Process Overview

COMP 3700
Software Modeling and Design

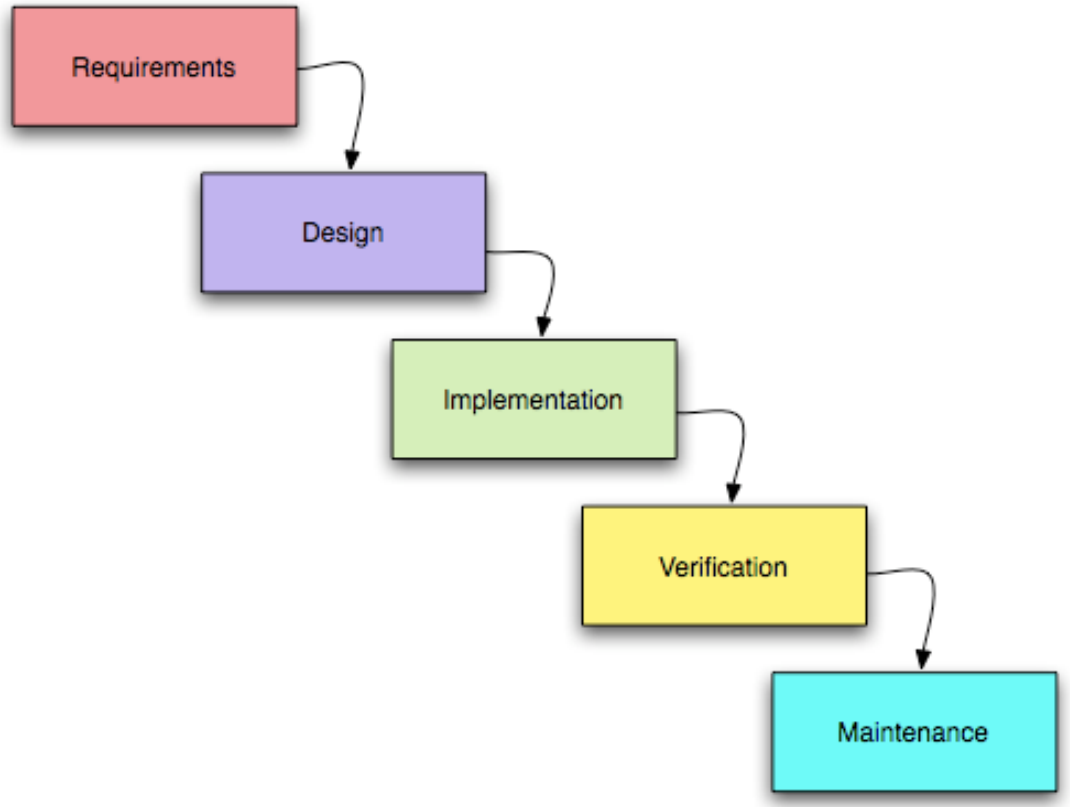
Shehenaz Shaik

Software Development: Stages

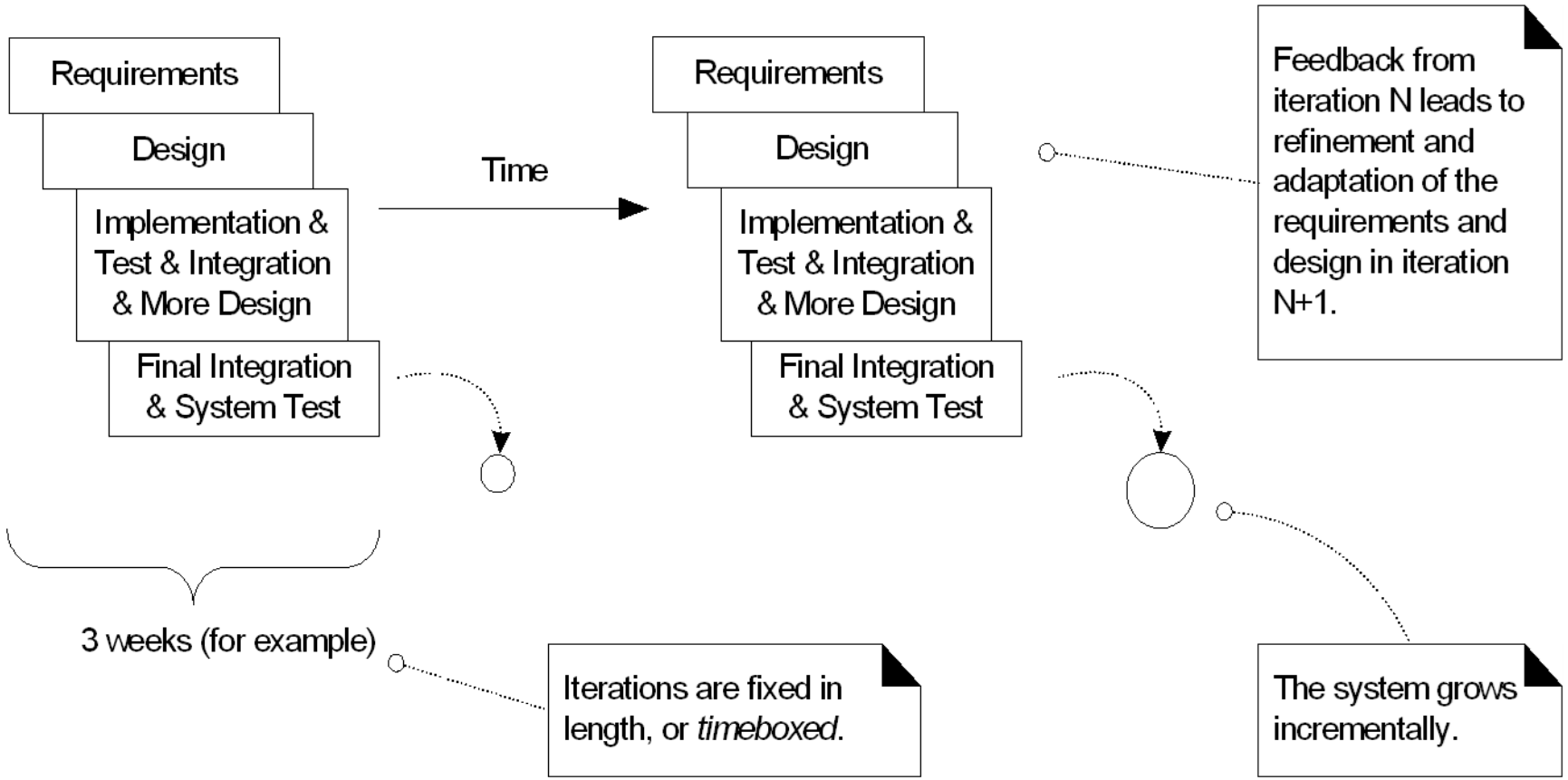
- System conception
- Analysis
- System design
- Class design
- Implementation
- Testing
- Training
- Deployment
- Maintenance

Waterfall approach

- Sequential approach
- Applicability
- Limitations
- High failure rate



Iterative approach



System Conception

COMP 3700
Software Modeling and Design

Shehenaz Shaik

System conception

- Deals with genesis of an application
- Knowledge of system
- Defer details and understand big picture

- Steps
  1. Devise a system concept
  2. Elaborate the concept
  3. Prepare problem statement

1. Devising a system concept

- Add new functionality
- Remove restrictions and generalize
- Simplification of tasks
- Automation of manual processes
- Integration of functionality
- Analogies from other problem domains
- Global cultural and business practices

2. Elaborating a concept

- Who / What / Where / When / Why / How

- Who is the application for?
- What problems will it solve?
- Where will it be used?
- When is it needed?
- Why is it needed?
- How will it work?

ATM Case Study: System concept

- Automated Teller Machine (ATM)
- System concept

> Develop software so that customers can access a bank's computers and carry out their own financial transactions without the mediation of a bank employee.

