From Waterfall to Agile

- **Waterfall Model:** A linear, phase-driven software process. Each phase (requirements → design → implementation → testing → maintenance) is completed before the next begins. Changes are costly once a phase is finished, and testing is done only after development is complete.
- **Agile Methods:** An iterative, incremental approach that welcomes changing requirements even late in development. Work is done in short cycles (sprints) that produce working software frequently. Testing and integration happen continuously alongside development.
    - **2025 Stats:** 61% of organizations have used Agile for 5+ years
- **Comparison:** *Waterfall* emphasizes upfront planning and heavy documentation, suitable when requirements are well-understood and unlikely to change. *Agile* emphasizes adaptability and customer feedback, delivering value faster and adjusting as needs evolve.

Scrum Framework

- **Scrum** is an Agile framework for managing work in iterations (sprints). It breaks projects into small increments to be completed in fixed-length sprints (often 1–4 weeks). The focus is on continuous improvement and delivering potentially shippable software each sprint.
    - **2025 Stats:** 87% of Agile organizations use Scrum as their preferred framework
- **Roles:** In Scrum, team members fill one of three roles – **Product Owner**, **Scrum Master**, or **Development Team**. The Product Owner manages the backlog and priorities (representing the customer's interests). The Scrum Master facilitates the process and removes impediments. The Development Team (Scrum Team) is self-organizing and builds the product increment.
- **Events & Artifacts:** Key Scrum events include **Daily Stand-ups** (15-min daily team sync), **Sprint Planning**, **Sprint Review** (demonstrate increment), and **Sprint Retrospective** (reflect and improve). Scrum artifacts include the **Product Backlog**

DevOps Culture

- **DevOps** combines software **Dev**elopment and IT **Ops** (operations) to streamline the path from code to deployment. It is a set of practices and a culture that emphasize close collaboration between developers and operations teams.
- **Continuous Integration & Delivery:** DevOps teams use automation for building, testing, and deploying code (CI/CD pipelines). This enables rapid, reliable releases – code changes can go live faster and with fewer errors.
    - **2025 Trend:** 60% of businesses adopting GitOps for deployment automation (Forrester)
    - AI/ML integration in DevOps workflows for predictive monitoring and anomaly detection
- **Key Principles:** DevOps focuses on shared responsibility for outcomes, automation of repetitive tasks, and continuous feedback. By breaking down silos between dev and ops, DevOps achieves faster delivery **without sacrificing reliability**. Teams monitor