

State of Oregon
Information Resources
Management Division

Software Development Lifecycle
Methodology

Table of Contents

INTRODUCTION	3
SDLC Goals	3
SDLC Benefits	3
SDLC Applicability	3
Life Cycle Stages	4
Initiation and Definition Stage	4
Planning Stage.....	4
Execution and Control Stage	5
Closure Stage	5
INITIATION AND DEFINITION STAGE	6
Function	6
Process	6
Tasks.....	8
Deliverables	8
Roles	8
PLANNING STAGE	9
Function	9
Process	9
Tasks.....	12
Deliverables	12
Roles	12
EXECUTION AND CONTROL STAGE	13
Function	13
Process	13
Tasks.....	15
Deliverables	16
Roles	16
CLOSURE STAGE.....	17
Function	17
Process	17
Tasks.....	17
Deliverables	17
Roles	18
Appendix A: Inspections, Walkthroughs & Peer Reviews.....	19
Appendix G: Roles and Responsibilities.....	24
NEEDS ASSESSMENT TEAM.....	26
IMPLEMENTATION TEAM	27
Appendix H: Glossary	28

INTRODUCTION

An Software Development Lifecycle Methodology (SDLC) is a generic, tailorable, scalable process guiding the high level stages of computer applications development. The process begins when the idea for a new application is first conceived, and ends when the application is taken out of service. The SDLC provides a common framework of life cycle stages for discussion about the problem to be solved. It structures the development process and gives the project team a road map to follow. It provides a "preflight" checklist to assure that all the needed elements are addressed.

The SDLC establishes required documents and other deliverables, and seeks to establish standardized methodologies. The SDLC provides a detailed description of roles and responsibilities with respect to the development process.

SDLC Goals

The overall goal is to improve the quality and efficiency of software applications development in the Information Resources Management Division (IRMD). To help do so the SDLC facilitates communication among developers and customers and improves documentation of software activities.

SDLC Benefits

This SDLC process should benefit the agency in a number of ways:

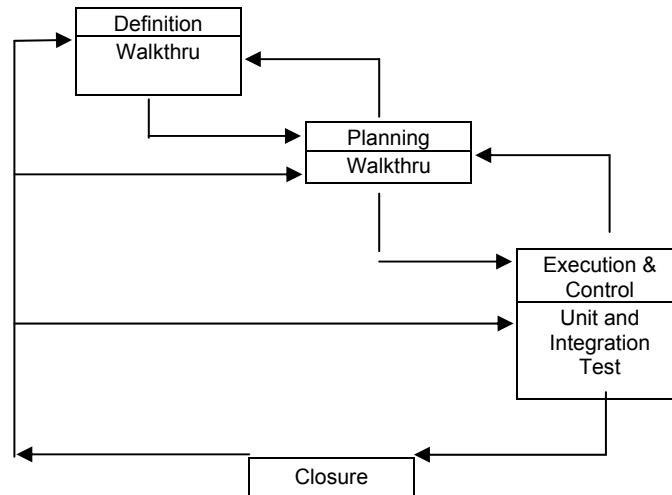
- Because customers and developers commit to the requirements and the development plan, the probability that the final product will meet the customer's needs is greatly increased.
- It provides a common framework for software development which results in consistency among applications.
- Duplication of development effort is greatly reduced. Functionality and data structures that are common to other applications are discovered and used when appropriate to eliminate redundancy.
- Application developers are guided in their work by one common document. The progression through each phase allows for easier tracking of the development process.
- A defined, documented process tends to mitigate the effect of personnel turnover.
- Use of the SDLC makes application maintenance easier. Necessary modifications are much easier to complete with system documentation available.
- Improved methods, techniques, and technologies can be captured in the SDLC and become standards for IRMD or other agencies to follow.

SDLC Applicability

Projects vary in size and complexity from a simple single customer program that produces a report to a department-wide requisition tracking system. The SDLC will be applied with common sense and flexibility to ensure that the benefits gained are proportional to the time and effort required by using it. Indeed, relatively minor maintenance level activities do not require the full-blown use of the SDLC process.

Life Cycle Stages

The SDLC describes specific stages for the entire life cycle of IRMD information systems. These fundamental stages are described next.



The stages are iterative: for example, changes in the scope or requirements during the Execution & Control Stage will necessitate repetition of previous stages. Whenever the process returns to previous stages, the project continues through each subsequent stage until reaching the end of the life cycle. A project may cross over into multiple stages simultaneously.

Initiation and Definition Stage

This is an administrative step to provide management with information for decision making. The purpose of the Initiation and Definition stage is to initiate a project and establish its priority relative to other projects. The Project Charter describes the project, identifies affected areas and personnel.

The customer requirements, functions, and business rules are determined and documented during the Initiation and Definition stage. Various techniques are available: customer interviews; Joint Application Development (JAD) workshops; prototyping, etc.

Planning Stage

The project plan developed during this stage lays out how the project will be executed. Effort estimates and a project schedule are included. The Project Plan is reviewed by IRMD senior management and the State CIO.

The development team uses the requirements specifications from the Needs Assessment Stage to define the physical architecture of the system. Prototyping is recommended. A data conversion/archiving plan is produced during this stage if applicable. The design is validated by a walkthrough.

Execution and Control Stage

The application is developed and unit tested during the Development stage. The database is created and the data dictionary is completed. Tests of the data transition from existing systems are performed. A demonstration of the user interface must be provided to customers and the review team for evaluation.

A test system is installed and evaluated during the Testing stage. Latent coding errors are resolved at this stage. Any other changes trigger a return to the appropriate stage. In fact, some reiteration through the stages is a common practice in even successful projects. The stage is complete when the development and testing teams agree that the application satisfies all defined requirements.

The application is deployed into a production environment. That is, the production system is installed and training is conducted at this time. Customer sign-off occurs here.

Closure Stage

Final administrative procedures for the ongoing maintenance of the application are set up and documented. Application is supported with corrective changes (bug fixes) and enhancements (new requirements).

The sections to follow in this document provide guidance for each of the stages of the SDLC life-cycle. For each stage function, process, tasks, deliverables, roles, and forms will be described. In some stages the validation tasks are identified.

INITIATION AND DEFINITION STAGE

Function

This is an administrative step to provide management with needed information for decision making. The purpose of the Project Charter stage is to initiate an information systems development project and establish its priority relative to other projects.

In this stage a Needs Assessment is performed to document a common understanding between customer and developer in the form of a requirements document. Here we determine the customer requirements, functions and business rules. Various techniques are available: customer interviews; Joint Application Development (JAD) workshops; prototyping, etc.

Process

The initiator of the project writes the Project Charter, possibly with the assistance of other division and/or applications development staff. The Project Charter will include a general description of the project, any background information, the proposed technical approach and an estimate of how long it will take to complete the Needs Assessment and Planning Stages. Identification of the project manager, customer contact and resources who will be involved in the Needs Assessment Stage and Planning Stage are to be made at this time.

