

ch2

Software engineering is:

establishment and use of sound engineering principles in order to obtain economically that is reliable and works efficiently on real machines.

Some realities:

- understand the problem before a software solution is developed.
- software should exhibit high quality and maintainable.
- software should be maintainable.
- design becomes a pivotal activity.

A Layered Technology

1. tools
2. method
3. process model
4. quality focus

- The application of a **systematic, disciplined, quantifiable approach** to the **development, operation, and maintenance** of software; that is, the application of engineering to software.

A Process model:

1-Framework : (Framework activities - Umbrella Activities)

Framework activities :

- (1)Communication
- (2)Planning
- (3)Modeling (Analysis of requirements & Design)
- (4)Construction (Code generation & Testing)
- (5) Deployment

Umbrella Activities:

- tracking and control
- Risk management.
- Software quality assurance
- Technical reviews
- Measurement
- Reusability management

Adapting a Process Model:

- the overall flow of activities, actions, and tasks and the interdependencies among them.
- degree to which the customer and other stakeholders are involved with the project.
- degree to which team organization.
- degree to which work products are required

Essence of Practice in Software engineering : (Polya suggests)

1. Understand the problem (**communication and analysis**).
2. Plan a solution (**modeling and software design**).
3. Carry out the plan (**code generation**).
4. Examine the result for accuracy (**testing and quality assurance**).

Understand the Problem

- **Who has a stake in the solution to the problem?** That is, who are the stakeholders?
- **What are the unknowns?** What data, functions, and features are required to properly solve the
- **Can the problem be compartmentalized?** Is it possible to represent smaller problems that may be easier to understand?
- **Can the problem be represented graphically?** Can an analysis model be created?

Hooker's General Principles :

- **Every software project is precipitated by some business need :**
 1. The Reason It All Exists
 2. KISS (**Keep It Simple, Stupid!**)
 3. Maintain the Vision
 4. What You Produce, Others Will Consume
 5. Be Open to the Future
 6. Plan Ahead for Reuse
 7. Think!
 - the need to correct a defect in an existing application;
 - the need to the need to adapt a 'legacy system' to a changing business environment;
 - the need to extend the functions and features of an existing application, or
 - the need to create a new product, service, or system.