204362 – Object-Oriented Design

Introduction to Systems Analysis and Design

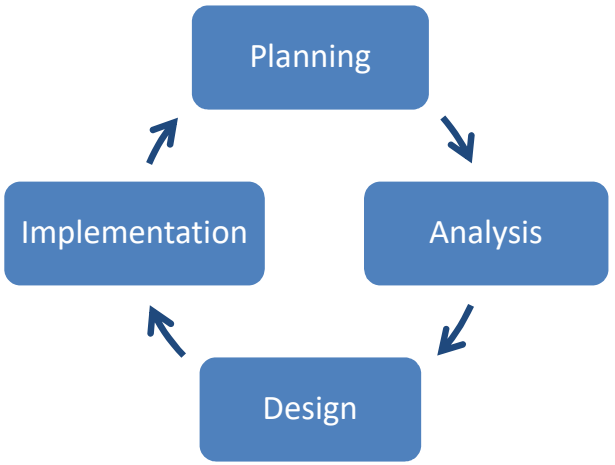
Adapted for 204362
by Areerat Trongratsameethong

Objectives

- Understand the fundamental systems development life cycle and its four phases.
- Understand the evolution of systems development methodologies.
- Be familiar with the Unified Process and its extensions.
- Be familiar with the different roles on the project team.

SYSTEMS DEVELOPMENT LIFE CYCLE

Systems Development Life Cycle



SDLC: Planning

1. Project Initiation
   - Develop a system request: วิเคราะห์ความต้องการในภาพรวม
   - Conduct a feasibility analysis: วิเคราะห์ความเป็นไปได้
     - Technical feasibility (Can we build it?)
     - Economic feasibility (Will it provide business value?)
     - Organizational feasibility (If we build it, will it be used?)
   - Develop work plan: ผู้จัดการโครงการ (Project Manager) สร้างแผนการดำเนินงาน
   - Staff the project: กำหนดคนที่จะทำงานในโครงการ
   - Control and direct the project: วางเทคโนโลยีที่จะช่วยให้ควบคุมและกำหนดทิศทางการทำงานในทีม

<u>Why</u> should we build this system?

SDLC: Analysis

1. Develop analysis strategy
   - Analysis of the current system (called the *as-is system*) and its problems
   - Design a new system (called the *to-be system*)
2. Gather requirements
   - Requirement Captures
3. Develop a system proposal
   - Describes what business requirements the new system should meet.

<u>What</u> should the system do for us?
<u>Where</u> and <u>when</u> will it be used?

SDLC: Design

1. Develop a design strategy:
   - Developed by the company's own programmers.
   - Outsourced to another firm.
   - buy an existing software package.
2. Design architecture and interfaces: describes the hardware, software, and network infrastructure to be used.
3. Develop databases and file specifications: define exactly what data will be stored and where they will be stored.
4. Develop the program design: defines the programs that need to be written and exactly what each program will do.

The design phase decides **how the system will operate**, in terms of the hardware, software, and network infrastructure; the user interface, forms and reports; and the specific programs, databases, and files that will be needed.

<u>How</u> will we build the system?

SDLC: Implementation

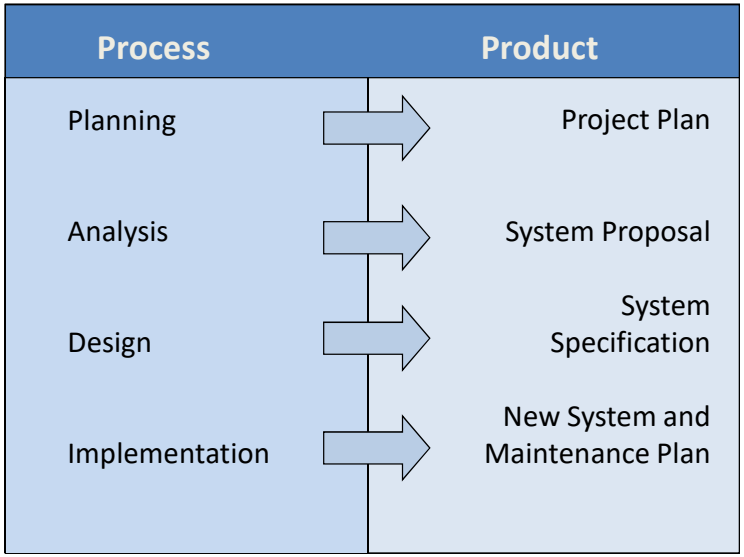
1. Construct system: The system is built and tested to ensure it performs as designed.
2. Install system
   - Implement a training plan for the users: to teach users how to use the new system and help manage the changes caused by the new system.
3. Establish a support plan: formal or informal post-implementation review as well as a systematic way for identifying major and minor changes needed for the system

<u>Build</u> the system!