The project manager oversees the completion of the Project. When there is a cost that will be incurred by the customer to complete the Needs Assessment and Planning Stages, an attachment must be placed with the Project Charter. The attachment will provide an estimation of cost to the customer for each of the two stages. The project manager will then contact the Sponsor or functional business owner and setup the billing agreement to be used for the Needs Assessment and Planning Stages. The project manager will complete the Service Level Agreement (SLA) or billing agreement for new development or amend the existing SLA or billing agreement with a Service Level Estimate for the maintenance to an existing system.

The project manager will route the completed Project Charter form to the SDAC Manager, the GGDC Manager, the customer contact and the functional business owner.

The project manager will present the document to the State CIO for review if it is an enterprise application project or to the Systems Development Manager for single agency application efforts.

The project manager will create an electronic project timeline that identifies the steps and tasks to produce the necessary deliverables and complete the Needs Assessment and Planning Stages. Each task must contain the estimated duration and assigned resources. After creating the initial timeline, the project manager will compare the time estimates to the time estimates from the Project Charter and make any necessary adjustments. If applications development staff will develop the application, but the project's priority and/or available resources do not allow scheduling of the project, the proposal will be filed with other approved, unscheduled projects.

Needs Assessment Process

The Technical Lead or analyst conducts a needs assessment. The project manager should identify those components of the Requirements document that are necessary for the project. The goal in this stage is to develop a complete and unambiguous description of prioritized needs that the application must address. The project manager/technical lead and customer contact finalizes the proposed list of needs assessment members and begin to schedule needs assessment meetings. The analyst should review any existing documentation about how things are done, including data used and produced. The customer contact should provide samples of data used (input) or produced (output) to the analyst. Examples of data input sources include paper forms or existing databases. Examples of data output may include reports, queries and database updates.

Data elements, functions, existing resources and applications, database requirements, performance requirements, external interface requirements, security requirements, audit requirements and archival requirements are gathered from the needs assessment team and documented in the Requirements document by the analyst. Initial estimates regarding the frequency of updates and amount of data are also determined. The technical lead inventories the hardware and software resources of the customer community.

Using the information in the Requirements document, the technical lead should determine if there are areas of overlap with other applications. In these cases, integration analysis should be conducted during the system design stage. Any billing requirements the application may have if it will be a sellable service (e.g., unit of measure (month, day, hit, session), cost per unit, etc.) are recorded in the Requirements document.

Any one of several techniques may be used to obtain the information in the Requirements document. Joint Application Development (JAD) Workshops; meetings with the customer; and/or rapid prototyping. Rapid prototyping is a suggested technique for ironing out graphical user interface specifications, proof-of-concept issues, and otherwise providing a flow of communication between and among developers and customers. Prototyping the user interface has been shown to be very productive during the early stages of requirements analysis. However, whether or not prototyping will be used in the Needs Assessment stage is a project unique decision. The SDLC does not mandate its use during this stage.

A requirements specification inspection is conducted to ensure all information needed for design and for compliance with standards has been captured. The inspection participants should decide if the changes are significant enough to warrant a re-inspection, and if so, the process should be repeated.

When there are changes to the scope of the project or content of any document a project change control revision log will be created and reflect the changes made. If there is an existing revision log for the document it will be updated to reflect the current change. The project change control revision log will be reviewed by the project manager prior to continuing to the next stage to determine the impact on the project plan.

At this point, coordination/communication with applicable outside data sources or applications sources should be established.

Tasks

- Complete the Project Charter
- Route Project Charter to State CIO or Systems Development Manager for review
- Obtain Commitment from other involved divisions GGDC/DOIT/eGov
- Send copy of Project Charter to customer and obtain approvals
- Obtain signatures on billing agreement
- Complete the Service Level Agreement (SLA)
 - Data on projected cost of the Needs Assessment and Planning stages should be included
 - After the Planning stage, the SLA will be updated with projected cost of remaining stages
- Route Billing Agreement to GGDC to obtain agreement number
- Add Billing Agreement number to Systems Development Billing System
- Develop Schedule for the Needs Assessment and Planning Stages
- Determine if similar applications exist and determine feasibility of integration
- Complete the Requirements Document
- Draft list of customer deliverables
- Begin planning with owners and stewards of outside data sources or applications
- Conduct Inspection of Requirements Document

Deliverables

[Project Charter](#)

- Schedule for Needs Assessment and Planning Stages
- [Project Change Control Revision Log](#)

[Estimate Cost Summary Worksheet](#)

[Billing Agreement](#)

[Requirements Document](#)

[Project Status Report](#)

Roles

Project Sponsor/ Customer Contact
Project Manager
Technical Lead
Developer(s)
Review Team
Data Administrator

PLANNING STAGE

Function

The purpose of this stage is to produce a Project Plan to determine how the project will be conducted. The Project Plan establishes effort estimates and a schedule for the project. In addition, the Plan should:

- Finalize team membership
- Determine roles and percent of time devoted to the project
- Identify the technical approaches which will be used to produce the application

The Project Manager works with the Systems Development Technical lead and customer contact to assign team members and to dedicate resources

During this stage, the development team uses the documented requirements from the Needs Assessment Stage to define the physical implementation of the system. A data conversion/archiving plan is produced during this stage if applicable. After a successful walkthrough of the design document, the project moves to the Development stage.

Software design is a process through which requirements are translated into a representation of software. It can also be thought of as a problem-solving process whereby the software designer applies techniques and principles to produce a logical model that describes and defines a solution to a problem. From a project management standpoint, software design is conducted in two steps. *Conceptual design* is a high level logical representation of what the application should do. *Detail design* is a low level representation and specific as to what the application should do.

The design stage is the period of time in the software life cycle during which the designs for architecture, software components, interfaces, and data are created, documented, and verified to satisfy requirements.

Prototyping is a highly recommended technique for ironing out design issues, user interface specifications, and otherwise providing a flow of communication between and among developers and customers. However, whether or not prototyping will be used in the design stage is a project unique decision. The SDLC does not mandate its use during this stage.

Process

The Project Manager in conjunction with the technical lead is responsible for leading the project through the SDLC process.

The project manager is responsible for keeping all project documentation up-to-date. This may be accomplished by filing all project documents, including the Project Charter, in an electronic project file. A folder should be created on the common drive under a separate development project folder. The Project Manager and/or Technical lead should put copies of all documents in this folder. Forms with signatures should be kept by the Project Manager.

Any additional customer deliverables should be identified at this time and listed in the Project Plan Template.

