SDLC

NYSITS.org / Sarah Lauser

PRE-TEST QUESTIONS

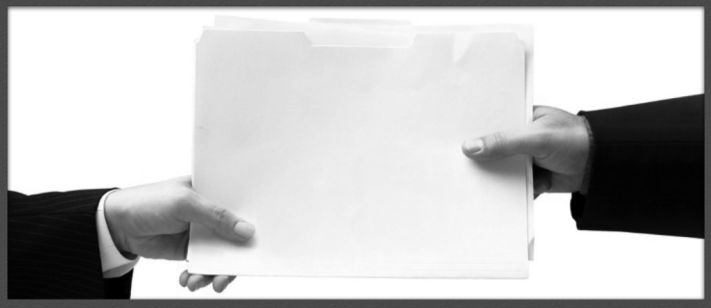
- What does SDLC stand for?
- Name at least 3 standard SDLC deliverables.
- How are SDLC and Project Management related?
- Give examples for each type of requirement - functional, technical, operational, and transitional.
- Describe at least 2 SDLC methodologies.

SYSTEM DEVELOPMENT LIFECYCLE

6 PHASES

PHASE 1

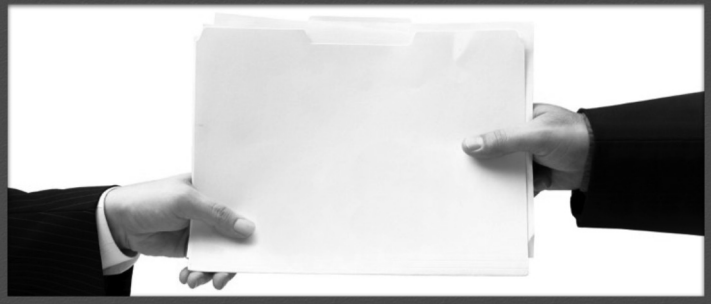
System Initiation



DELIVERABLES

Validated Solution, System Schedule

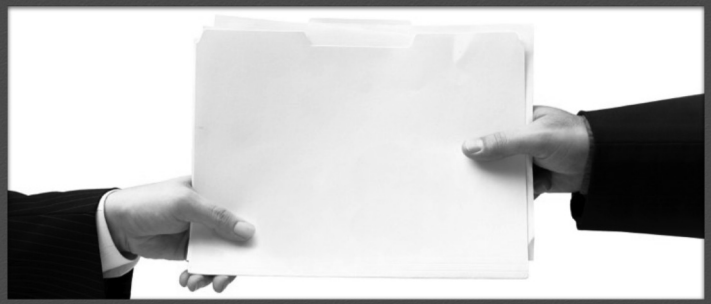
PHASE 2
System Requirements Analysis



DELIVERABLES

Business Requirements, Process Model,
Logical Data Model, Functional Specification

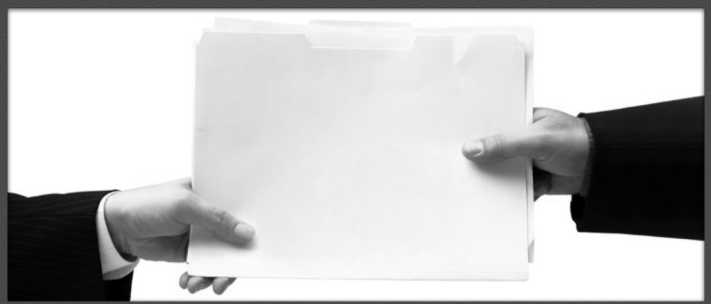
PHASE 3
System Design



DELIVERABLES

Technical Architecture, System Standards,
Database, Prototype, Technical Specification

PHASE 4
System Construction

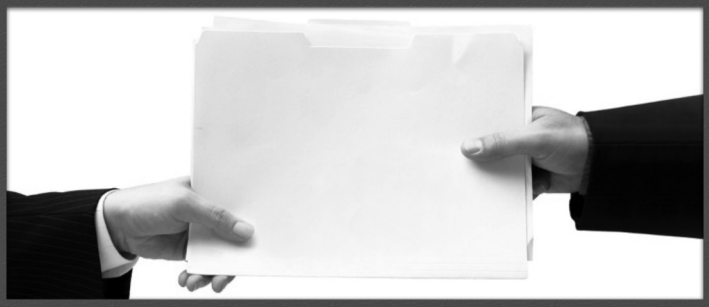


DELIVERABLES

Refined System Standards, Unit/Integration/System Test
Results, User and Training Materials, Documentation

PHASE 5
System Acceptance

Saturday, April 24, 2010



DELIVERABLES

Data Validation Results, Acceptance Test Results, Refined Training Materials and Documentation