Putting the SDLC Together

- Each phase consists of steps that lead to specific deliverables: แต่ละเฟสประกอบด้วยหลายขั้นตอน ในแต่ละขั้นตอนจะมีการส่งมอบเอกสารต่างๆซึ่งอาจจะอยู่ในรูปไดอะแกรมต่างๆเพื่อความเข้าใจตรงกันของคนทำงานในทีมพัฒนาและผู้ใช้งานระบบ
- The system evolves through gradual refinement: เอกสารที่ส่งมอบในเฟสก่อนหน้าจะเป็นข้อมูลนำเข้าของเฟสถัดไป เอกสารที่ส่งมอบในแต่ละเฟสจะถูกปรับแต่งให้มีความละเอียดมากขึ้นในเฟสถัดๆไป
- Once the system is implemented, it may go back into a planning phase for its next revision (การปรับปรุงแก้ไข), a follow-on system, or maintenance releases

Processes and Deliverables



SYSTEMS DEVELOPMENT METHODOLOGIES

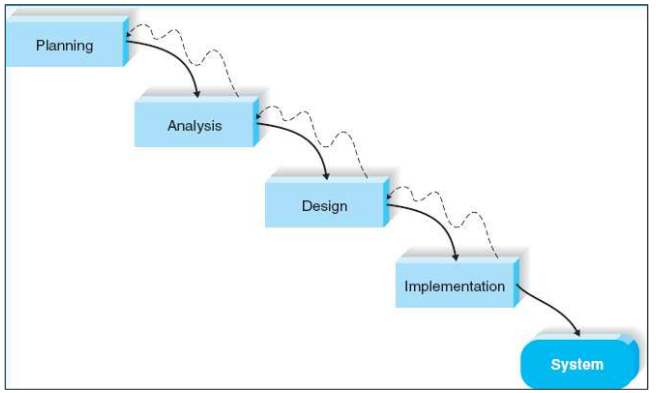
Systems Development Methodologies

- A *methodology* is a formalized approach to implementing the SDLC
- Well-known methodologies include:
  - Waterfall development
  - Parallel development
  - V-model
  - Rapid application development
  - Agile development

Categories of Methodologies

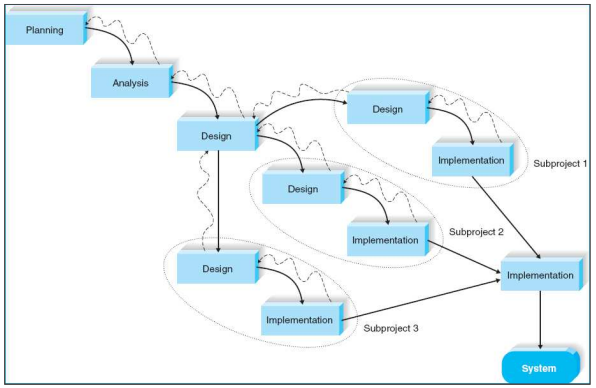
- Structured Design
  - Waterfall Development
  - Parallel Development
- Rapid Application Development
  - Phased
  - Prototyping
  - Throwaway Prototyping
- Agile Development
  - eXtreme Programming

