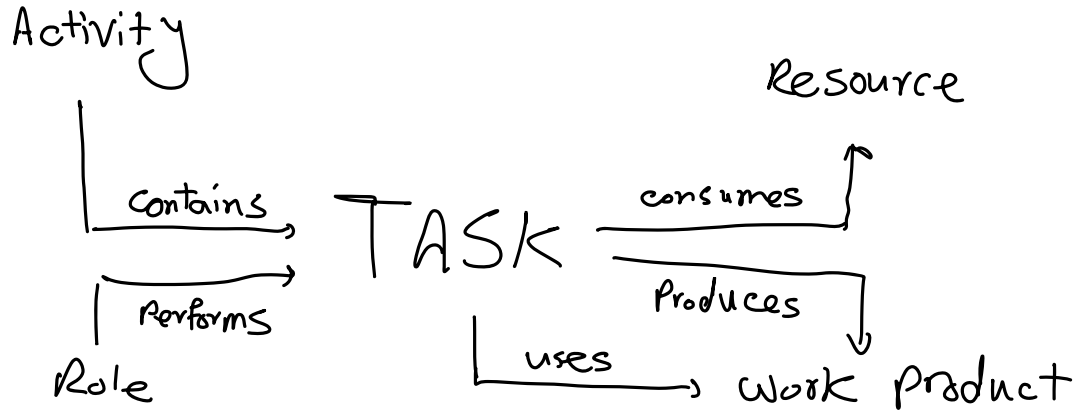


Software Processes and Agile Practices



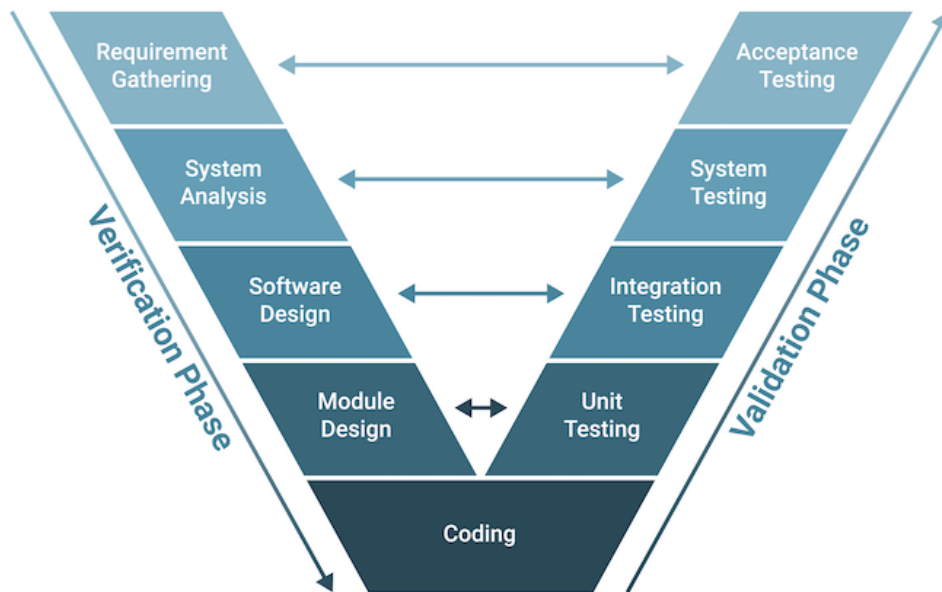
Software Engineering Activities

PROJECT MANAGEMENT PHASE	SPECIFICATION PHASE	DESIGN & IMPLEMENTATION PHASE	VERIFICATION & VALIDATION PHASE
Creating a Process	Identifying Ideas or Needs	Designing Architecture	Developing Test Procedures
Setting Standards	Eliciting Requirements	Designing Databases	Creating Tests
Managing Risks	Expressing Requirements	Designing Interfaces	Executing Tests
Performing Estimations	Prioritizing Requirements	Creating Executable Code	Reporting Evaluation Results
Allocating Resources	Analyzing Requirements	Integrating Functionality	Reviewing & Auditing
Making Measurements	Managing Requirements	Documenting	Client Demonstrations
Improving Process	Formulating Potential Approaches		Conducting Retrospectives

