defined?			
	Are there entity/process update associations incorrectly defined?			
	Are there entity/process delete associations incorrectly defined?			

Environment Readiness Checklist

Project Name		Project Code		Author	
Checklist Purpose	This Checklist is to verify the readiness of the environment for testing before starting test execution. For all negative responses the Test Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email.				

	ITEMS TO BE CHECKED	YES	NO	COMMENTS
1.	Has the Client signed off on the Test Strategy?			
2.	Is the Test Environment Ready?			
	• Hardware			
	o <Input Each Component>			
	• Software			
	o <Input Each Component>			
3.	Is the Test Bed Created?			
	• Is Data available as per expected format (Test Data Guidelines – Planning)			
	• Is the Data actually populated?			
	• Is the populated data sufficient?			
4.	Has the Software transfer been completed and the initial version been loaded (Load Management)?			
5.	Have the User IDS and Passwords been setup to access the environment from Client/Developers?			
6.	Logistics			
	▪ Is the Testing Team Available and ready to start testing?			
	▪ Is the Test Lab setup Complete?			
7.	Is the Interaction model (Project Planning) defined and established as documented in the Test Strategy?			
8.	Is the Client aware of the Defect Management process as defined in the Strategy?			
9.	Is the Entry Criteria defined and established per the Project Strategy plan?			
	▪ <Enter Each Criteria here>			

Any Other Potential Issues:

Execution Plan

Project Name:
Project Code:

Activity No.	Activities / Sub-Tasks	Planned Date		Resource	Total Test Cases/Scripts	Test Cases/ Scripts Completed	Comments
		Start Date	End Date				

<Company Logo>

Factors Affecting Estimation Checklist

Project Name		**Project Code**		**Author**	

Checklist Purpose	Review the factors that can affect the work effort estimates.

Context	**Factor**			
		Yes	**No**	**Comments**
Estimation Factors	Understanding the Functionality of the Application – Knowledge Transfer			
	Training on the functionality and required tools for the resources			
	Clarifications raised by the Test Team on the functionalities			
	Preparation of Test Project Plan and Test Strategy			
	Preparation of Test Conditions			
	Review of Test Conditions			
	Preparation of Test Cases			
	Review of Test Cases			
	Preparation of Test Scripts			
	Review of Test Scripts			
	Preparation of Data Requirement document for Testing			
	Preparation of Test Execution Plan			
	Preparation of Traceability Document			
	Review of Test Execution Plan and Traceability			

Context	Factor			
		Yes	No	Comments
	Document			
	Verification of the Environment readiness			
	Multiple iteration of Test Execution as planned in the Strategy			
	Defect Management			
	Test Execution monitoring and management			
	Test Closure report			
	Archival of test documents			
	Lessons learned			

<Company Logo>

Final Test Summary Report

Rough Draft (Version 1.0)

Prepared B y:

Date:

Revision History

Date	Issue	Description	Author

Table of Contents

1 Introduction

1.1 Executive Summary

< Highlights of the project in terms of schedule, size and defect counts, as well as important events that happened during the life of the project, which would be of interest to the management >

1.2 Project overview

< This section covers the business of the client, overview of the project >

1.3 Scope of testing

< A note on the scope of testing and details regarding the scope of testing >

2 Test Methodology

2.1 Test documents

< A brief note on the test documents>

2.2 Test Iterations

< The details of Test Iterations carried out>

2.3 Defect Management

< A brief note explaining the Defect Management process followed during execution >

3 Measurements

3.1 Traceability Matrix

< The details of the trace from the requirements through to the scripts >

3.2 Planned Vs. Actual

< Details of Planned versus Actual schedule with reasons for variation>

3.3 Test Scripts summary

< The Final Test Scripts summary at the end of Test Execution>

3.4 Features untested/Invalid

< Details pertaining to the scripts that was untested, invalid or not delivered and the reasons>

4 Findings

4.1 Final Defect Summary

< Summary of Defects at the end of test execution>

4.2 Deferred Defects

< Details of test cases that failed and are in deferred status with reasons for deferring the defect>

5 Analysis

5.1 Category-wise Defects

< A chart should be generated to display the count of defects category-wise >

5.2 Status-wise Defects

< A chart should be generated to display the count of defects Status wise >

5.3 Severity-wise Defects

< A chart should be generated to display the count of defects severity-wise >

5.4 Issues

< Details of issues experienced during the course of the project, that were documented and highlighted to management>

5.5 Risks

< Defects reported should be analyzed and any risks that could affect the business that we foresee>

5.6 Observations

< Any other critical events that cannot be classified under issues and risks>

6 Test team

< Names and roles of personnel from all parties involved during the project>

7 Appendices

< Appendices, as referred to in any of the sections above should be mentioned here >

Function/ Test Matrix

Test case

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Business Function																									

<Company Logo>

Function
Test Plan

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Table of Contents

A. BACKGROUND

B. INTRODUCTION

C. ASSUMPTIONS

D. TEST ITEMS

List each of the items (programs) to be tested.

E. FEATURES TO BE TESTED

List each of the features (functions or requirements) which will be tested or demonstrated by the test.

E. FEATURES NOT TO BE TESTED

Explicitly lists each feature, function, or requirement which won't be tested and why not.

F. APPROACH

Describe the data flows and test philosophy.
Simulation or Live execution

H. ITEM PASS/FAIL CRITERIA

Itemized list of expected output and tolerances

I. SUSPENSION/RESUMPTION CRITERIA

Must the test run from start to completion?
Under what circumstances may it be resumed in the middle?
Establish check-points in long tests.

J. TEST DELIVERABLES

What, besides software, will be delivered?
Test report
Test software

K. TESTING TASKS Functional tasks

Administrative tasks

L. ENVIRONMENTAL NEEDS

Security clearance
Office space & equipment
Hardware/software requirements

M. RESPONSIBILITIES

Who does the tasks in Section 10?
What does the user do?

N. STAFFING & TRAINING

O. SCHEDULE

P. RESOURCES

Q. RISKS & CONTINGENCIES

R. APPROVALS

Functional Specification Review Checklist

Project Code		Project Name		Author	

Checklist Purpose	Review a Functional Specification for content completeness and clarity (not to be confused with ambiguity reviews. For all negative responses the QA Project Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email.