Development Team

The first step should be to finalize the selection of the development team members. The Proposed Scope section of the Project Charter should guide the project manager and technical lead in determining the areas of IRMD, and outside IRMD (if applicable), from which team members should be selected. Once the development team has been selected, they will work together to develop the Project Plan. At the beginning of the planning stage, development area staff will also be assigned to the review team.

Review Team

The appropriate functional areas will be represented on the review team:

Data Base Administrator
Senior Software Engineer
Systems Administrator
Quality Assurance

Test Team

The initial membership of the Test Team should be established. The appropriate functional areas will be represented on the test team:

Customer Contact
Developer
Data Administrator
Database Administration
Quality Assurance Analyst

Project Plan

One or more meetings of the project manager and technical lead should be held to gather the information necessary to estimate effort for each stage and set up the project schedule. The project manager must determine the particular approaches and techniques that will be used to complete the project. It is at this step that the project manager customizes the project from the overarching, generic SDLC framework to a framework more appropriate for the scope of the project. Using the Project Plan, project specifics are analyzed and documented. Topics addressed could be: If any of the development team members are located outside of Salem and it becomes necessary to bring the whole team together, how will it be accomplished? How will customer sign-off of deliverables be accomplished? Will rapid prototyping be used during Design and/or Development to facilitate communication between the analysts and the customers? And so on.

Estimates of additional hardware, system software or utilities and network hardware that may be needed to support the test or production environments. These projections are produced jointly by cooperation of the project managers and the managers of the appropriate support units, GGDC, ENS or Systems Development and Consulting.

The project manager creates an estimated project schedule in electronic format that is accessible by all development team members. The project schedule must include at a minimum, the planned start and end dates for each stage. The project schedule should also include the resources needed if known at the time (which programmers are assigned to which major tasks or modules). The project manager updates the project schedule as major milestones are reached and keep the schedule up to date.

After the Project Plan has been finalized, the plan should be sent to IRMD systems managers to obtain the necessary sign offs. Once all the signatures are obtained, the project plan should be routed to the State CIO and/or Systems Development Manager and presented to the Project Sponsor and stakeholders.

The high level design (high-level structure) is completed, rough draft user interfaces are created (where applicable) and identified in the Design Document. Upon completion of the conceptual design and prior to moving to detailed design, a walkthrough of the conceptual design will be conducted.

During the detail design phase, functional and technical specifications are written according to the requirements and design approach outlined in the Conceptual Design. Once these are completed, unit test plans are developed and copies of the functional specifications are given to the customer representative so acceptance test plans can be created. An inspection of the design documents is also conducted.

The Technical Lead should determine if other applications exist or are being developed which have similar functionality. If such similarities exist, analysis should be conducted to determine the feasibility of integration during this project. Every effort should be made to integrate the functions when possible.

The technical lead transforms the conceptual data model created during Needs Assessment into a *physical database diagram* that represents the physical tables which will be accessed by the application. In order to decide how to make the logical to physical transformation, the technical lead takes into consideration factors including which functions will be automated and how the data will be accessed, and the developer makes the necessary changes to optimize the performance of the application. Database Administration (DBA) staff frequently act as consultants during the physical database design.

The technical lead will determine if any of the required tables already exist, or if similar tables exist which could, with minor modifications, meet the needs of the application. If existing tables will require modification to suit the needs of this application, the technical lead should request the changes in writing from the personnel who are currently responsible for the tables.

The companion to the physical database diagram is the *data dictionary*. It is created using the physical database diagram; the detailed functional requirements; the logical conceptual data model and its associated data elements; and the volume and frequency information. The data dictionary follows IRMD standards for defining the physical equivalent of the data elements. Any specifications and algorithms for derived data elements are included. The draft physical database diagram and data dictionary becomes part of the project design documentation and is included in the design document.

When there are changes to the content of any document, a project revision log will be created and reflect the changes made to the document's content. If there is an existing revision log for the document it will be updated to reflect the current change. The project revision log will be reviewed by the project manager prior to continuing to the next stage.

Tasks

- Determine any additional customer deliverables and update the list (note: changes to functional requirements will require going back to Needs Assessment Stage)
- Develop the Project Plan
- Develop Project Schedule
- Present Project Plan and Schedule to Project Sponsor and stakeholders
- Develop Conceptual design
- Conduct walkthrough of Conceptual design
- Present Conceptual Design to Project Sponsor and stakeholders
- Develop Detail Design
- Develop Interface and Data Conversion Strategies
- Design Logical Data structures
- Determine hardware, software, network requirements
- Develop hardware/software/network installation or upgrade plan
- Combine the Conceptual and the detailed designs into the Design Document
- Refine Estimates
- Initiate Test Plan (unit, system, and acceptance)
- Conduct an inspection of the Design Document
- Update Project Schedule

Deliverables

[Project Plan](#)

- [Update Project Schedule](#)

[Logical Data Model](#)

[Conceptual Design Document](#)

[Detail Design Specifications](#)

Hardware, Software & Network Projections and Requirements

[Estimate Cost Summary Worksheet](#)

[Project Status Report](#)

[Project Change Control Revision Log](#)

Roles

Sponsor

Stakeholders/Customer Contact

Project Manager

Technical Lead

System and Network Administrators

Review Team

Developer(s)

Database Administrator

EXECUTION AND CONTROL STAGE

Function

During this stage, the application is constructed, unit tested, integrated, and modified during the development stage. The physical data structures are created. Interfaces are developed and data conversion processes from existing systems are tested. A demonstration of the user interface, often in the form of a software prototype, must be provided to customers and the review team for evaluation.

The deliverables produced in this stage relate to the physical nature of the application. These may include major integrated code modules as well as documentation (emails, memo, signoffs) to show that a reasonable level of developer – customer review has been achieved. It is vital that the customer be involved in this stage as the user interface and application logic are taking shape.

During this stage, the application is installed on a test platform and integration level testing is conducted. This stage always includes complete functional testing to ensure all requirements are satisfied by comparing the requirements documented with the actual functionality of the application. This may also include non-functional testing to ensure issues such as stress, recovery, performance, human interface, and reliability are satisfied.

During this stage production installation, acceptance testing, and customer sign-off is obtained on the application. All documents are finalized for delivery to the customer.

Process

During this stage, the Technical Lead, system designer and developer work together to create the physical data structures and modules which comprise the application. Finalization of the Data Dictionary should be accomplished and identification of existing code modules appropriate for reuse should be identified and implemented as applicable.

The Technical Lead or system designer may need to contact the appropriate DBA to gain the required access to the applicable development platform. The data structures are then created on this platform. The DBA reviews the application for security, efficiency and integrity. This review consists of automated tests on the data structures as well as a final analysis of the design deliverables. Depending on the selected platform, additional deliverables may be required.

Application products are presented to and reviewed by the customer representatives on a regular basis to help ensure that the product meets their expectations.