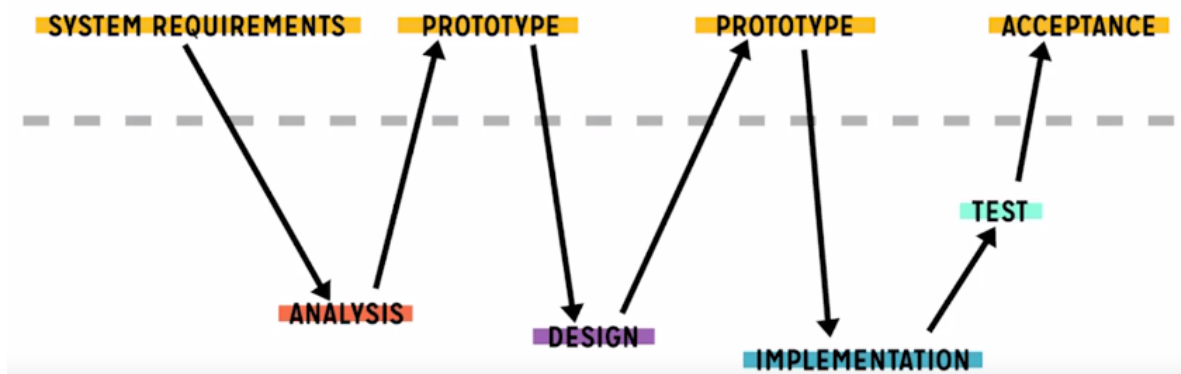
⚠ Linear Process models follow a pattern of phases completed one after another without repeating prior phases