Context	Task	Yes	No	Comments
Introduction				
	Are the purpose, scope and organization of the functional specification documented?			
Software Overview				
Product Description	Are there a description of why the product is being developed and a list of the important features and capabilities?			
Product Functional Capabilities	Is there a list of the functions that the software will be required to perform?			
	For several function capabilities, is there a table (or some other format) to illustrate the relationships between the functional capabilities? Note: this may be an update to the Requirements documentation			
User Characteristics	Are the intended user of the software in terms of job function,, specialized knowledge, or skill levels described?			
User Operations and Practices	Is there a description of how the users will normally use the software, and the tasks they will frequently perform described?			
General Constraints	Are algorithm limitations, user interface limitations, and data limitations			

Context	Task	Yes	No	Comments
	described?			
Assumptions	Are all the assumptions described?			
Other Software	Is there a description of how the system interfaces with other software described?			
Specific Functional Descriptions				
Description	Is the role of each function described?			
Inputs	Are all input sources specified?			
	Is all input accuracy requirements specified?			
	Is all input range values specified?			
	Are all input frequencies specified?			
	Are all input formats specified?			
Processing	If calculations using methods or specific standards used, are they referenced?			
	Are database definitions included?			
Outputs	Are the outputs of the function described?			
	Where there is a user interface, is it included?			
	Are all output destinations specified?			
	Is all output accuracy requirements specified?			
	Is all output range values specified?			
	Are all output frequencies specified?			
	Are all output formats specified?			
Reports				
	Are all report formats specified?			
	Are all calculations/formulas used in reports specified?			
	Are all report data filter requirements specified?			
	Are all report sorting			

Context	Task			
		Yes	No	Comments
	requirements specified?			
	Is a report totaling requirements specified?			
	Are all report formatting requirements specified?			
Non-Functional				
	Are all performance requirements specified for each function?			
	Are all design constrains specified for each function?			
	Are all attributes specified for each function?			
	All security requirements specified for each function?			
	Are all maintainability requirements specified for each function?			
	Are all database requirements specified for each function?			
	Are all Operational requirements specified for each function?			
	Are all installation requirements specified for each function?			
Interfaces				
	Are all user interfaces specified?			
	Are all batch interfaces specified?			
	Are all Hardware interfaces specified?			
	Are all Software interfaces specified?			
	Are all communications interfaces specified?			
	Are all interface design constraints specified?			
	Are all interface security requirements specified?			
	Are all interface maintainability requirements specified?			
	Are all human-computer interactions specified for user interfaces?			
	Have all internal interfaces been identified?			

Context	Task	Yes	No	Comments
	Have all internal interfaces characteristics been specified?			
	Are error message requirements described?			
	Is input range checking requirements described?			
	Is the order of choices and screens corresponding to user preferences defined?			
Additional Requirements				
Database	Are any specific requirements relating to the database, such as data base type, capability to handle large text fields, real-time capability, multi-user capability and special requirements relating to queries and forms defined?			
Administration	Are there any periodic updating or data management requirements defined?			
User Documentation	Are there user-documentation requirements to be delivered with the software defined?			
Other Requirements	Are there requirements not already covered above that need to be considered during the design of the software?			
Timing				
	Are all expected processing times specified?			
	Is all Data transfer Rates specified?			
	Is all system through put rates specified?			
Hardware	Is the minimum memory specified?			
	Is the minimum storage specified?			
	Is the maximum memory specified?			
	Is the maximum storage specified?			
Software	Are the required software environments/OS's			

Context	Task			
		Yes	No	Comments
	specified?			
	Are all of the required software utilities specified?			
	Are all purchased software products that are to be used with the system specified?			
Network	Is the target network specified?			
	Are the required network protocols specified?			
	Is the required network capacity specified?			
	Is the required/estimated network throughput rate specified?			
	Is the estimated number of network connections specified?			
	Are minimum network performance requirements specified?			
	Are the maximum network performance requirements specified?			

Impact Analysis Checklist

Project Name		Project Code		Author	

Checklist Purpose	This checklist help analyze the impacts of changes to the system. For all negative responses the Test Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email.

Context	Activity			
		Yes	No	Comments
1	Is the Enhanced Business Requirement available?			
2	Is the New Functional Specification Document for new requirements available?			
3	Have You understood the additional/new requirements?			
4	Is the Prototype document for new release Available?			
5	Are you able to identify the proposed changed?			
6	Are you able to identify the applications affected by the enhancements?			
7	Has the test Scope been adequately defined for the enhancements?			
8	Have the test conditions/cases been prepared for the enhancements and impacted application areas?			
9	Have you prepared a Test Plan/Strategy			
10	Have you prepared the Test data requirements for all the conditions/cases?			
11	Has the automation scope for the new/additional requirements been completed?			
12	Has the impact on the existing scripts analyzed?			
13	Has the Test Execution Plan been document for the new release?			
14	Is the Traceability matrix document been prepared?			
15	Are there any architectural changes due to new requirements?			
16	Are there any changes to the database(s) due to new requirements?			
17	Are the GUI changes due to new requirements been analyzed?			

EFFORT FOR Model Project

No.	Resource	Test Planning & Scripting	Test Execution	Test Closure	Total
		(All effort in Person Days)			
1	Test Engineers				
2	Project Manager/Test Lead				
	Total Person Days				
	Total Person Months	60.0	30.0	10.0	100.0
	Ratio	60.0%	30.0%	10.0%	100.0%
	Person Months (Only TE Effort)	0	0	0	0
	Team Size	4	3	0	7

Meeting Agenda

Meeting Agenda

DATE: **TIME:** **LOCATION:**
FACILITATOR: Phil Maternowski **RECORDER:**
MEETING OBJECTIVE:

HANDOUTS:

ATTENDEES: "X" indicates that the person was present.