The project manager and Technical Lead finalize the unit, integration, and system test plans. The plans should define the participants (the test team), where and when training will occur, how test results will be transmitted back to the developers and what the estimated length of testing time will be needed. All plans should be agreed to by the development team and coordinated with the review team and prospective testers.

Unit testing is performed by the developers after each module has been completed and is used to test for proper logic and functionality prior to system testing. The unit test plan and documenting of results should follow a standard format. The

application must be free of programming errors upon migration to the integrated test platform. An inspection is conducted on 10% of the code.

During the **Integrated System testing phase** all components of the applications new and existing are tested in totality to ensure all functionality new and old has not been compromised. Data migration plans to transfer data from existing applications to the new application should also be regression tested at this stage and execution of the system test plan should take place. Any migration requirements are finished and routed to the appropriate DBA.

Installation and configuration of hardware and or network equipment for the Integrated Test Environment are coordinated and implemented.

The Project Manager ensures that the **acceptance test plan** has been completed and the Project Schedule has been updated.

Application testing is not a trivial process. Each function in the requirement's specification is successfully tested. Testers examine and experiment with all aspects of the application. Each function should handle the expected data entered as well as with the unexpected (e.g., entering invalid dates, entering letters where only numbers should appear, trying to avoid adding required information or records, etc.). The application will also be tested for its efficiency, security, and integration across planned hardware platforms and adherence to documented requirements. Any problems encountered or modification requests should be reported immediately to the project manager.

Corrections that do not affect the functional requirements should be made immediately. These minor modifications/corrections entail a return to the Development stage. The application changes will take place on the development platform. After changes are made and unit tested, the application is migrated back to the test platform.

Modifications to the application take place during this stage to correct the obvious problems that do not affect the functional requirements. Major changes to the functional requirements trigger a reiteration of previous stages. Existing documentation is maintained by the project manager to reflect all requirements changes; however, progress through the subsequent stages should proceed much more quickly. The stage is complete when the test team agrees that the application satisfies all documented requirements.

At the end of the test period, all testers' comments will be filed in the project file. The test team signs off on the test plan, indicating that testing has shown the application satisfies all requirements. The procedures and agreements described in the plan for acceptance testing are executed in order to conduct a formal turnover of the new application system.

During **Acceptance Testing**, the project manager coordinates with the DBA and systems administrator to load the data to the acceptance test platform. The volume of data used for testing is large enough to detect possible performance problems during the load testing. Application access is granted to a specific set of testers.

Testers should attend training where a variety of documents will be provided to each reviewer. These documents should include the user instructions, system test plan, a place for reviewers' comments, and a list of items to be tested. A copy of the

Requirements Document should also be provided to the testers. Through this training the testers will discover not only how to use the application, but also what the application is intended to do.

In addition, the user manual and training plan for the production system need to be nearing completion.

The **Implementation Plan** is prepared as all components of the system are readied for implementation. The Technical Lead meets with the development team and project manager to review the Implementation plan. Establish contingency plans for backout and recovery. Review the hardware/software installation plans for completeness. Review the conversion approach for conversion of data and programs to production. Determine **GO NO GO** step. System documentation and operational procedures are developed or updated accordingly. Obtain commitments from involved staff.

Coordinate with the DBA and Data Center and other appropriate administrators to install the application on the appropriate production platform(s). Data is migrated and access is granted to the appropriate customers. The customer-training plan is initiated and formal notifications to the affected business entities are made.

When there are any changes to the content of the documents a Project Revision Log must be initiated and a description of the change entered. Customer sign-off of the application will be obtained upon successful completion of the acceptance test. The project manager will complete the project close-out form and conduct the Close-out meeting.

Tasks

- Create database
- Finalize the data dictionary
- Research existing code modules for reusability
- Write and unit test the code based on the detailed design
- Update Test Plan (system test, acceptance test)
- Conduct unit testing
- Document unit testing results
- Conduct an inspection on 10% of the code
- Conduct integrated system test
- Document system test results
- Conduct load testing and document results
- Create an Implementation Plan for the Implementation Stage
- Update acceptance test approach in Project Plan, if required
- Initiate customer training plan
- Update Project Schedule

Note: a sequential ordering of tasks is not implied here.

- Migrate application to the appropriate test environment
- Grant application access to the acceptance test team.
- Conduct training for acceptance test team
- Provide test team with requirements specification and acceptance test plan
- Conduct Acceptance test
- Document results of acceptance test

- Complete user instructions
- Finalize customer training plan
- Obtain Customer Sign-off
- Review Implementation Plan
- Execute Implementation Plan
- Implement customer-training plan
- Cut-over to production
- Obtain customer sign-off

Note: a sequential ordering of tasks is not implied here.

Deliverables

[Unit Test Plan](#)

[Acceptance Test](#)

[Implementation Plan](#)

[Detail Design Specifications](#)

Integrated System Test Plan

Constructed System

[Project Status Report](#)

Customer training plan

[Updated Project Schedule](#)

Customer documentation

- o IRMD [Documentation Standards](#) for Statewide Systems

Roles

Technical Lead

Data Administrator

Database Administration

Developer

Project Manager

Quality Assurance Analyst

Customer Contact/Stakeholders

Test Team

CLOSURE STAGE

Function

Final administrative procedures for the ongoing maintenance of the application are set up and documented. Application is supported with corrective changes (bug fixes) and enhancements (new requirements). The purpose of the Maintenance Stage is to establish and maintain a consistent process for installing and updating systems maintained by IRMD. This process ensures the quality of a product established during development is maintained throughout the life of that product. This process includes activities to upgrade (technology), enhance (functionality), or correct (remove defects) the software in support of the customer.

Process

Once the system is implemented, a Post Implementation Review should be conducted to determine the effectiveness of the System Development process and if the benefits were achieved by the Systems implementation. Information from the project issue logs can be used to track the problems that arise during the project. The review includes identifying the controls that were used to address these problems and what could be done differently in the future to eliminate their recurrence. It is also used to assess the overall effectiveness of the projected costs, timeframes, plans to actual results and judging customer satisfaction with the installed system.

As the system moves into a maintenance phase, it is inevitable that at some point modifications will be required. There are specific tasks and procedures performed to maintain the software system. Maintenance includes upgrading, enhancing, or correcting existing software systems. A majority of these tasks will be described by IRMD Software Configuration Management (SCM) Plan and its derivative plans. In brief, the tasks include the steps required to: identify and report desired changes, approve work on change requests, implement the change, approve the completed change, and release the changed software into production. In addition to software changes, any documentation requiring modification will also be updated.

Modifications should be requested in writing through Systems Development and Consulting Change Control Process. Minor changes will be implemented on the development platform and proceed through the Testing Stages and on to the Implementation Stage. Changes deemed major by application maintenance staff will trigger the entire SDLC life cycle process to be initiated again. The application oversight is responsible for notifying customers of changes and updating the customer documentation. All records of modification will be filed in the project file.

