Software Development Life Cycle (SDLC)

A mini-lecture series

CSE498 Collaborative Design (W) - Secure and Efficient C++ Software Development

02/12/2025

Kira Chan

https://cse.msu.edu/~chanken1/

Introductions

- Software Development Life Cycle (SDLC)
  - Sometimes called **Software Development Process**
- Describes the **process** in which a software is developed
- In small teams, you can kind of just agree on how to code up the software
- In bigger teams (1000+), there are a lot more challenges to nonstructured teams
- Different time zones, different languages, different culturesi

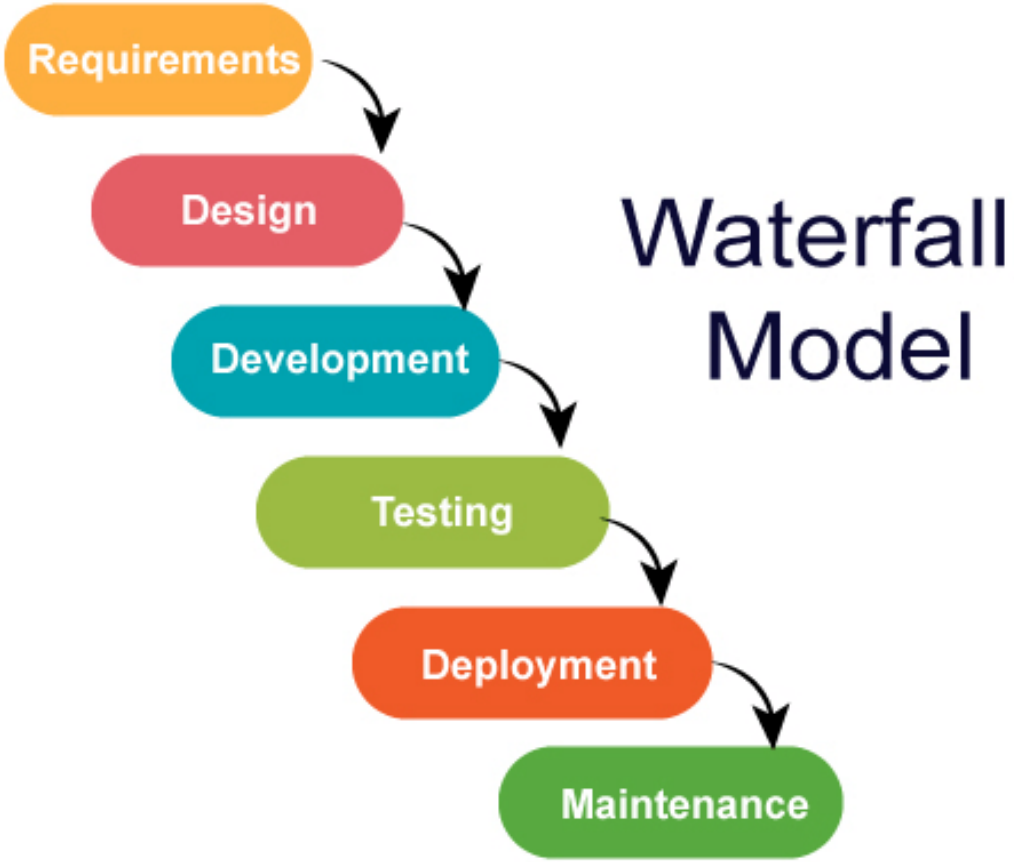
Motivation

- There is a need for structure in how teams develop the software together
- Let’s walk through the steps of creating a software system.

Life cycle

1. Requirements gathering (figure out what needs to be built)
2. Design (how will you build it?)
3. Coding (build the product!)
4. Testing (test the product!)
5. Deploying (ship the product.)
6. Maintenance (fix? Update the product if needed)

Oh, look we have a process!