↳ example : waterfall model

⚠ Unlike the waterfall model, V-model divides itself into two branches

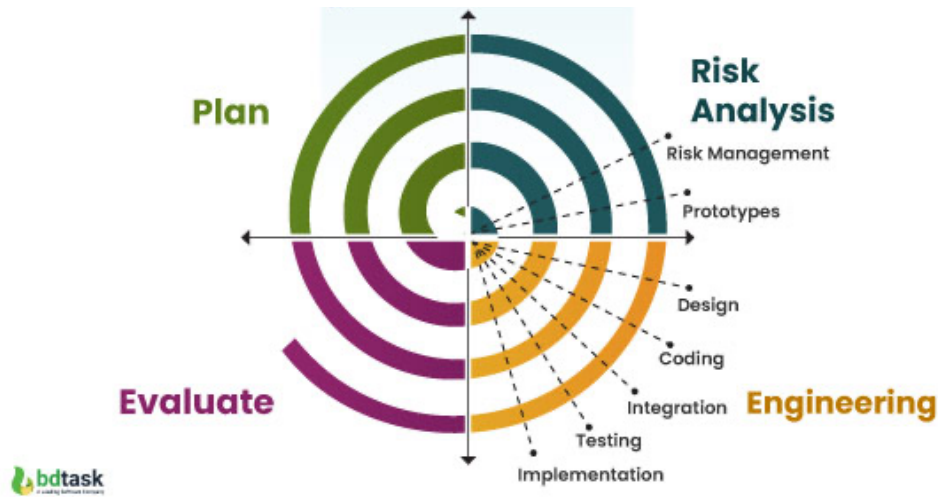


⚠ Sawtooth model gives you much needed client integration throughout the process

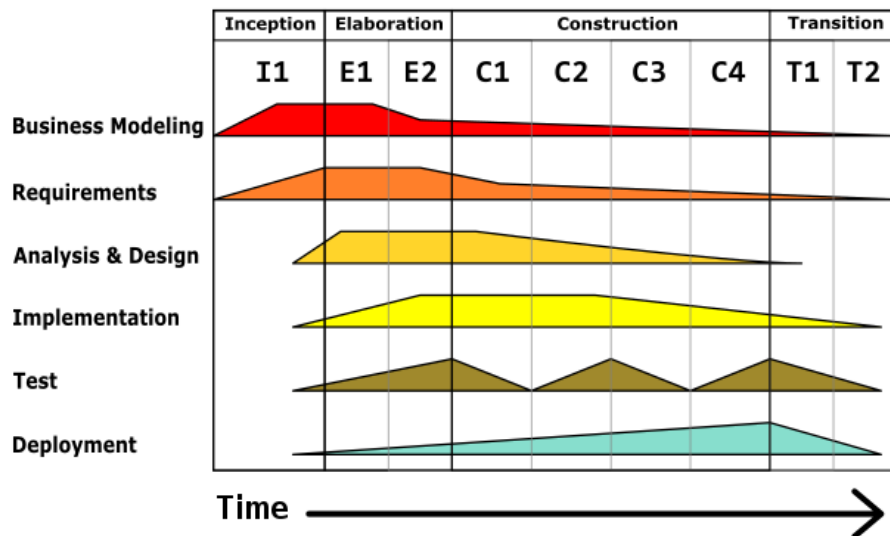


⚠ Iterative Software process models are ones which allow for repeating stages of the process

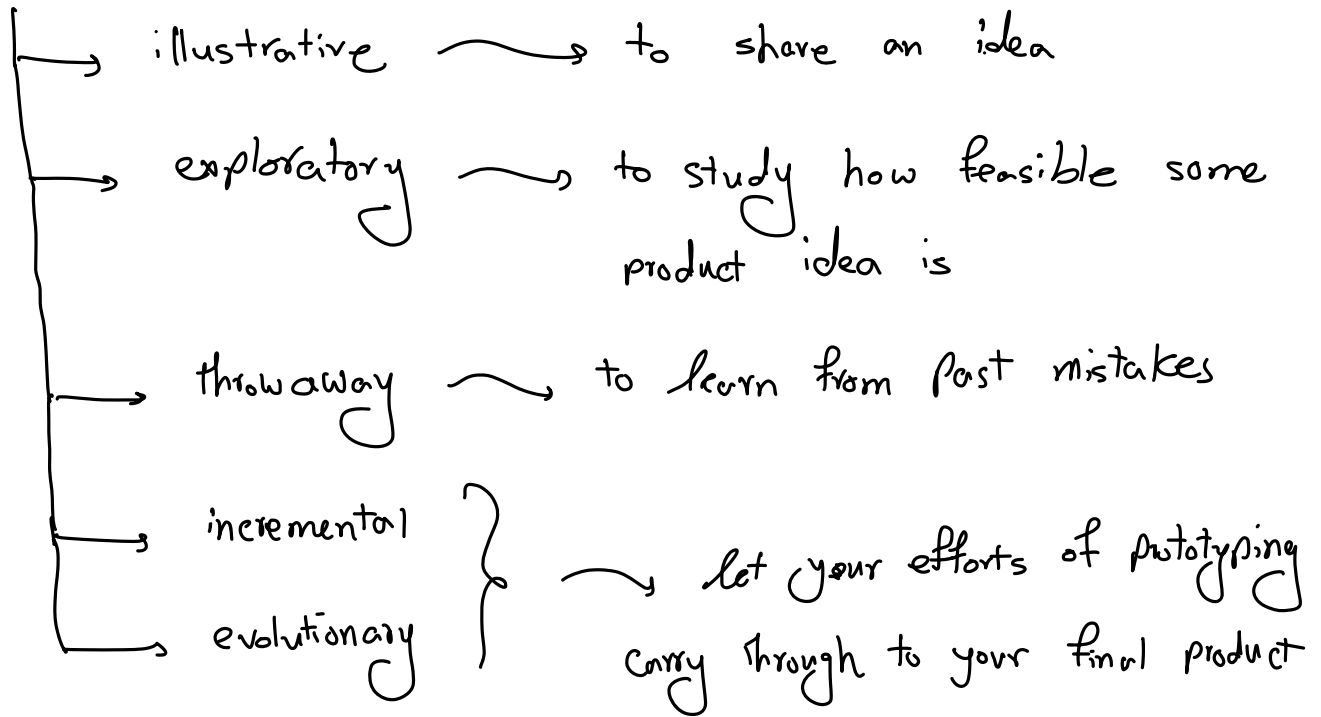
⚠ Spiral Model



⚠ Unified process is an iterative model of software development. It tries to emphasize gradual development as much as possible



⚠ Prototype types



⚠ Continuous delivery allows developers to deliver a product continuously as it's being developed. Whenever a developer commits a code change, it will be built, tested, integrated and released.