Tasks

- Evaluate Application Performance
- Perform Post Implementation Review
- Establish Change Control Process
- Perform Routine Maintenance

Deliverables

[Post Implementation Review Report](#)
[Documented Change Control Process](#)

[Service Level Agreement](#)

Software modifications

Updated documentation

- IRMD [Documentation Standards](#) for Statewide Systems

Roles

Application Ownership

Application Maintenance Staff

Customer contact

Database Administration

Technical Lead

Project Manager

Quality Assurance Analyst

Appendix A: Inspections, Walkthroughs & Peer Reviews

There are three types of reviews incorporated into the Software Development Lifecycle Methodology (SDLC): Inspections, Stage-end Walkthroughs and Peer Reviews. These reviews are intended to add value to software development projects by detecting potential problems early so that they are resolved prior to becoming costly, and adding checks and balances to ensure the project goals are being met.

1. Inspections

1.1 Definition

An inspection is a structured formal review conducted to detect potential problems in a software development product early in the project lifecycle. Issue resolution is mandatory and rework is formally verified. Inspections occur prior to the stage-end walkthrough.

1.2 Objectives

- a. Identify issues with deliverable items.
- b. Ensure the deliverable is meeting the overall objectives.
- c. Verify that work meets predefined criteria.
- d. Promote correctness and completeness.
- e. Ensure the appropriate parties technically agree on the work.
- f. Consider alternative implementations.
- g. Educate/Update Participants

1.3 Review Items – Project Managers and/or Technical Leads are responsible for ensuring that the following products are reviewed, and edited for spelling and grammar, before being released to the inspection team for final review.

- a. Requirements Document
- b. Design Document – For projects following the full SDLC. For Rapid Development projects the data dictionary and data model are inspected.
- c. Code (10%) – In selecting code for reviewing, the following criteria should be considered by the Project Manager:
 - The most critical section
 - The most used section in the product
 - The most costly section if defects were to exist
 - The least well known section
 - The most frequently changed section
- d. Migration Plan (optional)

1.4 Process

- a. Inspection Participants
 - Moderator – Coordinates and leads the inspection meeting. The Moderator may be the Project Manager unless the Project Manager is the Author of the product being reviewed.
 - Recorder – Records the issues discussed during the inspection meeting on the Project Review Issue List form. Typically, the recorder is not part of the project team.
 - Author – Person who developed the product.
 - Inspectors - Project team members who review the product.
- b. Planning
 - Moderator should review materials/product to ensure they are ready. If the product is not ready for review, the Moderator should postpone the

- inspection until the product is ready. Moderator should ensure completeness of the product and correct formatting and spelling of the product.
- Moderator and Author decide on the objectives of the review.
- c. Preparation
- Moderator is responsible for disbursement of the inspection package and sending out the meeting notice.
 - Inspectors should review the product and record time and issues found on Project Review Issue List form.
 - Moderator ensures inspection team is ready.
 - Overview session – Depending on the complexity of the product an overview session may be scheduled by the Moderator to include the Author and Inspectors prior to the inspection meeting.
- d. Inspections Meeting
- Moderator briefs the participants – briefing should include the subject of the inspection, objectives of the inspection and the ground rules for the meeting.
 - Author briefs the participants – briefing should include overview of the software application and the product to be inspected. Also include any special considerations or areas of difficulty.
 - Inspectors provide a copy of their Project Review Issue List to the Moderator.
 - Inspection methods: Moderator goes through the review item either page-by-page, line-by-line, or correlating comments (by like comments). The Moderator may also use the tester model (review following a test plan/script) or use cases.
 - Inspectors call out the issues from their list.
 - Participants *briefly* discuss the issue and *quickly* move on (the Moderator allowing a predefined period of time per issue).
 - Inspections do not allow for a search for a solution to each potential issue encountered.
 - Recorder documents all issues for later resolution.
 - Moderator ensures that all issues have been addressed and summarizes any action items.
 - The severity of the issue is determined by consensus. In cases where consensus is not reached, the Moderator makes the final decision.
 - There are only three possible outcomes for the product: accept, conditionally accept or re-inspect. A conditional acceptance does not require re-inspection. If the review item is conditionally accepted, the Author should submit the corrected items to the Moderator for review and acceptance. If the outcome of the inspection is a re-inspection, then the product should be cycled through the inspection process from the beginning.
- e. Follow-up (for inspections with an outcome of “conditional acceptance”)
- Author fixes problems.
 - Moderator and author ensure all issues recorded are adequately resolved.
 - Moderator completes the Project Review Management Report form.
 - All forms are saved and filed in project folders and issues are recorded in the QA Product Issue database.
 - Product is typically entered into or back into configuration control.

1.5 Guidelines

- a. Review the product, not the Author.

- b. Identify and discuss problems, but do not attempt to solve them.
- c. Limit debate and rebuttal (set specific limits at start of the meeting, e.g. 30 seconds per issue and 2 rebuttals per issue).
- d. Set an agenda.
- e. Develop a checklist for each product (currently under development).
- f. Insist on preparation.
- g. Managers do not belong on inspection teams.

1.6 Documentation

- a. Project Review Issue List (one per Inspector and one aggregate developed by Recorder)
- b. Project Review Management Report

2. Peer Reviews (Optional)

2.1 Definition

A peer review is a structured formal review conducted to detect problems in a product early in the development process. The peer review is similar to an inspection, the only difference is the participants of the review. The participants in a peer review are peers of the Author that are not on the project team.

2.2 Objectives

- a. Identify issues with deliverable items.
- b. Verify that work meets predefined criteria.
- c. Promote correctness and completeness.
- d. Ensure the appropriate parties technically agree on the work.
- e. Consider alternative implementations.
- f. Educate/Update Participants

2.3 Review Items

- a. Requirements Document
- b. Design Document – For projects following the full SDLC. For Rapid Development projects the data dictionary and data model are inspected.
- c. Code (10%) – In selecting code for reviewing, the following criteria should be considered by the Project Manager:
 - The most critical section
 - The most used section in the product
 - The most costly section if defects were to exist
 - The least well known section
 - The most frequently changed section
- d. Migration Plan

2.4 Process

- a. Peer Review Participants
 - Moderator – Coordinates and leads the peer review.
 - Recorder – Records the issues discussed during the peer review on the Project Review Issue List form.
 - Author – Person who developed the product.
 - Peers – Persons who review the product.
- b. Planning
 - Moderator should review materials/product to ensure they are ready. If the product is not ready for review, the Moderator should postpone the peer review until the product is ready. The Moderator should

ensure completeness of the product and correct formatting and spelling of the product.