( )	( )	( )
( )	( )	( )
( )	( )	( )
( )	( )	( )
( )	( )	( )
( )		

AGENDA

Time	Item	Presenter

NOTES

<Company Logo>

Metrics Definition

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Metrics

Project Name:
Project Code:
Date:

Metric No.	Metric Category	Metric Point	Metric Objective	Derivation

Minutes of the Meeting

Meeting Purpose		**Meeting Date**	
Start Time		**End Time**	
Attended By			
Distribution List			

Important Discussions:

Discussion Item #1	Details	Comments

Discussion Item #2	Details	Comments

Discussion Item #3	Details	Comments

Discussion Item #4	Details	Comments

Action Items:

a.
b.
c.
d.
e.
f.
g.
h.
i.

j.
k.

Project Baselines

(Values can be changed depending on the project requirements)

Condition to Case	
Simple	**1**
Medium	**3**
Complex	**5**
Buffer	20%

Project Schedule	
Planning	Execution
35	25

Test Case	Script
10	1

No of Test Cases Per day	
Planning	Execution
30	15

No of Test Scripts Per day	
Planning	Execution
2	1
Timelines	
Day-Hr	8
Week-Day	5
Month-Day	22

Project Completion Checklist

Project Name		Project Code		Author	
Checklist Purpose	This is checklist confirms that all the required key activities have been carried				

Context	Activity	Status			Comments
		Yes	No	Required/ Optional (R,O)	
	Are all the test cases are executed?				
	Are all the defects are either closed or deferred?				
	Are all Change Requests are closed?				
	Is the soft base delivered certified?				
	Has user training been completed?				
	Are the deliverables are handed over to the customer?				
	Is project Sign-off obtained from the customer?				
	Does the Project Directory contain the latest version of the documents?				
	Are all the documents archived and put in data warehouse?				
	Have Customer Feedback Forms been sent to Customer?				
	Has the Customer Supplied material been returned or released to other projects and the same communicated to the Customer?				
	Has the Forma*l Project Closure* communicated? (Customer, Senior Manager, Onsite Team, Quality Team, Inter Groups and Project Team)				
	Have the project directories been backed up?				
	Have the media been stored in a fireproof cabinet?				
	Has the Project Directory been withdrawn from the server?				
	Has the Project been marked as Closed in project database?				
	Have all Metric data collection been completed?				
	Has the Skill database been updated?				

Project Information Gathering Checklist

Project Name		**Project ID**		**Author**	

Checklist Purpose	This checklist is used to verify the information available and required at the beginning of the project. For all negative responses the QA Testing Manager should assess the impact and escalate as an issue. This can be accomplished through weekly status reports and/or Email.

Item No.	Activity	Yes	No	Comments
1	**Proposal Phase**			
	Is the QA team prepared to make QA estimates?			
	Has the Proposal been reviewed & approved?			
	Has Estimation and Risk assessment been completed?			
	Have Initial Work Products been sent to the Project server?			
2	**Vendor Contract (if applicable)**			
	Has the Contract been reviewed and approved?			Need process defined in QA Project Plan
	Does the Master contract and proposal exist?			
	Has the Proposal and communications been defined?			
	Has the Project Acceptance Notes / Communication been defined?			
3	**Project Initiation**			
	Has a Project Folder has been created?			
	Has the Project Manager been trained on his role?			
	Has the Project Kick-off meeting been completed?			
	Was the Manager who made the proposal present in the Project kick-off meeting?			
4	**Project Plan & Scheduling**			
	Have Audits & Reviews been planned?			
	Have the Project goals been identified?			
	Has Configuration Management been discussed?			
	Has the Staffing Plan been discussed?			
	Has the Training Plan been discussed?			
	Has the Status reporting method and frequency been discussed?			
	Has the Project scheduling been discussed?			
	Does the Project schedule include all the activities in the project?			

Item No.	Activity	Yes	No	Comments
	Has the QA Project Management plan has been reviewed by the Project Manager and others?			
	Has the QA Project schedule has been reviewed by the team?			
5	**Testing Process Overview**			
	Has the Testing process been reviewed & approved by Project Manager?			
6	**Project Folder**			
	Has the Estimation & Risk been discussed?			
	Have the Roles & responsibilities been discussed?			
	Have the Critical resources are planned?			
	Have the Project dependencies are identified?			
	Has the Project Folder is reviewed be the Quality Test for completeness?			

Project Status Report

Purpose	This template consolidates the QA Project related activities in all Key Process Areas. It is published to all project stakeholders weekly.

Project Name		**Project Code**	
Project Start Date		**Project Manager**	
Project Phase		**Week No. & Date**	
Distribution			

Key Activities

	Details	**Remarks**
Deliverables		
Decisions		

Weekly Progress for This Week

Item	**Key Processes**	**Activities/ Milestone, Deliverable**	**Planned**		**Actual**		**Status/Remarks**	**Owner**
			Start date	**End date**	**Start date**	**End date**		

Unplanned Activities

Item	**Activities**	**Start date**	**End date**	**Effort (Person Hours)**	**Comments**

Activities Planned for Next Week

Item	**Activities**	**Start date**	**End date**	**Effort (Person Hours)**	**Comments**

Planned but not completed

Change Requests (New)
Change Requests (Outstanding)
Issues (New)
Issues (Outstanding)

Prototype Review Checklist

Project Name		Project Code		Author	

Checklist Purpose	Review a prototype for content completeness and clarity. For all negative responses the Test Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email.

Context	Item	Yes	No	Comments
	Does the prototype reflect the initial client requirements?			
	Does the prototype design reflect the initial requirements?			
	Has a detailed interactive/visual user interface been created?			
	Is there an easy connection of the user interface components to the underlying functional behavior?			
	Does the prototyping tool provide an easy to learn language?			
	Is modification to the resulting prototyping tool language easy to perform?			
	Simplicity: Does the user interface provide an appropriate means of allowing a client to assess the underlying functional behavior as described by the initial requirements?			
	Is the prototype simple to use?			
	Conciseness: Does the prototype contain full-scale user interfaces without extraneous details?			
	Does the prototype contain a data model defining the data structures for the application itself?			
	Is the volatility/ persistence of the data represented?			

	Does the prototype accommodate new requirements?			
	Does the prototype address poorly defined requirements?			