⚠ Linear Process models can't work well with Agile Practices

⚠ There are 5 aspects of development that extreme programming focuses on improving

- ① Communication
- ② Simplicity
- ③ Feedback
- ④ Respect
- ⑤ Courage

⚠ Extreme programming 12 principles

The Planning Game

Simple Design

Pair Programming

40 Hour Work Week

Small Releases

Continuous Testing

Collective Code Ownership

On-Site Customer

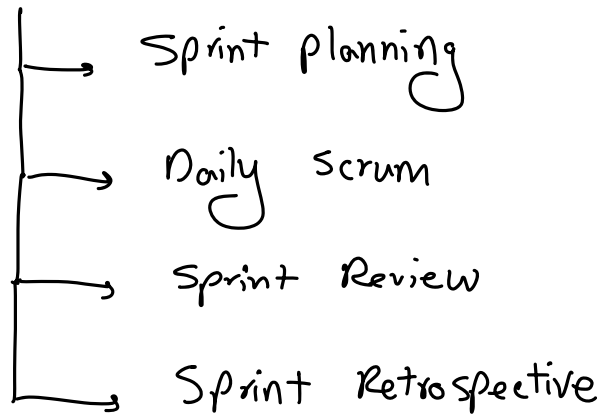
System Metaphor

Refactoring

Continuous Integration

Coding Standards

⚠ Scrum outlines four specific techniques for Inspection & Adaptation



⚠ Product Owner is just one person not a committee, but he/she can represent a committee

→ responsible for changes & prioritization in backlog

⚠ Scrum Master Duties to Product Owner

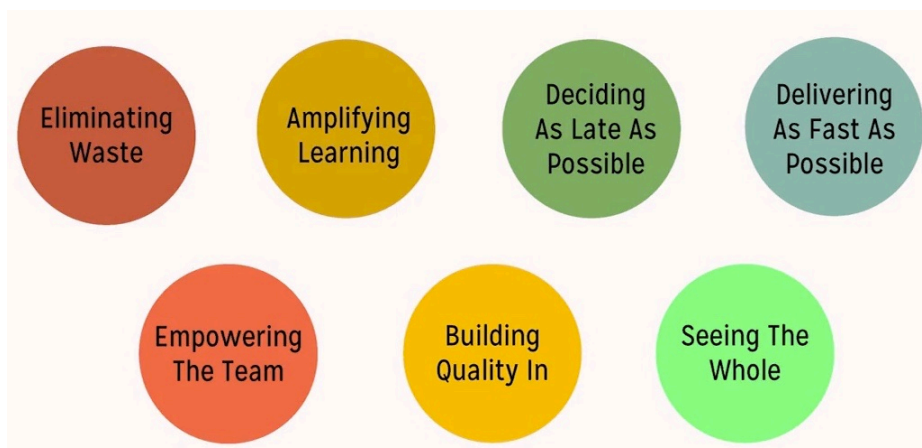
- Finding techniques to manage the backlog
- Helping the scrum team to understand the need for a clear & concise backlog
- Ensuring the product owner knows how to prioritize the backlog to get maximum value
- Facilitating scrum events

⚠ Scrum Master Duties to Development team

- coaching the team to self organize
- removing development roadblocks
- Facilitating scrum events

⚠ Agile UP combines Agile Practices with unified process

⚠ Lean Principles



⚠ Kanban helps Lean Processes to quantify the state of their system & achieve the ideals of Just-In-Time manufacturing

