Module 2

17 November 2024 02:05

★ Evolution of Software Engineering

- Programming by Intuition
- Early Development (1950's - 1960's)
  - AD Hoc Development
  - Assembly Language
  - FORTRAN (1957) & COBOL (1959)
- Software Crisis
  - SDLC by Winston Royce in late 70's
- Further Development
  - OOP (Object Oriented Language) C++, Smalltalk
  - Software Engineering Discipline
  - Encapsulation, Inheritence, Polymorphism
- Agile Methodology
  - Agile Manifesto (2001) flexibility, collaboration & customer feedback
  - Adoption of DevOps practices
- Software Development at Peak
  - 2010 : microservice architecture allowed scalable, maintainable services
  - Cloud Computing
  - Continuous Delivery & Continuous Deployment (CI/CD)
- Current Trend
  - AI & ML
  - Low Code / No code Platforms
  - Quantum Computing

★ Software Engineering

Branch of computer science that deals with the design, development, testing

Branch of computer science that deals with the design, development, testing and maintenance of software applications on budget, time & within the required timeline.

★ Key Concepts

1. SDLC
2. Requirement Engineering process of defining, documenting & maintaining software requirements
3. Software Design
4. Programming & Implementation
5. Testing
6. Maintenance
7. Version Control (Managing changes to software codebase using Git, SVN, Mercurial)
8. Project Management
9. Agile Methodologies a set of principles for software development under which requirements & solutions evolve through collaborative effort of cross-functional teams; Scrum, Kanban & Extreme Programming (XP)
10. DevOps DEVelopment + IT OPerationS; aims to shorten the development lifecycle & provide continuous delivery with high software quality.
11. Software Quality Assurance (SQA) ensuring that software meets specified quality standards
12. Software Metrics measurements used to quantify various aspects of software development & maintenance; code complexity, coverage & defect density.
13. Security Engineering
14. Risk Management
15. Usability Engineering

★ Adoptions in Software Engineering -

- Standardization (ISO/IEC 12207)

- Education & Certifications (IEEE, ACM)
- Tooling & Automation (IDE's, CI/CD)
- Collaboration & Community (GitHub, Stack Overflow)

☆ Prerequisite for engineering approach to software development

- Understanding SDLC
- Project Management Skills (SPM)
- Proficiency in Programming Language & Tools
- Testing & Quality Assurance
- Business & Domain knowledge
- Team Collaboration & Agile Practices