<Company Logo>

Quality Assurance Plan

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Table of Contents

1. Purpose Section

This section delineates the specific purpose and scope of the particular SQA plan. It should list the name(s) of the software items covered by the SQA plan and the intended use of the software. It states the portion of the software life cycle covered by the SQA plan for each software item specified.

2. Reference Document Section

This section provides a complete list of documents referenced elsewhere in the text of the SQA plan.

3. Management Section

This section describes the project's organizational structure, tasks, and responsibilities.

4. Documentation Section

This section identifies the documentation governing the development, verification and validation, use, and maintenance of the software. It also states how the documents are to be checked for adequacy. This includes the criteria and the identification of the review or audit by which the adequacy of each document will be confirmed.

5. Standards, Practices, Conventions, and Metrics Section

This section identifies the standards, practices, conventions, and metrics to be applied, and also states how compliance with these items is to be monitored and assured.

6. Reviews and Inspections Section

This section defines the technical and managerial reviews, walkthroughs, and inspections to be conducted. It also states how the reviews, walkthroughs, and inspections, are to be accomplished including follow-up activities and approvals.

7. Software Configuration Management Section

This section is addressed in detail in the project's software configuration management plan.

8. Problem Reporting and Corrective Action Section

This section is addressed in detail in the project's software configuration management plan.

9. Tools, Techniques, and Methodologies Section

This section identifies the special software tools, techniques, and methodologies that support SQA, states their purposes, and describes their use.

10. Code Control Section

This section defines the methods and facilities used to maintain, store, secure, and document the controlled versions of the identified software during all phases of development. This may be implemented in conjunction with a computer program library and/or may be provided as a part of the software configuration management plan.

11. Media Control Section

This section states the methods and facilities to be used to identify the media for each computer product and the documentation required to store the media, including the copy and restore process, and protects the computer program physical media from unauthorized access or inadvertent damage or degradation during all phases of development. This may be provided by the software configuration management plan.

12. Supplier Control Section

This section states the provisions for assuring that software provided by suppliers meets established requirements. In addition, it should state the methods that will be used to assure that the software supplier receives adequate and complete requirements. For previously developed software, this section will state the methods to be used to assure the suitability of the product for use with the software items covered by the SQA plan. For software to be developed, the supplier will be required to prepare and implement an SQA plan in accordance with this standard. This section will also state the methods to be employed to assure that the developers comply with the requirements of this standard.

13. Records Collection, Maintenance, and Retention Section

This section identifies the SQA documentation to be retained. It will state the methods and facilities to assemble, safeguard, and maintain this documentation, and will designate the retention period. The implementation of the SQA plan involves the necessary approvals for the plan as well as development of a plan for execution. The subsequent evaluation of the SQA plan will be performed as a result of its execution.

14. Testing Methodology

This section defines the testing approach, techniques, and automated tools that will be used.

Requirements Review Checklist

Project Name		**Project Code**		**Author**	

Checklist Purpose	Verify that the testing project requirements are comprehensive and complete. For all negative responses the Test Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email.

Context	**Task**	**Status**		
		Yes	**No**	**Comments**
Clarity				
	Are the requirements written in non-technical understandable language?			
	Is each characteristic of the final product described with a unique terminology?			
	Is there a glossary in which the specific meaning(s) of each term is (are) defined?			
	Could the requirements be understood and implemented by an independent group?			
Completeness				
	Is there an indexed table of contents?			
	Are all figures, tables, and diagrams labeled?			
	Are all figures, tables, and diagrams cross-referenced?			
	Are all of the requirements defined?			
	Are all of the requirements related to functionality included?			
	Are all of the requirements related to performance included?			
	Are all of the requirements related to design constraints included?			
	Are all of the requirements related to attributes included?			

Context	Task	Status		
		Yes	No	Comments
	Are all of the requirements related to external interfaces included?			
	Are all of the requirements related to databases included?			
	Are all of the requirements related to software included?			
	Are all of the requirements related to hardware included			
	Are all of the requirements related to inputs included?			
	Are all of the requirements related to outputs included?			
	Are all of the requirements related to reporting included?			
	Are all of the requirements related to security included?			
	Are all of the requirements related to maintainability included?			
	Are all of the requirements related to criticality included?			
	Are possible changes to the requirements specified?			
Consistency				
	Are there any requirements describing the same object that conflict with other requirements with respect to terminology?			
	Are there any requirements describing the same object that conflict with respect to attributes?			
	Are there any requirements that describe two or more actions that conflict logically?			
	Are there any requirements that describe two or more actions that conflict temporally?			
Traceability				
	Are all requirements traceable back to a specific user need?			
	Are all requirements traceable back to a specific			

Context	Task	Status		
		Yes	No	Comments
	source document or person?			
	Are all requirements traceable forward to a specific design document?			
	Are all requirements traceable forward to a specific software module?			
Verifiability				
	Are any Requirements included which are impossible to implement?			
	For each requirement is there a process that can be executed by either a human or a machine to verify the requirement?			
	Are there any requirements that will be expressed in verifiable terms at a later time?			
Modifiability				
	Is the requirements document clearly and logically organized?			
	Does the organization adhere to an accepted standard?			
Content				
General				
	Is each requirement relevant to the problem and its solution?			
	Are any of the defined requirements really designing details?			
	Are any of the defined requirements really verification details?			
	Are any of the defined requirements really project management details?			
	Is there an introduction section?			
	Is there a general description section?			
	Is there a scope section?			
	Is there a definitions, acronyms, and abbreviations			

Context	Task	Status		
		Yes	No	Comments
	section?			
	Is there a product perspective section?			
	Is there a product functions section?			
	Is there a user characteristics section?			
	Is there a general constraints section?			
	Is there an assumptions and dependencies section?			
	Is there a specific requirements section?			
	Are all of the necessary appendixes present?			
	Are all of the necessary figures present?			
	Are all of the necessary tables present?			
	Are all of the necessary diagrams present?			
Reliability				
	Are the consequences of software failure specified for each requirement?			
	Is the information to protect from failure specified?			
	Is a strategy for error detection Specified?			
	Is a strategy for correction specified?			
Hardware				
	Is the Hardware details specified?			
Software				
	Are the required software details specified?			
Communications				
	Are the required Communication/ network details specified			

<Company Logo>

Requirements Specification

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Table of Contents

3.1.1 User Interfaces

3.1.2 Hardware Interfaces

3.1.3 Software Interfaces

3.1.4 Communications Interfaces

3.2 System Features

3.2.1 System Feature 1

3.2.1.1 Introduction/Purpose of Feature
3.2.1.2 Stimulus/Response Sequence
3.2.1.3 Associated Functional Requirements
3.2.1.4 Introduction/Purpose of Feature
3.2.1.5 Stimulus/Response Sequence
3.2.1.6 Associated Functional Requirements
3.2.1.6.1 Functional Requirements 1
.
.
.
3.2.1.6.n Functional Requirements n

3.2.2 System Feature 2

.
.
.