https://medium.com/@chathmini96/waterfall-vs-agile-methodology-28001a9ca487

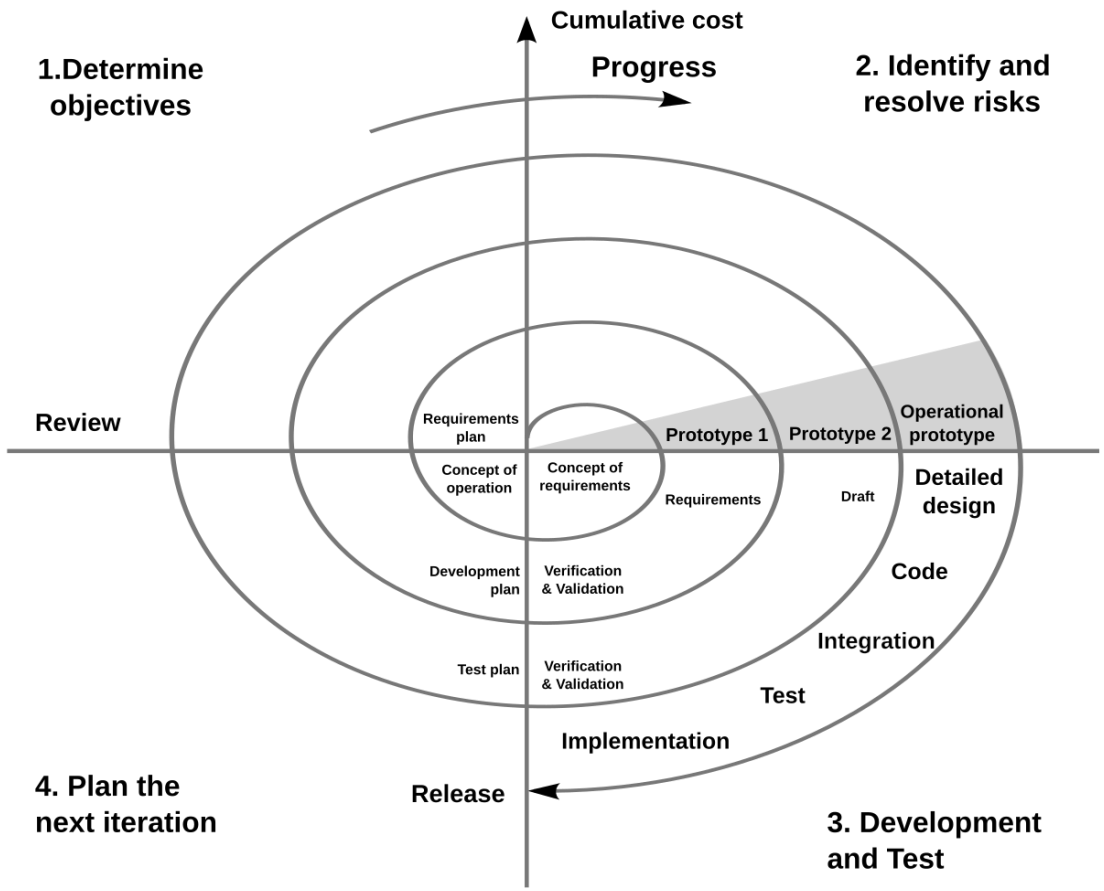
Process model

- Describes the way we can structurally develop a piece of software

Waterfall Model

- Pros:
  - Simple, concise, and details are clearly outlined
- Cons:
  - Rigid
  - Testing only comes towards the end

Spiral



Boehm, B

Agile

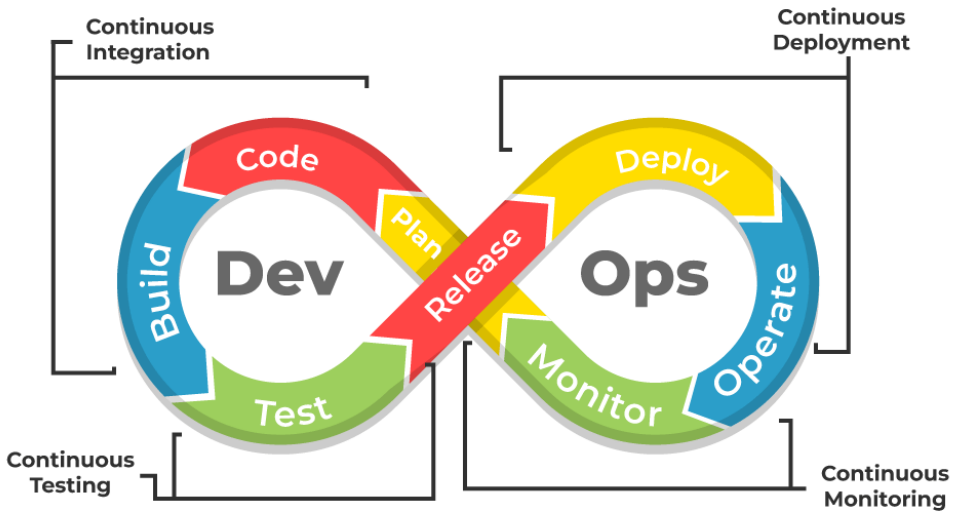
- Focuses on **flexibility, collaboration, and customer satisfaction**
- Quick iterable changes
- Focuses on developer creativity
- Often, the customer will literally have an office in the room
  - Pair Programming
  - Test Driven Development
  - Cross functional team
  - Daily Standup

Agile

- Pros
  - Fast delivery
  - Customer satisfaction
  - Changes in requirement is not that bad
- Cons
  - **Requires really good programmers and a high level of expertise**
  - Not really suitable for large projects that require planning and meticulous design
  - Difficult to estimate effort or time resources needed
  - Uncertainty and stress on developer
  - Developer burnout

DevOps

- Developments and operations
- Promotes the collaboration between development team and operations team
- Intention: increase speed to delivery



geeksforgeeks

Kanban

- Card Issue-based approach
- Outline the stages of your steps (features or tasks)
- Tackle each step one piece at a time
  - Think JIRA board or Trello board

Current model

- Mix-n-match of agile, devops, kanban, etc.
- Jira style
- “Agile but nobody really sticks to it, just get your tickets and do them” – friend from startup

Summary

- Process models intend to provide a structured approach to developing software
- You can mix and match processes
- Should figure out what works best for you and your team. There is no one size fit all answer
- Each process has pros and cons

Person of the Day
Ada Lovelace

- First ever computer programmer
- On her notes translating Charles Babbage's Analytical Engine, she added an algorithm that can compute Bernoulli numbers.
- She was the first to recognise that the machine had applications beyond pure calculation.