Structured Design 1



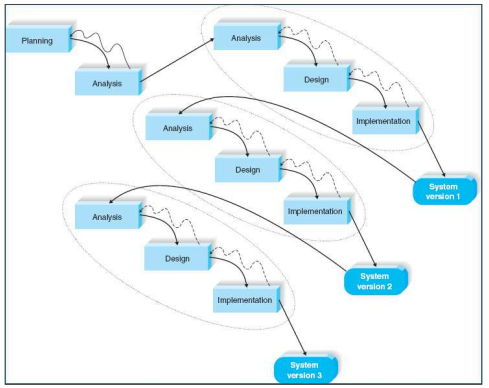
Waterfall Development

Structured Design 2



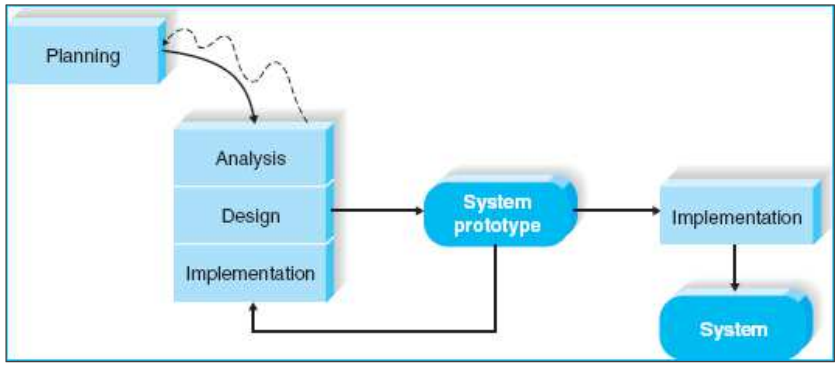
Parallel Development

Rapid Application Development 1



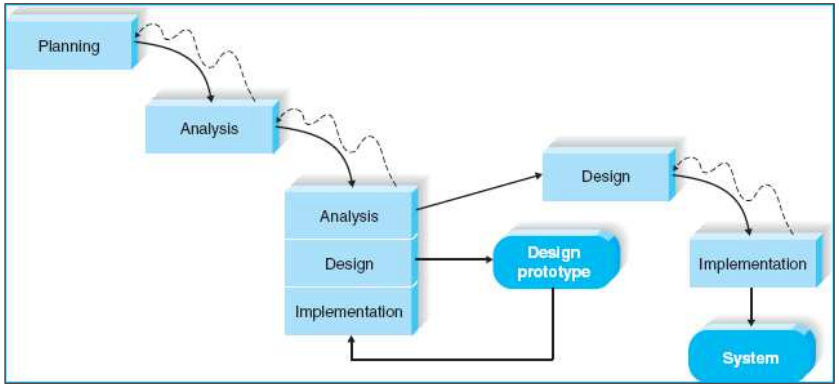
Phased Development

Rapid Application Development 2



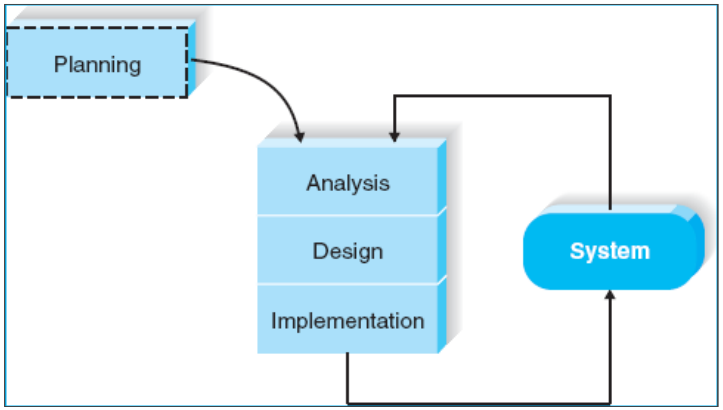
System Prototyping

Rapid Application Development 3



Throwaway Prototyping

Agile Development



Extreme Programming

Selecting the Right Methodology

Usefulness for	Waterfall	Parallel	Phased	Prototyping	Throwaway Prototyping	Extreme Programming
Unclear user requirements	Poor	Poor	Good	Excellent	Excellent	Excellent
Unfamiliar technology	Poor	Poor	Good	Poor	Excellent	Poor
Complex systems	Good	Good	Good	Poor	Excellent	Poor
Reliable systems	Good	Good	Good	Poor	Excellent	Good
Short time schedule	Poor	Good	Excellent	Excellent	Good	Excellent
Schedule visibility	Poor	Poor	Excellent	Excellent	Good	Good

Object-Oriented Analysis & Design

- Attempt to balance emphasis on data and process
- Uses Unified Modeling Language (UML)
- Characteristics of OOAD:
  - **Use-case Driven**: use cases are the primary modeling tools defining the behavior of the system. A use case describes how the user interacts with the system to perform some activity.
  - **Architecture Centric**: the underlying software architecture of the evolving system specification drives the specification, construction, and documentation of the system. Support 3 separated view.
    - **Functional**: describes the behavior of the system from the perspective of the user.
    - **Static**: describes the system in terms of attributes, methods, classes, and relationships.
    - **Dynamic**: describes the behavior of the system in terms of messages passed among objects and state changes within an object.
  - Iterative and Incremental: building up the three architectural views little by little.

