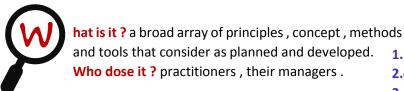
## **Principles that Guide Practice**



# **<u>Deployment Principles</u>**

- 1.managed Customer expectations
- 2.complete delivery package
  3.established A support regime
- 4.instructional materials
- 5.fixed first, delivered later

# Requirements models

1- Information domain

Behavioural domain

2- Functional domain

**Requirement Modeling principles** 

1. The information domain of a problem

3.The behaviour of the software must

4. The models that depict information,

5.The analysis task should move from

**Construction Principles** 

1. Understand (problem, basic design principles)

2.Pick a programming (language ,environment)

function, and behaviour must be

essential information toward

1. Coding principles (programming)

2. Testing principles (uncover effort)

implementation detail

**Preparation Principles** 

3.Create a set of unit tests

must be represented and understood

2. The functions that the software

performs must be defined

be represented

partitioned

<u>D</u>esign models

analysis models Characteristics of the software,

Customer requirements: construct effectively:

Modeling principles

- construct effectively :Architecture
  - Interface
  - Component detail

#### **Design Modeling principles**

- **1.Design** (traceable to the requirements model)
- 2.consider the architecture of the system to be built
- **3.Design of data** is as important as design of processing functions
- **4.Interfaces** (internal and external) designed with care
- **5.User interface** design (tuned to the needs -ease of use)
- **6.Component-level design** (functionally independent)
- **7.Components** (loosely coupled)
- **8.Design representations** (easily understandable )
- 9.The design (developed iteratively)

# **№** <u>V</u>alidation Principles

- **1.**Perform unit tests and correct errors you've uncovered.
- 2.Refactor the code



### Testing Principles

- 1.planned long before testing begins
- 2.Pareto principle
- 3.begin small to the large
- 4. Static testing can yield high results
- 5.Track defects uncovered by testing
- 6.Include test cases (correctly)

#### Principles that guide:



- 1. Be agile
- 2. Focus an quality
- 3. Be ready to adapt
- 4. Build an effective team
- **5. Establish mechanisms for** communication **and** coordination
- 6. Mange change
- 7. Assess risk
- 8. Provide value for others



- 1. Divide and Conquer
- 2. Under stand the use of abstraction
- 3. Strive for consistency
- 4. Focus on transfer of information
- 5. Build software that exhibits effective modularity
- **6. Look for patterns** ( help software developer to resolve problems)
- 7. Represent the problem and solution
- 8. That some one will maintain the software

### **C**ommunication Principles

- 1. Listen (focus on the word)
- 2. Prepare before you communication
- 3. Someone should facilitate the activity
- **4. Face to Face** communication is the best
- 5. Take notes and document decisions
- 6. Strive for collaboration
- 7. Focused, modularize your discussion
- 8. something is unclear, draw a picture
- 9. Move on ( agree , can't agree , unclear)

# $\sum$

# **A**gil Modeling Principles

- 1.software team (build software)
- 2.don't create more models than you need
- **3.Strive to produce** the simplest model
- **4.Build models** (amenable to change)
- 5. Explicit purpose for each model
- 6. Adapt the models you create to the
- 7.system at hand.
- 8.build useful models
- 9.Get feedback as soon as you can

# 1000

### **L**iving Modeling Principles

- **1.Stakeholder-centric** specific (stakeholders task)
- 2. Models and code (closely coupled)
- 3.Bidirectional information flow (models and code)
- 4. Model information (allow tracking system changes)
- 5. Assigned stakeholder rights and responsibilities
- 6.The states of various model elements (represented)

#### **D**lanning Principles

- 1. Understand the scope of the project
- 2. Involve the customer in the planning
- 3. Recognize that planning is iterative
- 4. Estimate based on what you know
- 5. Consider risk as you define the plan
- **7.Be realistic** (100%)
- 8. Adjust granularity as you define the plan
- 9. Define how you intend to ensure quality
- 10.Track the plan

#### 6.Describe how you intend to accommodate change