- Moderator and Author decide on objectives of the review.

c. Preparation

- Moderator is responsible for disbursement of the peer review package and sending out the meeting notice.
- Peers should review the product and record time and issues found on Project Review Issue List form.
- Moderator ensures peer review team is ready.
- Overview session – Depending on the complexity of the product an overview session may be scheduled by the Moderator to include the Author and Peers.

d. Peer Review Meeting

- Moderator briefs the participants – briefing should include the subject of the peer review, objectives of the peer review and the ground rules for the meeting.
- Author briefs the participants – briefing should include overview of the software application and the product to be reviewed. Also include any special considerations or areas of difficulty.
- Peers provide a copy of their Project Review Issue List to the Moderator.
- Peer Review methods: Moderator goes through the review item either page-by-page, line-by-line, or correlating comments (by like comments). The Moderator may also use the tester model (review following a test plan/script) or use cases.
- Peers call out the issues from their list.
- Participants *briefly* discuss the issue and *quickly* move on (the moderator allowing a predefined period of time per issue).
- Peer Reviews do not allow for a search for a solution to each potential issue encountered.
- Recorder documents all issues for later resolution.
- Moderator ensures that all errors and concerns have been addressed and summarizes any action items.
- There are only three possible outcomes for the product: accept, conditionally accept or re-review. A conditional acceptance does not require a re-review. If the review item is conditionally accepted the author should submit the corrected item to the Moderator for review and acceptance. If the outcome of the peer review is a re-review then the product should be cycled through the peer review process from the beginning.

e. Follow-Up (for peer reviews with an outcome of “conditional acceptance”)

- Moderator and Author ensure all issues recorded are adequately resolved .
- Moderator completes the Inspection Management Report form.
- All forms are saved and filed in project folders and issues are recorded in the QA Product Issue database.
- Product is typically entered into or back into configuration control.

2.5 Guidelines

- a. Review the product, not the author
- b. Identify and discuss problems, but do not attempt to solve them
- c. Limit debate and rebuttal (set specific limits at start of the meeting, e.g. 30 seconds per issue and 2 rebuttals per issue)

- d. Set an agenda
- e. Develop a checklist for each product (currently under development)
- f. Insist on preparation
- g. Managers do not belong on peer review teams

2.6 Documentation

- a. Project Review Issue List
- b. Project Review Management Report

Forms

- a. Project Review Issue List
- b. Project Review Management Report
- c. Stage-end Walkthrough form

Appendix G: Roles and Responsibilities

Application DBA – Performs design-related tasks as requested and required by an Application Development team.

Builds the physical data structures.

Make database design modifications

Troubleshoot problem queries and updates

Author – Person who developed the product.

Business Analyst – Provides knowledge about the functions and data/information of the processes to be automated. Typically, a representative of the customer community fulfills this role.

Customer representative – Defines projects and services (using Service Level Agreements, etc.). The single point of contact to assigned TRC customers.

Performs Consultant Information Assessments to provide management input regarding TRC policies, services, marketing and pricing structure.

Problem Resolution-liaison to IRMD customers

Provide any problem mitigation through coordinating resources and/or follow-up explanations and resolutions to a customer request and/or concerns

Provide oversight and ensure completion of assigned IRMD projects

Marketing (selling) IRMD services and products

Customer contact – member of the customer community, who provides knowledge about what the application needs to do. General responsibilities may include:

Coordinates activities with the customer community

Communicates with his/her upper management when necessary

Compiles equipment surveys and equipment needs evaluation.

Participates in needs assessments

Develops data element definitions and customer documentation.

Defines business rules

Coordinates customer training

Authorizes access and serves as the primary application contact after the application is installed.

Data Administrator – Analyzes, classifies and maintains an organization's metadata. Provides for the overall management of data as an organizational resource and asset. Helps as requested in the logical aspects of development projects (e.g., needs assessment, business rule specifications, logical database design).

Database Administrator – Responsible for the physical aspects of the database and applications. The database administrator works with the systems administrator to ensure a stable, reliable and recoverable DBMS environment. This includes, but is not limited to:

Allocates and manages physical (space) resources

Monitoring database activity and performance

Tunes database for performance

Installs and upgrades software

Plans, tests, and implements backup and recovery procedures

Developer – One or more individuals who conduct the needs assessment, design or develop the application. The developer may be from development staff, a program area or may be a contractor. General responsibilities include:

Conducts needs assessments

Evaluates software solutions

Develops the conceptual design of the system

Develops detailed program specifications
Develops logical and physical design of the database diagram
Write computer programs

Development Liaison – A development area staff member is assigned when a project is managed by either a contractor or program area staff (i.e., not managed by the development area). The Development Liaison's goals are to facilitate successful project outcome and ensure conformity with IRMD standards. General responsibilities include:
Acts as a technical consultant regarding information systems development, computer systems capabilities and SDLC standards and procedures.
Ensures that the development team has the current standards for deliverables.

Functional Owner – The business area executive who has the final responsibility for an application (usually a division director level) Organization or position who actually owns the data & application.
Has primary authority regarding business policy, and application and data access controls
Has primary authority to request application and/or data modifications
Approves and prioritizes proposed application modifications

Note: Application data may crowd organization boundaries and then may have multiple data owners.

Inspector – Reviews the product for the inspection meeting.

Moderator – Coordinates and leads a review meeting. The Moderator may be the Project Manager unless the Project Manager is the Author of the product being reviewed.

Network Administrator – Ensures connectivity and communication among networks, servers, and clients inside and outside IRMD.
Monitors and tunes the network to ensure acceptable levels of performance
Evaluates applications' connectivity requirements
Configures the network in regard to various protocol and bandwidth specifications
Ensures security is respected throughout the network environment

Peer – Person of equal or management level to the author of a product who performs a review of the product.

Project Manager – The person who oversees the project. The project manager may be a member of development, program area or contract team. General responsibilities include:
Facilitates meetings of the development team
Maintains a project file which contains copies of deliverables, correspondence and meeting minutes
Maintains timeline & associated tasks & assignments including progress tracking & reporting
Ensures compliance with proposal and established standards
Coordinates meetings with the review team and other Development staff, when applicable
Performs contract management, when applicable

Quality Assurance Analyst – Responsible for guiding software process improvement. The primary objective of quality assurance is to support data processing in increasing the quality of services provided. To accomplish this objective quality assurance will assure that:

Policies, procedures, and standards are developed, approved, implemented and maintained.
Information systems projects are reviewed at appropriate times during the project.
Quality concepts are promoted throughout information systems development.
Metrics are developed to measure the quality level of products and services produced for the customer.

Recorder – A scribe; person who documents all issues discussed during the review meeting.

Security Administrator – Develops and implements policy to ensure the integrity, protection, and availability to authorized persons of IRMD data and the hardware, software, and other components which are required for processing that data.

Server Administrator – Responsible for evaluating the impact and performance on the network server(s).