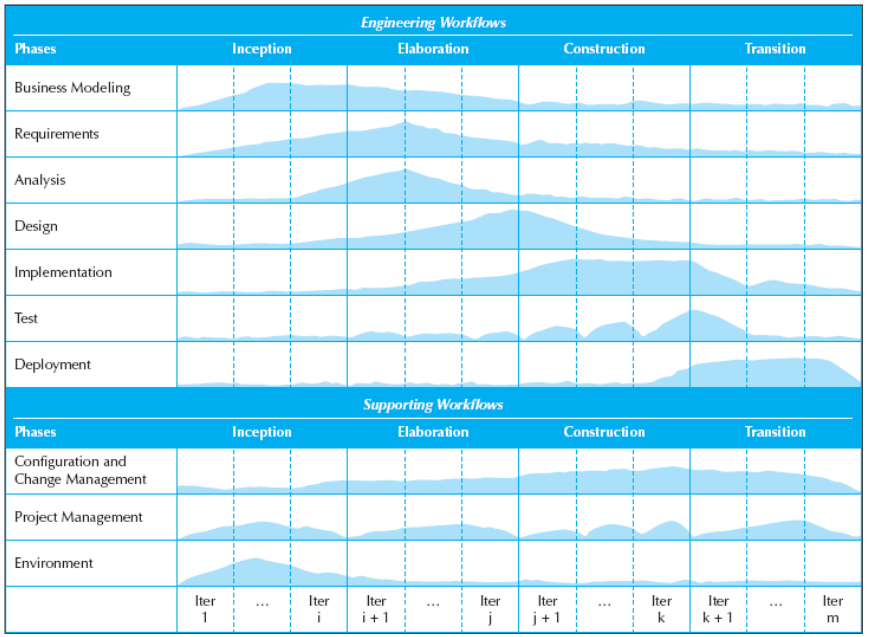
https://en.wikipedia.org/wiki/Unified_Modeling_Language
The Unified Modeling Language (**UML**) is a general-purpose, developmental, modeling language in the field of software engineering, that is intended to provide a standard way to visualize the design of a system.

THE UNIFIED PROCESS

The Unified Process

- A specific methodology that maps out when and how to use the various UML techniques for object-oriented analysis and design
- A two-dimensional process consisting of phases and flows
  - Phases describe how the system evolves over time
  - Workflows are collections of tasks that occur throughout the lifecycle, but vary in intensity

The Unified Process



Unified Process Phases

- **Inception**
  - Very similar to the planning phase of a traditional SDLC approach.
  - This includes feasibility analysis that should answer questions such as the following: technical feasibility, economic feasibility, and organizational feasibility
- **Elaboration**: analysis and design workflows are the primary focus during this phase.
- **Construction**: focuses heavily on programming.
- **Transition**: testing and deployment workflows.

Engineering Workflows

- Business modeling
- Requirements
- Analysis
- Design
- Implementation
- Testing
- Deployment

Supporting Workflows

- Project management
- Configuration and change management
- Environment
- Operations and support*
- Infrastructure management*

* Part of the *enhanced* unified process

THE UNIFIED MODELING LANGUAGE

Unified Modeling Language

- Provides a common vocabulary of object-oriented terms and diagramming techniques rich enough to model any systems development project from analysis through implementation
- Version 2.0 has 14 diagrams in 2 major groups:
  - Structure diagrams
  - Behavior diagrams

UML Structure Diagrams

- Represent the data and static relationships in an information system
  - Class
  - Object
  - Package
  - Deployment
  - Component
  - Composite structure

UML Behavior Diagrams

- Depict the dynamic relationships among the instances or objects that represent the business information system
  - Activity
  - Sequence
  - Communication
  - Interaction overview
  - Timing
  - Behavior state machine
  - Protocol state machine,
  - Use-case diagrams

PROJECT TEAM ROLES AND SKILLS

Project Team Skills

- Project team members are change agents who find ways to improve their organization
- A broad range of skills is required, including
  - Technical
  - Business
  - Analytical
  - Interpersonal
  - Management
  - ethical

Project Team Roles

Role	Responsibilities
Business Analyst	Analyzing the key business aspects of the system Identifying how the system will provide business value Designing the new business processes and policies
Systems Analyst	Identifying how technology can improve business processes Designing the new business processes Designing the information system Ensuring the system conforms to IS standards
Infrastructure Analyst	Ensuring the system conforms to infrastructure standards Identifying infrastructure changes required by the system
Change Management Analyst	Developing and executing a change management plan Developing and executing a user training plan
Project Manager	Managing the team Developing and monitoring the project plan Assigning resources Serving as the primary point of contact for the project

Summary

- All systems development projects follow essentially the same process, called the system development life cycle (SDLC)
- System development methodologies are formalized approaches to implementing SDLCs
- Object-Oriented Systems Analysis and Design (OOSAD) uses a use-case-driven, architecture-centric, iterative, and incremental information systems development approach

Summary

- The Unified Process is a two-dimensional systems development process described with a set of phases and workflows
- The Unified Modeling Language, or UML, is a standard set of diagramming techniques
- The project team needs a variety of skills