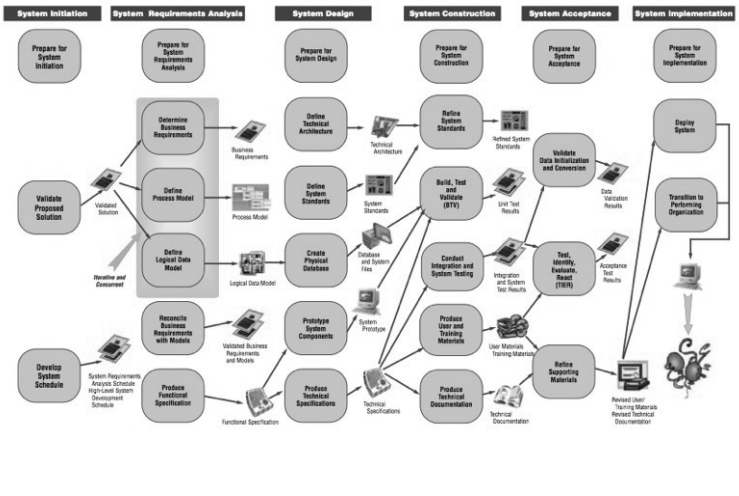
Saturday, April 24, 2010

PHASE 6
System Implementation

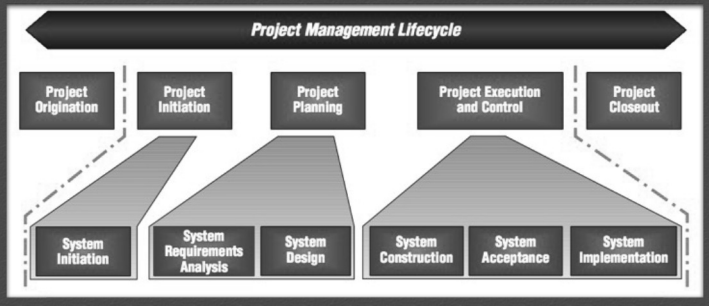
Saturday, April 24, 2010

Process	Measurements of Success	Yes	No
Prepare for System Implementation	Has anyone verified that every Consumer has the right level of system access and security?		
	Is there a checklist of all system components that can be used to verify that all the right versions of all components of the system are in the production environment?		
	Do the managers of Technical Services and Technical Support agree with your estimate of extra work for their units associated with new system deployment?		
Deploy System	Do your team members agree that their part of the effort as outlined in the Project Implementation and Transition Plan is reasonable and achievable?		
	Do the training evaluation forms filled out by Consumers and Customers being trained in the new system reflect scores equal or higher to those anticipated in the Project Implementation and Transition Plan?		
	Have you had to "freeze" or "fall back" in system deployment activities no more than originally anticipated in the deployment plan?		
	Is the volume of support calls within the range originally anticipated in the deployment plan?		
Transition to Performing Organization	Has the Performing Organization agreed to transition all of the remaining defects along with the system itself?		

Saturday, April 24, 2010



Saturday, April 24, 2010

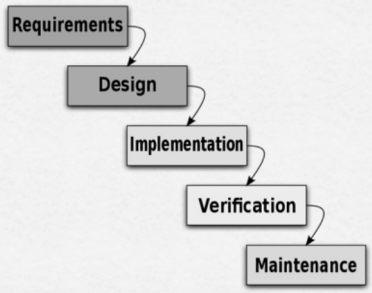


Integrating SDLC with the Project Management Lifecycle

Saturday, April 24, 2010

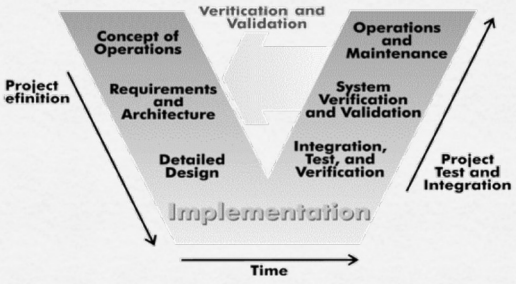
SDLC Methodologies

- Waterfall
- Fountain



SDLC Methodologies

- V



SDLC Methodologies

frequent "releases" in short development cycles	
Agile	real-time communication
XP	pairs programming
JAD	users participate in the SD process
LD	adding value to the customer
RAD	minimal planning, rapid prototyping
Scrum	product backlog, daily standup meetings