3.2.m System Feature m

.
.

3.3 Performance Requirements

3.4 Design Constraints

3.5 Software System Attributes

3.6 Other Requirements

4. Supporting Information

4.1 Table of Contents and Index

4.2 Appendices

Retest Matrix

Test case

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Business Function																									

<Company Logo>

Sample Test Plan

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

<Company Logo>

Screen Data Mapping

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Screen Data Mapping

Item No.	Test Case ID	Screen Reference (Optional)	Field Name	Data Required	Data Type	Data Format	Comments
1	TS-01		Account Number	aabbcc	alphabets		
2	TS-01		Account Number	10099	numeric	######	
3	TS-05		As-of-Date	31101999	date	dd-mm-yyyy	

<Company Logo>

System/Acceptance Test Plan

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Table of Contents

I. Introduction

a. System Description

(i.e., brief description of system)

b. Objective

(i.e., objectives of the test plan)

c. Assumptions

(e.g., computers available all working hours, etc.)

d. Risks

(i.e., risks if unit testing is not completed)

e. Contingencies

(e.g., backups procedures, etc.)

f. Constraints

(e.g., limited resources)

g. Approval Signatures

(e.g., authority to sign-off document)

II. Test Approach and Strategy

a. Scope of Testing

(i.e., tests to be performed)

b. Test Approach

(e.g., test tools, black-box)

c. Types of Tests

(e.g., unit, system, static, dynamic, manual)

d. Logistics

(e.g., location, site needs, etc.)

e. Regression Policy

(e.g., between each build)

f. Test Facility

(i.e., general description of where test will occur)

g. Test Procedures

(e.g., defect fix acceptance, defect priorities, etc.)

h. Test Organization

(e.g., description of QA/test team)

i. Test Libraries

(i.e., location and description)

j. Test Tools

(e.g., capture/playback regression testing tools)

k. Version Control

(i.e., procedures to control different versions)

l. Configuration Building

(i.e., how to build the system)

m. Change Control

(i.e., procedures to manage change requests)

III. Test Execution Setup

a. System Test Process

(e.g., entrance criteria, readiness, etc.)

b. Facility

(e.g., details of test environment, lab)

c. Resources

(e.g., staffing, training, timeline)

d. Tool Plan

(e.g., specific tools, packages, special software)

c. Test Organization

(e.g., details of who, roles, responsibilities)

IV. Test Specifications

a. Functional Decomposition

(e.g., what functions to test from functional specification)

b. Functions Not to Be Tested

(e.g., out of scope)

d. Unit Test Cases

(i.e., specific unit test cases)

c. Integration Test Cases

(i.e., specific integration test cases)

e. System Test Cases

(i.e., specific system test cases)

V. Test Procedures

a. Test Case, Script, Data Development

(e.g., procedures to develop and maintain)

b. Test Execution

(i.e., procedures to execute the tests)

d. Correction

(i.e., procedures to correct discovered defects)

c. Version Control

(i.e., procedures to control software component versions)

e. Maintaining Test Libraries

d. Automated Test Tool Usage

(i.e., standards)

e. Project Management

(i.e., issue and defect management)

f. Monitoring and Status Reporting

(i.e., interim vs. summary reports)

VI. Test Tools

a. Tools to Use

(i.e., specific tools and features)

b. Installation and Setup

(i.e., instructions)

c. Support and Help

(e.g., vendor help line)

VII. Personnel Resources

a. Required Skills

(i.e., manual/automated testing skills)

b. Roles and Responsibilities

(i.e., who does what when)

c. Numbers and Time Required

(e.g., resource balancing)

d. Training Needs

(e.g., send staff to tool training)

VIII. Test Schedule

a. Development of Test Plan

(e.g., start and end dates)

b. Design of Test Cases

(e.g., start and end dates by test type)

c. Development of Test Cases

(e.g., start and end dates by test type)

d. Execution of Test Cases

(e.g., start and end date by test type)

c. Reporting of Problems

(e.g., start and end dates)

d. Developing Test Summary Report

(e.g., start and end dates)

e. Documenting Test Summary Report

(e.g., start and end dates)

<Company Logo>

System Summary Report

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Table of Contents

Technical Design Review Checklist

Project Name		Project Code		Author	

Checklist Purpose	Review the technical design for clarity and completeness. For all negative responses the QA Project Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email, depending upon the severity.

Context	Task			
		Yes	No	Comments
Technical Design				
	Is the logic sequencing erroneous?			
	Is the processing inaccurate?			
	Do procedure handle input or output parameters incorrectly?			
	Do procedures not accept all data within allowable ranges?			
	Are limit and validity checks made on input data?			
	Are there recovery procedures not implemented or are inadequate?			
	Is required logic missing or inadequate?			
	Are values erroneous or ambiguous?			
	Is data storage erroneous or inadequate?			
	Is variable missing or not declared properly?			
	Is the database not compatible with the data environment?			
	Does the modular structure reflect a high inter-modular dependence?			
	Are there algorithms not evaluated for accuracy or speed?			
	Is the control structure not expandable?			