Service Center Representative – Acts as first line contact advocate between end user and technical staff. Unless technical staff requires more information from user to resolve problem, the Service Center representative will be the only contact liaison the end user will require. General responsibilities include:

Initial troubleshooting and diagnosis of problem

Contacting correct technical staff required to resolve problems when out of the realm of knowledge of the Service Center representative.

Logging calls into an electronic tracking media and providing a source of information to end users when specific IRMD wide applications have become disabled or unreachable.

Systems Administrator – Ensures the reliability, stability and recoverability of specific server environments. Normally the primary focus involves server hardware, associated operating systems, and DASD (Direct Access Storage Device). General responsibilities include:

Develops and implements backup strategies

Installs and upgrades operating system software

Configures and optimizes storage systems

Monitors and tunes system to ensure acceptable levels of performance

Technical Advisor – Person who provides technical knowledge to the development team.

Technical Lead – Person who oversees and manages the technical aspects of the project, this typically includes conceptual and detail design, construction and testing phases. This person works closely with project manager during Initiation and Definition stage as well as Needs Assessment. This role also provides technical knowledge to the development team.

Web Administrator: Ensures the reliability, usability, and recoverability of the web server software.

Web Designer – Person responsible for the look and feel (graphic design) of the web based application.

Web Master – Responsible for look and feel (graphic design) and adopting a demonstration, distribution, and public relations environment.

Acts as point of contact for web design.

Ensures web design standards are followed

NEEDS ASSESSMENT TEAM

The Needs Assessment Team is responsible for developing the requirements specification document and the customer deliverables list.

DEVELOPMENT TEAM

The development team is responsible for conducting the project tasks during the design and development stages of the project. Specific responsibilities are defined in the project plan. The number of individuals on the Development Team will vary depending on the size of the project.

REVIEW TEAM

The review team is responsible for reviewing and approving the deliverables during each stage of the project. The team should ensure that the appropriate documentation and notification is complete and that the SDLC and IRMD standards are followed. Review team members are assigned during the planning phase. Five key functional areas will be represented on the review team:

- Database Administration
- Data Administration
- Network Administration
- Web Administration
- Server Administration
- Systems Administration
- Quality Assurance

TEST TEAM

The Test Team is responsible for testing and evaluating the functionality of the application before it is released to the customer.

IMPLEMENTATION TEAM

The Implementation Team is responsible for defining and implementing the Implementation plan in the Testing and Implementation stages. The activities included with implementing the system are installation of hardware and software, data migration, acceptance testing, sign-off and customer training.

Appendix H: Glossary

Acceptance Testing: Formal testing conducted to determine whether a system satisfies its acceptance criteria and thus whether the customer should accept the system.

Analysis: Activities involved in understanding an issue, its associated processes and components, and possible methods that can be used to facilitate the processes electronically.

ANSI: American National Standards Institute.

Application: Software programs that interact with associated data to achieve a defined functionality.

Attribute: 1. Any detail that serves to qualify, identify, classify, quantify or express the state of an entity. An entity is comprised of attributes. When the logical E/R model is transformed into the physical schema, attributes will become columns in the physical tables. 2. A characteristic of a data element (e.g., data type, length, format, etc.).

Business Function: A specific action a customer performs or needs to perform.

Business Rule: Calculations, constraints, and conditional logic by which an organization operates.

CASE: Computer-Aided Software Engineering the combination of graphical, dictionary, generator, project management and other software tools designed to assist in one or more phases of software development.

Change Request Form: The Change Request form should be used by the project manager to document any change to the project's requirements or design of the application. The project manager should include any changes to the project estimate and schedule on the form. The project manager and customer contact should sign the form.

Client: The customer.

Column: data element or field, A basic unit of data that can be identified and described (Synonyms: data element;field).

Constraint: 1. An external, management or other factor that may confine business or development in terms of resource availability, dependencies or time scales. 2. Within Oracle DBMS - a rule or restriction concerning a piece of data that is enforced at the data level rather than at the object or application level (e.g. foreign key constraint, check constraint).

Customer: Client.

Data: Information which may be housed on a storage medium for communication, retrieval and processing by automatic means and presentation as information that is understandable to humans.

Data Administrator: A person or group that ensures the utility of data used within an organization by defining data policies and standards, planning for the efficient use of data, coordinating data structures among organizational components, logical database designs, and defining data security procedures.

Database: A shared collection of logically related data.

Database Administrator: A person or group responsible for the physical aspects of the database management system and applications. The database administrator works with the systems administrator to ensure a stable, reliable and recoverable DBMS environment.

Database Management System: Computerized management facilities that are used to store and manipulate data and to ensure privacy, recovery and integrity.

Data Dictionary: 1. A repository of all definitive information about the relevant data in an enterprise, including characteristics, relationships, usage and responsibility. 2. A central source within a database of information about database objects, users, privileges, events, and use.

Data Document: The data document can be a separate document attached to the requirements document or included in the requirements document. This documentation will consist of detailed information (i.e. type, size, edit criteria) on an existing systems data or a new systems data.

Data Element: A basic unit of data that can be identified and described.
(Synonyms:column;field)

Data Migration Plan: This plan, section 2.6 of the design document, describes data migration and transformation tasks which need to be accomplished throughout the development, test, and implementation stages. This plan shall include the data migration map and data migration/transformation activities. This should include proposed options for bad data. This plan includes taking existing data and transforming /migrating into the correct values/format of the new application.

Data Model: A logical view of entities and their relationships. (Synonym: Entity Relationship Diagram [ERD])

Deliverable: The product or documentation resulting from an application development stage.

Design: The determination of the overall architecture consisting of physical processing components, hardware, software, people and the communication among them which will satisfy the requirements.

Detailed Functional Requirements: See **Functional Requirements, Detailed.**

Developer's Manual: A guide for application development staff containing specific instructions and standards for applications development.

Domain: 1. The set of valid values for an attribute or column. 2. A group of computers whose hostnames share a common suffix, the domain name (e.g., dms.state.fl.us).

Eighty-Hour Rule: Each task should only incorporate 80 hours. This allows the project to be spaced out into manageable increments for the team to follow and achieve. During this 80 hours it is important to keep records of the estimated amount of time for the task and the actual time (to use for future tasks as well as future projects).

Entity: A thing of significance about which information needs to be known or held.

Entity Relationship (E/R) Diagram: A logical view of entities and their relationships. (Synonym Data Model)

Equipment Survey: deliverable of the needs assessment stage which provides an inventory of the hardware and software resources available in the customer community.

Estimating: The project manager calculates the estimated time needed to perform tasks of a project to estimate how long the project will take and how much the project will cost for all resources.

Field: A basic unit of data that can be identified and described (Synonyms: data element; field).