ATM Case Study: Elaborating the concept

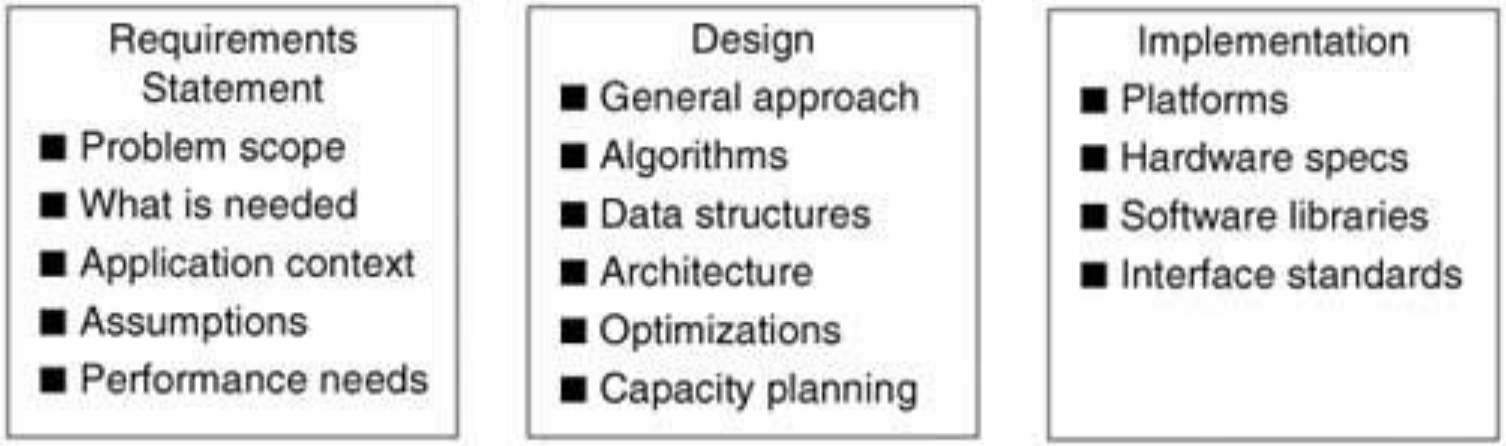
- Who is the application for?
- What problems will it solve?
- Where will it be used?
- When is it needed?
- Why is it needed?
- How will it work?

> Develop software so that customers can access a bank's computers and carry out their own financial transactions without the mediation of a bank employee.

3. Preparing a problem statement

- Requirements statement
  - Objectives
  - System behavior
  - Performance specifications
  - Software engineering standards
- Recommendations
  - Separate requirements from design decisions
  - Avoid describing system internals
  - Proof-of-concept implementation (optional)

3. Preparing a problem statement (Contd.)



- Problem statements may be Incomplete

ATM Case Study: Problem statement

- System concept

> Develop software so that customers can access a bank's computers and carry out their own financial transactions without the mediation of a bank employee.

ATM Case Study: Problem statement

> Develop software so that customers can access a bank's computers and carry out their own financial transactions without the mediation of a bank employee.

Design the software to support a computerized banking network including both human cashiers and automated teller machines (ATMs) to be shared by a consortium of banks. Each bank provides its own computer to maintain its own accounts and process transactions against them. Cashier stations are owned by individual banks and communicate directly with their own bank's computers. Human cashiers enter account and transaction data.

Automatic teller machines communicate with a central computer that clears transactions with the appropriate banks. An automatic teller machine accepts a cash card, interacts with the user, communicates with the central system to carry out the transaction, dispenses cash, and prints receipts. The system requires appropriate recordkeeping and security provisions. The system must handle concurrent accesses to the same account correctly.

The banks will provide their own software for their own computers; you are to design the software for the ATMs and the network. The cost of the shared system will be apportioned to the banks according to the number of customers with cash cards.

ATM Case Study: Representation