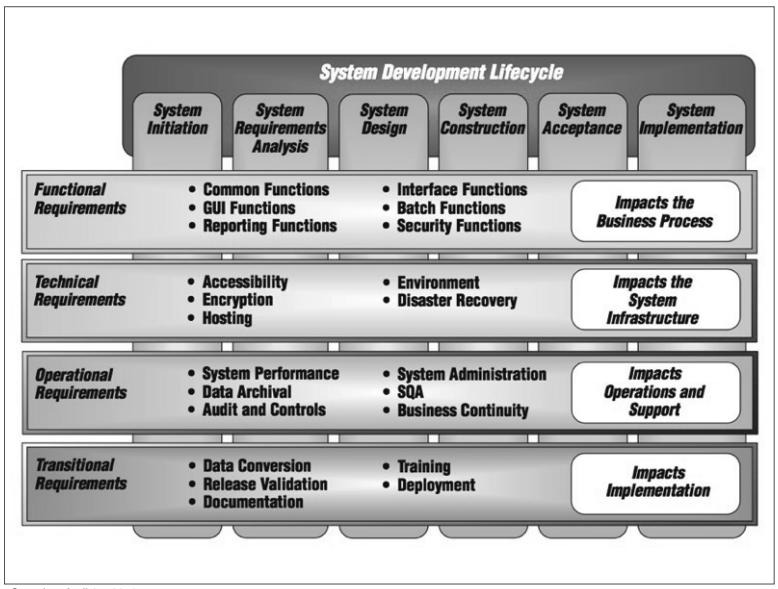
REQUIREMENTS

FUNCTIONAL

TECHNICAL

OPERATIONAL

TRANSITIONAL



SQA

Software Quality Assurance

SQA

Software Quality Standards

Software Quality Assurance Processes

Software Quality Controls

PROJECT ROLES

- Facilitator
- Business Analyst
- Database Administrator
- Data/Process Modeler
- Technical Lead/Architect
- Application Developers
- SQA Analyst
- Technical Services
- Information Security Officer
- Technical Support

BREAK REVIEW

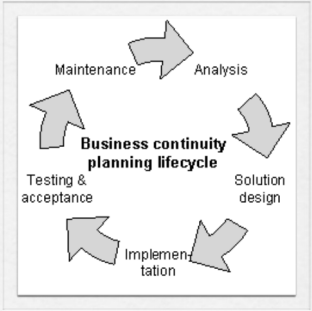
- What does SDLC stand for?
- Name at least 3 standard SDLC deliverables.
- How are SDLC and Project Management related?
- Give examples for each type of requirement - functional, technical, operational, and transitional.
- Describe at least 2 SDLC methodologies.

GLOSSARY

Mmmmm.... Alphabet Soup

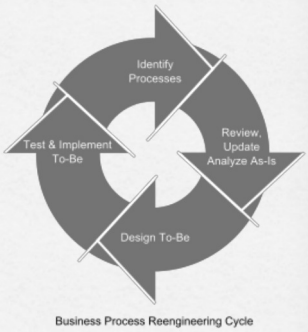
BCP/DR

Business Continuity Planning
Disaster Recovery



BPR

Business Process Re-engineering



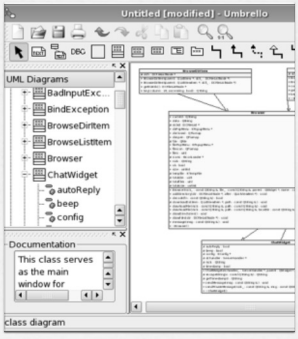
Business Process Reengineering Cycle

CMM

Capability Maturity Model

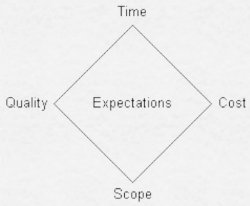
CASE

Computer-Aided Software Engineering



CSSQ

cost, scope, schedule, quality



Saturday, April 24, 2010

CSF

Critical Success Factor

Saturday, April 24, 2010

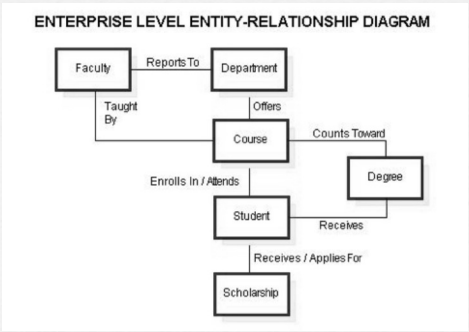
CRUD DIAGRAM SAMPLE

Entity Process	Customer	Customer Order	Customer Account	Customer Invoice	Vendor Invoice	Product
Receive Customer Order	R	C	CR			
Process Customer Order	CRU		RU			R
Maintain Customer Order	U		U		RU	
Terminate Customer Account	U		U		RU	
Fill Customer Order	RU		RU			RU
Ship Customer Order			U		C	
Validate Vendor Invoices					R	
Pay Vendor Invoices					RU	
Invoice Customer	RU		RU	C		
Maintain Inventory						CRUD