Format: A pre-established layout for data. Programs accept data as input in a certain format, process it, and provide it as output in the same or another format. All data is stored in some format with the expectation that it will be processed by a program that knows how to handle that format. Generically, data formats tend to fall into bitmaps (strings of 0's and 1's) that describe images or sound patterns (or both), text formats (in which usually each byte value is mapped to a character), and numeric data formats (used by spreadsheet and other database programs).

Full-time Equivalence (FTE): Percentage of full-time devoted to the project.

Function: An action.

Functional Requirements: A report documenting the hierarchy of actions within an application. As a project proceeds through its development cycle, higher level actions are decomposed into more specific steps.

Functions and Work Flow Documentation: A deliverable of the needs assessment stage which identifies customer community tasks and the flow of information through those tasks.

IEEE: Institute of Electrical and Electronics Engineering.

Implementation Plan: A deliverable in the beta testing stage which describes the procedures to move an application into production.

Information: Meaningful data.

Information Resource Management: 1. The concept that information is an important organizational resource and must be planned, managed and protected like other resources such as people, materials and financial resources. 2. The policy, action, and procedures concerning information, both automated and non automated, that management establishes to serve the overall current and future needs of an enterprise.

Information Systems: 1. That part of an organization responsible for the development, operation and maintenance of computer-based systems. 2. A defined and interacting collection of data, automated procedures and processes, along with the organized deployment of people, machines and other resources that carry out those procedures and processes. (Synonyms: Information Technology; Data Processing)

Software Development Lifecycle Methodology: A structured approach to developing applications in a data processing environment. It defines the stages, tasks, deliverables and techniques which lead through all steps in the life-cycle of a system.

Infrastructure: The environment of hardware, software, network, communications, and database management system components that supports technology.

Input Products: Any data source which the application requires such as documents, electronic data, data entry, etc.

Inspection: A structured formal review conducted to detect potential problems in a software development product early in the project life-cycle.

Maintenance: The modification of a software product, after delivery, to correct faults, to improve performance or other attributes, or to adapt the product to a changed environment.

Metadata: Information about an organization's data. Meta-data defines data structures and elements, itemizes associated attributes, describes how electronic data relates to real world items, notes business rules and how they affect certain data items, tracks how, when and who collects and/or changes data values.

Needs Assessment: Collection and validation of the requirements of the application customer.

Needs Assessment Summary: A deliverable of the needs assessment stage which reviews the results of the needs assessment activities.

Nullable: The capability of having no value.

Open Issues: Concerns identified but not resolved.

Output Products: Screens, reports, graphs, maps, queries, etc. which the application should produce.

Peer Review: A structured formal review conducted to detect problems in a product early in the development process. The peer review is similar to an inspection, except that the review team consists of peers to the author who may or may not be a part of the project team.

Physical Database Diagram: A representation of the database structures of an application.

Platform: Specific computer software, as in the phrase "platform-independent". It may also refer to a specific combination of hardware and operating system and/or compiler, as in "this program has been ported to several platforms". It is also used to

refer to support software for a particular activity, as in "This program provides a platform for research into routing protocols".

Process: 1. The set of steps that must be accomplished to complete a task. 2. Transformation of incoming data flow(s) into outgoing data flow(s). 3. An instance of a program running in a computer.

Production: The state of an application once it has been installed for use by the customer community.

Project File: A hardcopy record of deliverables, correspondence, meetings and notes accumulated during an application development project.

Project Management: The application of knowledge, skills, tools, and techniques to project activities in order to meet or exceed stakeholder needs and expectations from a project. Effective project management is necessary throughout the entire systems development life cycle.

Project Plan: A deliverable of the planning stage which identifies the staff involved in the project, estimated time, approaches and techniques which will be used.

Project Charter: The deliverable of the Project Charter Stage that initiates and prioritizes an application development project.

Project Schedule: A schedule of estimated project deadlines produced during the planning stage.

Prototype: A model or simulation of all or part of an application.

Record: A named collection of related data items or elements. Also see **Row**.

Reinspection: A repeat of an inspection of a product that was previously inspected.

Relationship: Any significant way in which two things of the same or different type may be associated.

Requirement: A feature an application must possess to meet the needs of the customer community.

Requirements Document: Requirements document provides documentation that is fundamentally a way of organizing thinking to be sure that all key points are considered in the requirements gathering process. The document is used in the determination and documentation of the customers requirements, functions and business rules.

Row: An entry in a table, consisting of values for each relevant column. Each row must be different in some way from any other row in the same table.

Security: The protection provided to prevent unauthorized or accidental access to a database or its elements; the updating, copying, removal or destruction of the database; or the changing or running of a proscribed application.

Software Test Plan: A project deliverable containing plans for unit, integration, and system testing activities. It also identifies who will participate in testing, where and

when training will occur, how test results will be transmitted back to the development team and the estimated length of the test period.

Stage-End Walkthrough: A planned review of all deliverables at the end of an SDLC stage. This is a high level review to ensure all deliverables are complete prior to moving to the next SDLC stage. The stage-end walkthrough should occur at the end of every stage beginning with the Needs Assessment stage and ending with the Test Stage.

System: A defined and interacting collection of real world facts, procedures and processes, along with the organized deployment of people, machines and other resources that carry out those procedures and processes.

Table: 1. A collection of rows in a relational database. 2. An article of furniture consisting of a flat top supported on one or more legs.

TAR: Technical Assistance Request. A formal call to Oracle's support office asking for help with an Oracle-related problem. TARs may be initiated by phone or via Oracle's on-line (Internet) support site. TARs should be initiated by IRMD's designated technical contact for Oracle.

Thirty-two Hour Rule: Real world scheduling and estimating. A schedule should anticipate a worker to work 32 hours of a 40 hour work week. This 32 hours is actual work on a particular project and takes into account other work related events such as phone calls, breaks, and meetings.

Training Plan: A deliverable of the beta testing stage which describes how customer training will be conducted.

Trigger: Any device that activates some other mechanism. An event that precipitates another event or series of events.

Unit Test: The process of debugging an individual component of an application.

UNIX: A multi-user, multitasking operating system originally designed for minicomputers, then revised for use on mainframes and personal computers. There are now many versions of UNIX used on many different platforms. UNIX is written in the C programming language.

User: Any individual or group of individuals who interact with the computer at a business application level.

User Manual: Documentation that contains instructions about how to navigate through the application.

Version Control: A means of identifying subsequent releases of an application.

View: A means of accessing a subset of a database as if it were a single table.

Walkthrough: A presentation and discussion of the products or documentation resulting from an application development stage.

Windows NT: Windows New Technology. A 32-bit operating system from Microsoft for high-end workstations, servers, and networks.

Work Flow: 1. Rules and procedures which define how information flows through the tasks. 2. The tasks, procedural steps, organizations or people involved, required input and output information, and tools needed for each step in a business process.