Context	Task			
		Yes	No	Comments
	Do control structures ignore the processing priorities?			
	Are the interface protocols incorrectly used?			
	Is data not converted according to the correct format?			
	Is there no consideration to round off or truncation?			
	Are the indices used incorrectly?			
	Are there infinite loops?			
	Are database rules violated?			
	Are there special cases not covered?			
	Is error handling deficient?			
	Are timing considerations neglected?			
	Are interface specifications misunderstood or implemented wrongly?			
	Are the functional specifications misallocated among the various software modules?			
	Is the system functionality correct but does not meet performance requirements?			
	Is the system not sufficiently complex to match the problem being solved?			
	Are there actions in response to given inputs inappropriate or missing?			
	Do algorithmic approximations provide insufficient accuracy or erroneous results for certain values of the input?			
	Are there errors in the detailed logic developed to solve a particular problem?			
	Do singular or critical input values yield unexpected results that are not appropriately accounted for in the code?			
	Are there algorithms that do not cover all the necessary			

Context	Task			
		Yes	No	Comments
	cases?			
	Are there algorithms that are incorrect or produce the wrong solution?			

<Company Logo>

Test Approvals

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Deliverable Approvals			
Test Deliverable Name	**Approval Status**	**Approver**	**Approved Date**

<Company Logo>

Test Case Log

Test Name:
Test Case Author:
Tester Name:
Project Name:
Project ID:
Test Cycle Number:
Date Tested:

Test Case ID	Condition	Expected Results	Actual Results	Requirements ID
Enter ID	Enter Specific Test Condition	Describe the Specific Results Expected Upon Executing the Condition	Record "Pass" or "Fail"	Enter the ID that Traces Back to the Specific Requirement

Test Case Preparation Review Checklist

Project Name		Project Code		Author	

Checklist Purpose	This Checklist ensures that test cases have been prepared as per specifications. For all negative responses the Test Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email,.

Context	Activity	Status		
		Yes	No	Comments
	Is the Approved Test Plan is available?			
	Are the Resources identified to implement the test Plan?			
	Are the Base Line Documents available?			
	Is the Domain Knowledge imparted to Team members to work with the application?			
	Has the Test condition document completed?			
	Have Test cases have been developed for all the Requirements?			
	Has the traceability been verified?			
	Have all the Basic flows in use cases have been covered?			
	Have all the alternate flows in use cases been covered?			
	Have any changed requirements have been covered fully?			
	Have non-testable requirements has been escalated?			
	Have the test cases been written for data flow across interfaces?			
	Have the test cases been written for all types of tests defined in the Project Plan?			
	Have all the positive and negative cases has been identified?			
	Are all boundary cases identified?			
	Have test cases been written for non-functional requirements?			
	Have test cases been written for GUI /hyperlink testing in Web applications?			
	Have test cases been written to test date integrity?			

<Company Logo>

Test Case Template

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Date: _____ Tested by: _____
System: _____ Environment: _____
Objective: _____ Test ID _____ Req ID _____
Function: _____ Screen: _____
Version: _____ Test Type: _____
(Unit, Integ., System, Accept.)

Condition to Test:

_____
_____
_____
_____

Data/Steps to Perform:

_____
_____
_____
_____

Expected Results:

_____
_____
_____

Actual Results: Passed h Failed h

_____
_____
_____
_____

<Company Logo>

Test Condition /
Test Case Matrix

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Test Condition / Test Case Matrix

Item No.	Requirement Details/ Source (Fun Spec / Bus Req / Other)	Condition No.	Test Condition	Test case No.

<Company Logo>

Test Defect Details Report

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Test Defect Details Report

Item No.	Defect ID	Script ID	Test Case Description	Expected Result	Actual Results	Detected By	Defect Status	Severity	Priority	Reported Date	Closed date

Company Logo>

Test Defect Report

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Project Name:

Defect Report

Defect ID	Date	Test Script ID	Test Case Description	Expected Result	Actual Result	Status	Severity	Defect Type	Tester Comment	Developer Comment	Client Comment

<Company Logo>

Test Execution Plan

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Test Execution Plan

Project Name:
Project Code:
Date:

Activity No.	Activities / Sub-Tasks	Planned Date		Resource	Total Test Cases/Scripts	Test Cases/ Scripts Completed	Comments
		Start Date	End Date				

<Company Logo>

Project Milestones

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Test Project Milestones

Milestone	Date Due	Actual Due

<Company Logo>

Test Schedule

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Test Schedule

Test Step	Start Date	End Date	Responsible
Information Gathering			
Test Planning			
Test Case Design			
Test Development			
Test Execution/ Evaluation			
Prepare for the Next Test Iteration			
Conduct System Testing			
Conduct Acceptance Testing			
Summarize Tests/ Project Closure			

<Company Logo>

Test Strategy

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Table of Contents

1 Introduction

1.1 Project Overview
< An introduction to the project including an outline of the project scope >

1.2 About the Client
< Client's business in association with the project>

1.3 Application / System overview
< A concise and high-level explanation of our understanding of the functionality of the application and the break-up of business functions >

2 Testing Scope
<General application scope should be provided in this section. >

2.1 Testing Objectives
< Test objectives as they relate to specific requirements>

2.2 Types of Tests
< Types of testing such as functionality testing, Non-functionality testing, operational acceptance testing, regression testing, performance testing etc should be mentioned here >

2.3 Within Scope
< Transactions, reports, interfaces, business functions etc.>

2.4 Out of Scope
< Define what is NOT specifically covered in testing>

2.5 Assumptions
< Test assumptions in conjunction with the test scope> >

2.6 Baseline documents
< The list of Baseline documents, prototype with version numbers>

3 Testing Approach

3.1 Testing methodology

3.1.1 Entry criteria

< List of criteria that need to be fulfilled before test planning should begin >

3.1.2 Test Planning Approach

< The approach to be adopted in preparing necessary test wares, ex. manual test cases or automated test scripts, approach for creating test data etc.>

3.1.3 Test documents

< List of test documents, their definition and purpose >

3.1.4 Test Execution methodology

<A description of how the tests will be executed?