CRUD

create, read, update, delete

Saturday, April 24, 2010



ERD

Entity Relationship Diagram

Saturday, April 24, 2010

GUI

Graphical User Interface

Saturday, April 24, 2010

JAD

Joint Application Design

Saturday, April 24, 2010

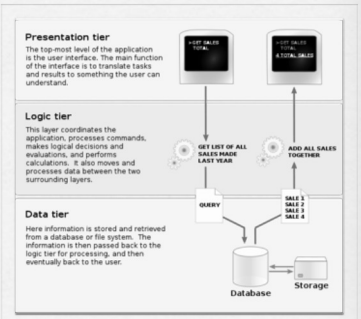
LAN/WAN

Local Area Network
Wide Area Network

WLAN
Wireless Local Area Network

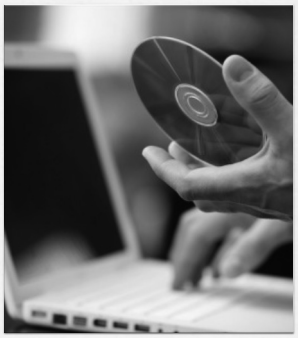
MT/CS

Multi-Tier/Client-Server



RAD

Rapid Application Deployment



SEI

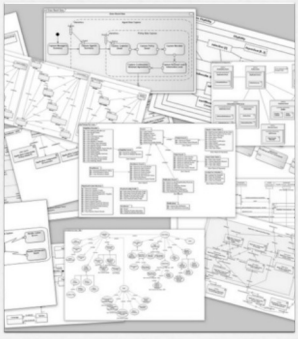
Software Engineering Institute

TQM

Total Quality Management

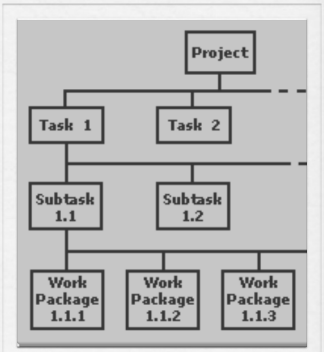
UML

Unified Modeling Language



WBS

Work Breakdown Structure



REVIEW QUESTIONS

QUESTION 1

_____ is the process of translating a task into a series of commands that a computer will use to perform that task.

A. Project design

B. Installation

C. Systems analysis

D. Programming

QUESTION 2

_____ spend most of their time in the beginning stages of the SDLC, talking with end-users, gathering information, documenting systems, and proposing solutions.

A. Business analysts

B. Project managers

C. Network engineers

D. Database administrators

QUESTION 3

The _____ determines whether the project should go forward.

A. feasibility assessment

B. opportunity identification

C. system evaluation

D. program specification

QUESTION 4

A _____ is used to schedule the time it will take to complete computer tasks or program development.

A. WBS

B. Gantt chart

C. data flow diagram

D. data dictionary

QUESTION 5

Within data flow diagrams, the transformations that occur within the lowest level are described by

A. development methodologies

B. structure charts

C. selection constructs

D. process specifications

QUESTION 6

Which of the following is the tool used by database designers to document a conceptual data model?

A. Entity-Relationship diagram

B. Partition statement

C. Matrix diagram

D. Data-flow diagram

QUESTION 7

Technical writers generally provide the _____ for the new system.

A. programs

B. network

C. analysis

D. documentation

QUESTION 8

In a systems development process, users are made active members of development project teams. This is an example of

A. RAD

B. JAD

C. Waterfall

D. documentation

QUESTION 9

The stage in a system's life cycle in which logical and physical specifications are produced is called

A. initiation

B. design

C. construction

D. acceptance

QUESTION 10

In most organizations, the entire system-building effort is driven by

A. availability of packaged applications

B. existing hardware

C. user training requirements

D. user information requirements

SECTION THREE

III

System Development Lifecycle

SECTION III: SYSTEM DEVELOPMENT LIFECYCLE
TABLE OF CONTENTS

Section III Introduction

There are currently many different methodologies employed for system development projects within New York State agencies. Many methodologies are driven by the application development tools, by the software architecture within which the application will operate, or by the "build versus buy" decision. There are standard phases and processes, however, that all system development projects should follow, regardless of environment and tools. This section describes the standard phases and major processes of the New York State System Development Lifecycle (SDLC), using a common language and in sufficient detail to enable a Project Manager to plan and manage a system development project.

System Development Lifecycle Overview

The material in this section is organized according to a generic system development lifecycle. While no two development efforts are exactly alike, all projects should progress through the same six phases:

1. **System Initiation** – in which the Business Case and Proposed Solution developed during Project Origination are re-examined to ensure that they are still appropriately defined and address an existing organizational need. This validation effort provides the Project Team with the basis for a detailed schedule defining the steps needed to obtain a thorough understanding of the business requirements and an initial view of staffing needs. In addition, a high level schedule is developed for subsequent system development lifecycle phases.

2. **System Requirements Analysis** – in which the needs of the business are captured in as much detail as possible. The Project Manager leads the Project Team in working with the Customers to define what it is that the new system must do. By obtaining a detailed and comprehensive understanding of the business requirements, the Project Team can develop the Functional Specification that will drive the system design.

3. **System Design** – which builds upon the work performed during System Requirements Analysis, and results in a translation of the functional requirements into a complete technical solution. This solution dictates the technical architecture, standards, specifications and strategies to be followed throughout the building, testing, and implementation of the system. The completion of System Design also marks the point in the project at which the Project Manager should be able to plan, in detail, all future project phases.

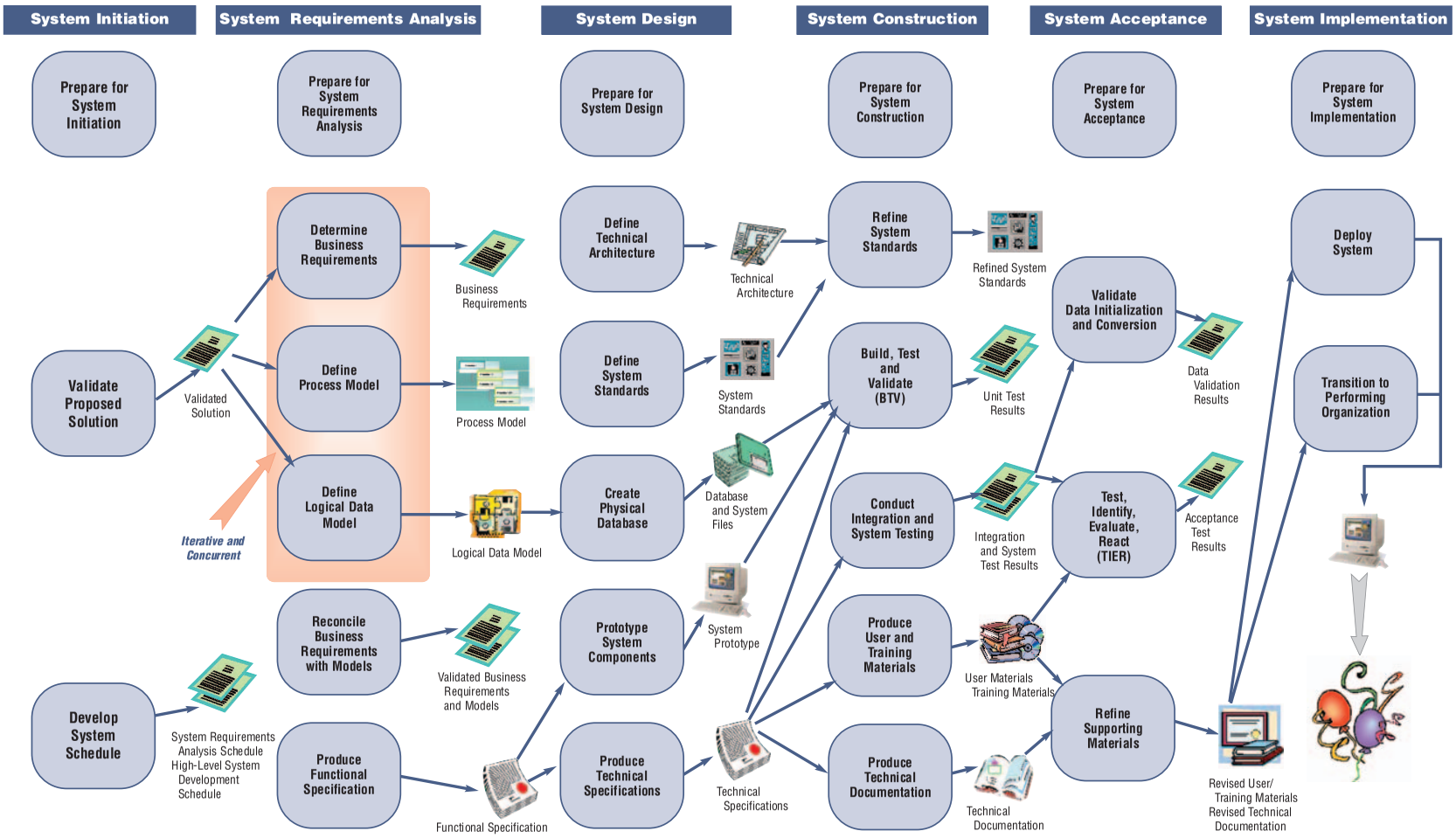
4. **System Construction** – throughout which the Project Team builds and tests the various modules of the application, including any utilities that will be needed during System Acceptance and System Implementation. As system components are built, they will be tested both individually and in logically related and integrated groupings until such time as a full system test has been performed to validate functionality. Documentation and training materials are also developed during this phase.

5. **System Acceptance** – during which the focus of system validation efforts shifts from those team members responsible for developing the application to those who will ultimately use the system in the execution of their daily responsibilities. In addition to confirming that the system meets functional expectations, activities are aimed at validating all aspects of data conversion and system deployment.

6. **System Implementation** – the final phase of the lifecycle, which comprises all activities associated with the deployment of the application. These efforts include training, installation of the system in a production setting, and transition of ownership of the application from the Project Team to the Performing Organization.