3.1.5 Test Execution Checklist

< List of items that need to be available with the test team prior to the start of test execution >

3.1.6 Test iterations

< Number of iterations of testing planned for execution, the entry and exit criteria and the scope of each test iteration>

3.1.7 Defect Management

< Entire defect management process. It includes defect meeting, defect resolution, etc.>

3.1.8 Defect Logging & Defect Reporting

<A note on defect logging process and a sample defect log template that will be used during test execution should be mentioned here.>

3.1.9 Defect Classification & Defect Life cycle

<A detailed note on the life cycle of a defect, the different defect severity levels, defect categories>

3.1.10 Defect Meetings

< A detailed defect meeting procedure indicating the parties to the defect meeting, their responsibility and the frequency of defect meeting>

3.1.11 Exit criteria

<Exit criteria for test execution>

4 Resources

4.1 Skills required for the project

<An analysis of the skills required for executing the project>

4.2 Training Schedule

<Project specific training needs with a timetable>

4.3 Offshore

4.3.1 Test personnel

<List of test team personnel and their role in the project along with date of inclusion in the project>

4.3.2 Roles & Responsibilities

4.4 Onsite

4.4.1 Test personnel

<List of test team personnel and their role in the project along with date of inclusion in the project>

4.4.2 Roles & Responsibilities

4.5 Client

<Roles and responsibilities of client or client's representative>

4.5 Test infrastructure

4.6.1 Hardware

<List of hardware requirements for test execution>

4.6.2 Software

<List of software requirements for test execution>

5 Project Organization and Communication

<Project organization chart, the turn around time for the review and sign off for the documents submitted to the clients>

5.1 Escalation model

<In case of issues and concerns, the escalation procedure and time lines to escalate >

5.2 Suspension and Resumption criteria

<List of circumstances under which test activities will be suspended or resumed should be mentioned here>

5.3 Risk, Contingency and Mitigation Plan

<Risks of the project, contingency and mitigation plan for the risks identified>

5.4 Schedule

5.4.1 Milestones

<A high-level schedule for the different stages of the project with clear indication of milestones planned with a list of activities>

5.4.2 Detailed-Plan

<A detailed project plan using MS-Project with all identified tasks and subtasks, resources to be used with dates fitting into the milestones as mentioned in the high-level schedule>

5.4.3 Deliverables

<A list of deliverables associated with the project as mentioned in the test documents, the mechanism for obtaining client acceptance for the deliverables>

6 Appendix

<Appendix, as mentioned in any of the sections above should be mentioned here>

<Company Logo>

Traceability Matrix

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Traceability Matrix

Item No.	Bus Rule Ref	Fun Spec Ref	Function Description	Test Condition ID	Test Case ID	Defect ID	Comments

<Company Logo>

Unit Test Plan

Rough Draft (Version 1.0)

Prepared by:

Date:

Revision History

Date	Issue	Description	Author

Table of Contents

1. **Introduction Section**
    a. ***Test Strategy and Approach***
    b. ***Test Scope***
    c. ***Test Assumptions***
2. **Walkthrough (Static testing)**
    a. ***Defects Discovered and Corrected***
    b. ***Improvement Ideas***
    c. ***Structured Programming Compliance***
    d. ***Language Standards***
    e. ***Development Documentation Standards***
3. **Test Cases (Dynamic testing)**
    a. ***Input Test Data***
    b. ***Initial Conditions***
    c. ***Expected Results***
    e. ***Test Log Status***
4. **Environment Requirements**
    a. ***Test Strategy and Approach***
    b. ***Platform***
    c. ***Libraries***
    d. ***Tools***
    e. ***Test Procedures***
    f. ***Status Reporting***

Unit Testing Checklist

Use Case/Functional Spec Name: _____

Use Case/Functional Spec Number:_____

Expected Testing Actions	Completed			Comments/Explanation
	Yes	**No**	**N/A**	
Was every field verified to allow only data of the correct format to be entered [ex. numeric (signed/unsigned), alphabetic, alphanumeric (special characters), date, valid & invalid]; Check error messages for invalid data?				
Was every field verified to allow only data of allowable values to be entered (ex. tables, ranges, minimum, maximum); Check error messages for invalid data?				
Was every field verified that business rules for the field were enforced (ex. mandatory/not mandatory when another field is present, relational edits);?				
Was every error message tested?				
Was every field verified to handle all invalid values?				
Were all upper and lower case field conditions verified?				
Were all internal tables were verified or addressed to have sufficient capacity to provide for maximum volumes (ex. dataset population, # of transactions to accept); Check error messages?				
For numerical fields, were all zero values have been tested?				
Were all valid data conditions tested based on data dictionary definitions?				
Were the specifications reviewed to ensure are conditions have been tested?				
Were all alpha fields are validated for “blank” conditions?				
Was it verified that all data is being retrieved from and written to the correct physical database?				
Were all fields initialized properly?				
Were all fields that are protected validated?				
Was all data being retrieved from and written to appropriate files and fields verified?				
Was every calculation verified to provide correct results over the entire ranges of involved data items based of the business rules?				
Was every output value and its format verified? (ex. Rounding/Truncation).				
Was data passed to all other systems verified to be in the correct format by the receiving system?				
Was data passed from other systems verified to be in the correct format?				

Expected Testing Actions	Completed			Comments/Explanation
	Yes	**No**	**N/A**	
Were all required security requirements, as specified in the design specification, verified?				
Were all outputs verified to identify the security level classification appropriate to the information being present?				
Were all error conditions are trapped and handled according to the standards for the environment(s) in which the software item will execute (ex. error codes, error messages)?				
Was it verified that the software items do not leave corrupted data when unexpected error conditions occur (ex. General Protection Fault, Syntax Error, abnormal exit)?				
Were all messages verified to be clear and understandable by typical end users of the software item?				
Did typical users of the instructions verify all the instructions to be concise and understandable?				
Did the typical audience of the documentation verify the documentation to be clear and understandable?				
Were all tabs, buttons, hyperlinks, field tabbing operates in a logical manner according to the REL IT standards in which the software item will execute?				
Were all commands verified to be available using either a mouse, keyboard?				
Were tests performed to indicate that response times meet requirements as specified in requirements and will be acceptable in the environment(s) in which the software item will execute (run-time for large volumes)?				
Was the ode reviewed?				
Was all undefined loop iterations verified?				
Were all the programming standards are satisfied?				
Were invalid codes verified?				
Were invalid data relationships verified?				
Were invalid date formats verified?				
Has the Client signed off on the Test Strategy?				
Was it verified that the software items meet all standards applicable to the environment(s) in which the software item is expected to execute.?				
Was it verified that the software items meet all requirements imposed by corporate standards regarding financial controls and privacy.				
Was it verified that the software could be adapted to execute in the specific environment(s) in which it is required to execute?				