The following diagram illustrates every phase, process and deliverable in the system development lifecycle.

Figure 0-1 *NYS Project Management Guidebook* The System Development Lifecycle

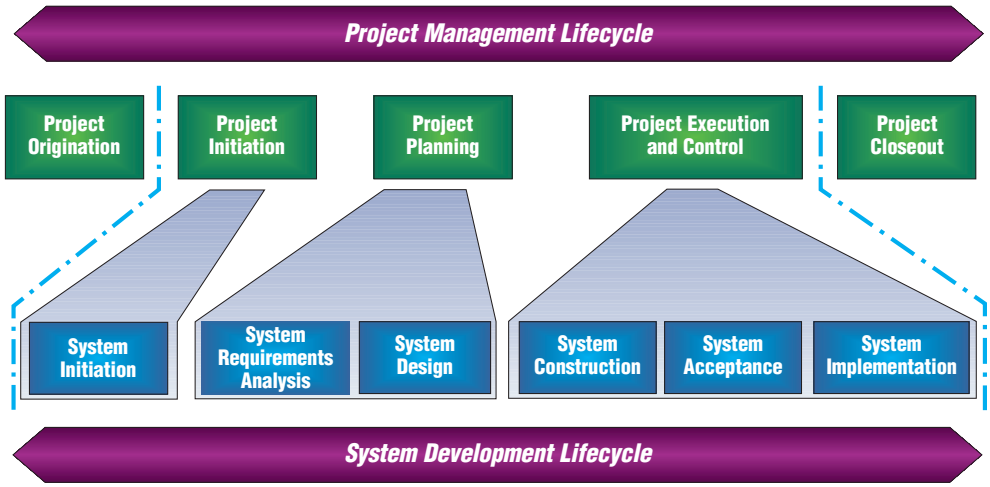


NYS Office for Technology
Project Management Office

Mapping the Project Management and System Development Lifecycles

The phases of the system development lifecycle generally align with the phases of the project management lifecycle; however, SDLC phases do not correspond one-to-one with the project management phases. One of the challenges for system development projects is aligning the SDLC with the project management lifecycle. The following diagram demonstrates how the phases of the two lifecycles may be integrated.

Figure 0-2



In reality, each phase of the SDLC can be thought of as a mini-project in itself, requiring planning, execution, and analysis. As the Project Team proceeds through the project, they will need to create a clear and detailed plan for the phase immediately in front of them, along with a higher-level view of all remaining phases. As the team executes each phase, they will collect additional information that will enable the detailed planning of subsequent phases. Some of this information will be a natural by-product of having performed the processes associated with the current phase (e.g., as the detailed technical design evolves throughout the System Design phase, the team will have a much better understanding of the modules that will need to be built during construction, and will therefore be able to refine any prior estimates and plans for System Construction). Additional information can be obtained through a focused

analysis effort, performed at the completion of each phase. This assessment is analogous in many respects to conducting the Post-Implementation Review as described in Section I, Project Closeout, although it is typically conducted in a less formal fashion. The responsibilities of the Project Manager include assessing how closely the phase met Customer needs, highlighting those aspects of the phase that worked well, identifying lessons learned and best practices in an attempt to derive ways to improve upon processes executed throughout the project, and, most importantly, communicating results.

The SDLC defined in this section may appear to have characteristics of a classic "waterfall" approach, which assumes that each phase and process is completed and agreed upon before the next phase begins. The reality is, however, that phases generally overlap, with each successive phase introducing changes to the work of the prior phase, resulting in an iterative process.

This SDLC is also consistent with newer techniques for system development, such as Rapid Application Development (RAD). RAD allows users to participate in an iterative design and development process. Conceptually, the project "loops" through the Design, Construction, and Acceptance phases, followed by re-Design, revised Construction, Acceptance, and so on. Project management deliverables such as the Project Scope Statement, Project Schedule, and budget estimates are refined to reflect increasing clarity of scope and requirements with each iteration.

While there is the potential to compress Requirements Analysis, Design, and Construction in RAD approaches, compression introduces increased risks. It is important, therefore, to include risk analysis in each iteration of the design, build, and evaluate loop. When a prototype is presented, Project Managers must actively and diligently address the management of Customer expectations and the maintenance of current documentation.

The RAD approach has advantages, since it usually achieves results quickly, the design is less abstract, and users have assurance that up-to-date requirements are considered. Its disadvantages include difficulty in controlling the process and ensuring the creation of an acceptable product.

Many factors may impact your choice of approach to follow when developing a system. The better you know your Customers and Stakeholders, and the better you understand the factors that may influence their assessment of the project, the more likely it will be that your approach will suit their personalities, preferences, vision, and needs.

The key is to pick the approach that you believe will provide the best complete solution, balancing the preferences of your Customers, the abilities of your Project Team, and the overall business drivers (legislated timeframes, overall time to market, etc.).

In any approach, the basic SDLC processes must be performed – what differs is the timing of their execution. As with the project management methodology, if processes or deliverables are skipped, the Project Manager must record the reasons why, and must describe how the objectives of that process/deliverable will otherwise be met.

Understanding the Breadth of System Development Projects

When assessing the scope of a system development project, it is important that the needs, goals, and challenges of the project are understood from many perspectives. The ***business requirements***, which define the high-level Customer objectives and vision for the system, are used to determine the scope of the system. When capturing the business requirements, it is essential that the Project Team look at all aspects of the system, including:

- **Functional Requirements** – describing processes and tasks that the Consumer must be able to accomplish through the use of the system. These can typically be categorized as processes that require action on the part of Consumers (data entry, selection of a system command, etc.), and those that are not directly related to human interaction with the system (for example, off-hours processing or the automated exchange of information between systems).
- **Technical Requirements** – identifying technical aspects and constraints that must be considered when defining the new system. Considerations may include accessibility needs of Consumers, whether or not the storage and

handling of data must follow specified encryption regulations, or whether the system will operate on internal agency hardware or will be hosted at either an internal or external data center.

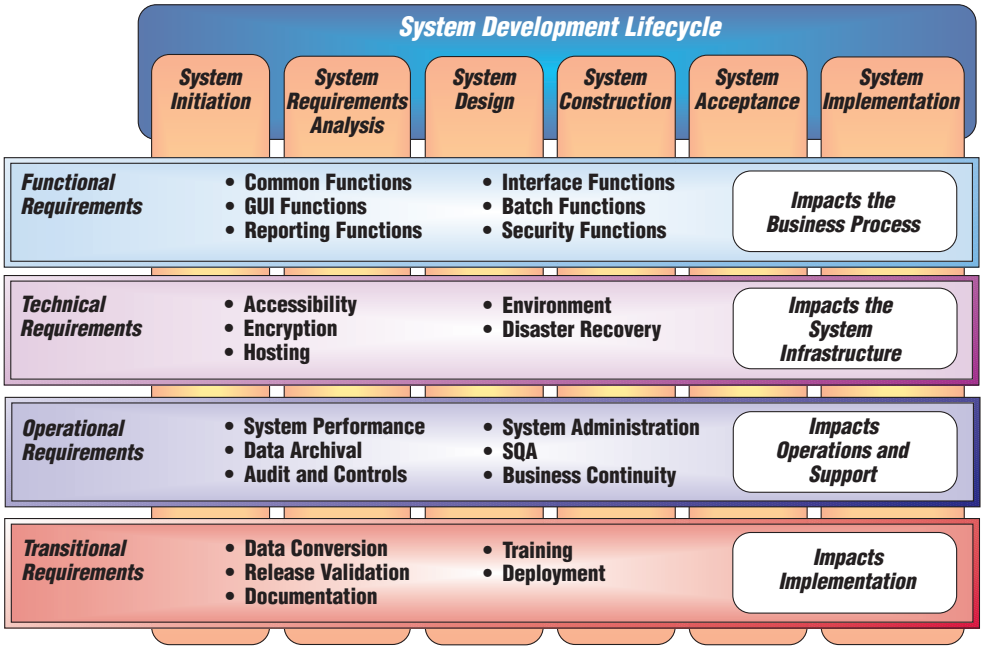
- **Operational Requirements** – specifying any administrative constraints or expectations that must be supported by the system. These requirements may include system performance expectations, technical infrastructure constraints, security mechanisms that must be followed, the need to regularly archive data, and any mandated audit and control processes.
- **Transitional Requirements** – defining the realm of conditions that must be satisfied prior to physically implementing the system in a production environment, or to relegating support responsibilities to the Performing Organization. Data conversion requirements and development and delivery of Consumer training programs and materials fall into this category.

Formally organizing thoughts along these four dimensions will drive the identification of tasks to be performed beginning in System Initiation and continuing throughout the lifecycle. The Project Manager is responsible for creating this broad view of requirements and communicating it to the Project Team, establishing a pattern that should be carried throughout all project phases.

The following diagram illustrates some representative categories of requirements that the team should consider when defining tasks and activities throughout the system development project.

Figure 0-3

Representative SDLC Considerations



It should be noted that not all considerations may be applicable to every project, and additional categories may be discovered that are not represented in the diagram. The fundamental point is that for those considerations that do apply to the project, there will be corresponding activities required throughout each and every phase of the SDLC, all contributing to the eventual implementation of the system. Regardless of whether the Project Team is performing System Initiation, Requirements Analysis, Design, Construction, Acceptance, or Implementation activities, they will need to understand and address the full realm of functional, technical, operational, and transitional requirements to ensure a successful project. In an attempt to reinforce this point, this diagram will be revisited in each of the individual SDLC phases that follow, drawing specific references to the processes relevant to each phase of the lifecycle.