Comments:	
Completed by: **_____ Date_____** **Developer**	**Accepted by:** **_____ Date_____** **Development Manager**

Developer Unit Testing Guidelines

Use Case/Functional Spec Name: _____

Use Case/Functional Spec Number:_____

Expected Testing Actions	Completed			Comments/Explanation
	Yes	**No**	**N/A**	
Every field was verified to allow only data of the correct format to be entered [i.e. numeric(signed/unsigned), alphabetic, alphanumeric(special characters), date, valid & invalid]; Check error messages for invalid data.				
Every field was verified to allow only data of allowable values to be entered (i.e. tables, ranges, minimum, maximum); Check error messages for invalid data.				
Every field was verified that business rules for the field were enforced (mandatory/not mandatory when another field is present, relational edits); Test for each error message.				
Every field is validated to handle all invalid values.				
Verify all upper and lower case field conditions.				
All internal tables were verified or addressed to have sufficient capacity to provide for maximum volumes (i.e. dataset population, # of transactions to accept); Check error messages. This is a form of load testing that maybe accomplished at this level of testing.				
For numerical fields, all zero values have been tested.				
All valid data conditions are tested based on data dictionary definitions.				
Specifications have been reviewed to ensure are conditions have been tested.				
All alphs fields are validated for “blank” conditions.				
It was verified that all data is being retrieved from and written to the correct physical database.				
All fields are initialized properly.				
All fields that are protected are validated.				
It was verified that all data is being retrieved from and written to appropriate files and fields.				
Every calculation was verified to provide correct results over the entire ranges of involved data items based on the business rules provided.				
Every output value and its format were verified (ie:Rounding/Truncation).				
Data passed to all other systems was verified to be in the correct format by the receiving system.				
Data passed from other systems was verified to be in the correct format				

Expected Testing Actions	Completed			Comments/Explanation
	Yes	**No**	**N/A**	
It was verified that the required security, as specified in the design specification, is provided.				
All outputs were verified to identify the security level classification appropriate to the information being present.				
It was verified that all error conditions are trapped and handled according to the standards for the environment(s) in which the software item will execute (i.e. error codes, error messages).				
It was verified that the software item does not leave corrupted data when unexpected error conditions occur (i.e. General Protection Fault, Syntax Error, abnormal exit).				
Messages were verified to be clear and understandable by typical end users of the software item.				
All instructions were verified to be concise and understandable by typical users of the instructions.				
Documentation was verified to be clear and understandable by the typical audience of the documentation.				
It was verified that all tabs, buttons, hyperlinks, field tabbing operates in a logical manner according to the REL IT standards in which the software item will execute.				
All commands have been verified to be available using either a mouse, keyboard.				
Tests were performed to indicate that response times meet requirements as specified in requirements and will be acceptable in the environment(s) in which the software item will execute (run-time for large volumes).				
Verify code has been reviewed.				
Validate no undefined loop iterations.				
Validate all programming standards are satisfied.				
Validate invalid codes are identified.				
Validate invalid data relationships are not used.				
Validate invalid date formats are not used.				
Validate page overflows are properly handled.				
It was verified that this software item meets all standards applicable to the environment(s) in which the software item is expected to execute.				
It was verified that the software item meets all requirements imposed by corporate standards regarding financial controls and privacy.				

Expected Testing Actions	Completed			Comments/Explanation
	Yes	**No**	**N/A**	
Comments:				

Completed by: **_____ Date_____** **Developer**	**Accepted by:** **_____ Date_____** **Development Manager**

Developer Unit Testing Guidelines Field Explanations

Assumption: Unit testing is done in support of specific use case or functional spec.

- Use case/functional spec name: List the name
- Use case/functional spec number: List the number
- Yes, NO, N/A: Developer checks the appropriate box for the test. Either the test was executed, not executed but could have or not applicable.
- Comments/Explanation: Developer documents any information about the testing which would be useful in future tests. Of course, all tests should be executed if warranted.
- Completed by: Developer signs and dates guideline sheet and forwards to Manager.
- Accepted by: Development Manager signs and dates guideline sheet and then passes to QA during turnover.

<Company Logo>

Use Case Review Checklist

Project Name		**Project Code**		**Author**	

Checklist Purpose	Review use cases for clarity and completeness. For all negative responses the QA Project Manager will assess the impact and escalate as an issue to the concerned parties for resolution. This can be accomplished through weekly status reports or Email, depending upon the severity.

Context	**Task**			
		Yes	**No**	**Comments**
Use Cases				
	Are the end-user actions identified?			
	Is there enough detail identified so that contribution of information system output items can be related to those actions?			
	Is all the information used in taking an action identified and related to the action?			
	Is the output from the system user test related to specific actions?			
	Does the end user correctly understand the output reports and screens?			
	Does the end-user understand the type of logic and computation performed to produce the output?			
	Can the end-user identify the contribution the output makes to the actions taken?			
	Can the end-user identify whether the actions taken are correct?			
	Is the relationship between the system output and business actions defined?			
	Have the actors been			

Context	Task			
		Yes	No	Comments
	identified correctly?			
	Is each use case step written in a simple declarative statement?			
	Are the sequence of steps correct documented?			
	Are concurrent steps defined?			
	Are the steps written from the user's point of view (or programming)?			
	Have system boundaries defined, ex. What is in the system and what is outside?			
	Has time been described adequately?			
	Have pre and post conditions been defined?			
	Have the flow of events been documented correctly?			
	Have complex use cases handled correctly?			
	Have alternative paths been identified?			
	Have includes been defined correctly to handle common behavior?			
	Have extends been defined to handle the extension of existing use case behavior?			
	Has inheritance been addressed?			
	Have all the interfaces been defined?			
	Has traceability between use cases been defined?			