Software Quality Assurance

In the way that project management provides the umbrella under which all project activities are directed, software quality assurance provides the foundation on which all system development activities should occur so that the highest quality system possible will be delivered. According to the IEEE Standard Glossary of Software Engineering Terminology, quality is defined as the degree to which a system, component, or process meets specified requirements and Customer needs and expectations.

As will be stressed throughout the following chapters, it should be noted that simply meeting requirements is not enough to guarantee a successful system development effort. Ultimately, Customer needs and expectations can be met only if the requirements are fully and correctly captured in the first place.

Analogous to the Quality Assurance Plan associated with the project management lifecycle, software quality assurance programs should be comprised of three components – quality standards, quality assurance processes, and quality controls.

Software Quality Standards define the programming standards, and development/testing standards to be followed throughout the project.

Software Quality Assurance Processes define practices and procedures to be used by the Project Team to meet the quality standards, and to provide management with evidence that these procedures are being followed.

Software Quality Controls comprise a series of reviews and audits that evaluate deliverables with respect to defined standards and acceptance criteria. These controls include software testing techniques and peer reviews.

The key to these SQA efforts is that they must be performed throughout all phases of the project. In addition, all SQA efforts should ideally be performed by a third party, independent from the team members responsible for delivering the system. Availability of staff and budget are two factors that must be considered in determining the feasibility of applying an independent SQA Analyst or team to the project. In developing the

overall system development plan, the Project Manager needs to allocate sufficient time and resources to perform the appropriate level of SQA activities, and must obtain management commitment to providing these resources as called for in the Project Schedule.

Project Roles and Responsibilities

As presented in the Section I Introduction, Project Roles and Responsibilities, there are many groups of people involved in both the project and project management lifecycles. When staffing system development projects, there are a number of roles that should be considered. It should be noted that the SDLC only provides details to the phase and process level, whereas the PM lifecycle further decomposes activities down to individual tasks. As a result, while the roles identified within the SDLC are representative of those that are typically required in a system development effort, the function of the role as it relates to a given SDLC process may not be specifically described within that process narrative.

The **Project Team** consists of a Project Manager and a variable number of Project Team members who are responsible for planning and executing the project. Team members specific to the System Development Lifecycle are described below.

The **Facilitator** leads sessions to identify business requirements and issues, keeps sessions focused and productive, draws out issues and ideas from all participants, and maintains clear and open communications within the session.

The **Business Analyst** effectively leads discussions with the Customers to determine the business requirements, participates in preparing the data and process models, prepares module specifications, test data, and user documentation materials, assists in prototyping activities, and develops strategies for testing and implementation.

The **Database Administrator** is responsible for providing and maintaining database administration policies and procedures, approving and executing database scripts, performing database tuning activities, and transforming a pictorial representation of the system data (the Logical Data Model) into physical database tables that support the final system.

The **Data/Process Modeler** develops and maintains data and process models to represent the business information needs in the area under study, develops and defines the data dictionary, validates models with the Customers, and participates in prototyping.

The **Technical Lead/Architect** drives the logical process and data models into an application architecture, establishes architecture guidelines, and develops strategies for the creation and distribution of applications.

Application Developers include all those responsible for developing prototypes, technical specifications, and application code, and for executing test scripts.

The **Software Quality Assurance (SQA) Analyst** is responsible for establishing and executing the Quality Assurance Plan, for assisting in the preparation of test scripts and test data, and for participating in integration and acceptance testing efforts.

Technical Services (HW/SW, LAN/WAN, TelCom) include all those responsible for the ordering, installation and maintainence of hardware and software components, LAN/WAN components and telecommunications components.

The **Information Security Officer (ISO)** is responsible for identifying and enforcing security standards and processes.

Technical Support (Help Desk, Project Administration, Documentation, Trainers) includes all those responsible for supporting the development of the new system. Support includes the documentation of user, training, operation materials, and help files, training for Customers, responding to technical and business questions forwarded to the Help Desk, and supporting the project and associated administrative processes.

Figure 0-4 New York State System Development Life Cycle Templates

Phase	Template	Description	Page in Text	Page in Appendix
System Requirements Analysis	Business Requirements Document	A document containing detailed functional, technical, operational and transitional requirements for the system being developed	45	99
System Requirements Analysis	Functional Specification	A document describing the logical grouping of related processes and the mapping of those processes to business requirements and data items.	59	103
System Design	Technical Architecture	A document describing the system architecture in terms of hardware, software, tools and peripherals, and the distribution of system components and processes across this architecture.	81	109
System Design	System Standards	A document detailing the standards to be applied and adhered to throughout the project.	87	115
System Design	Technical Specifications	A compilation of system diagrams, module specifications, and test plans that serve as a detailed, comprehensive blueprint for the system.	109	121
System Construction	Defect Log	A document used to log defects encountered when performing integration, system, data validation or acceptance testing, and track their resolution.	145	131