Concept：

1. Design and analysis:

Design: emphasizes a conceptual solution that fulfills the requirements, rather than its implementation.

Analysis: emphasizes an investigation of the problems and requirements, rather than a solution

1. OOD principle:

Abstraction, encapsulation, polymorphism, Hierarchy

1. UML

Visual language for specifying, constructing and documenting the artifacts of systems.

1. UP

Popular iterative software development process for building object-oriented systems.

1. Software development process

Software development process describes an approach to building, deploying, and possibly maintaining software.

1. Domain model

A visual representation of conceptual classes or real-situation objects in a domain.

1. Main phases of UP

Inception: approximate vision, business case, scope, vague estimates

Elaboration: refined vision, iterative implementation of the core architecture

Construction: iterative implementation of remaining lower risk and easier elements, and preparation for development.

Transition: beta tests, deployment

1. Iterations

Development is organized into a series of short, fixed-length mini-projects called iterations.

1. Requirement

FURPS+

Functional

Usability

Reliability

Performance

Supportability

1. Agile method

Short timeboxed iterations with evolutionary refinement of plans, requirements, and design is a basic practice the methods.

1. Disciplines and Phases

Disciplines: a set of activities in one subject.

Artifact: the general term for any work product.

1. Use case

Text stories of some actor using a system to meet goal.(a collection of related success and failure scenarios that describe an actor using a system to support the goal)

Actor: something with behavior

Scenario: specific sequence of actions and interactions between actors and the system. (Use case Instance)

1. Use case model

A set of all written use cases and a model of system’s functionality and environment.

1. Use case relationship

Extend

Include

Generalization

1. Use case realization: how particular use case is realized within the design model in terms of collaborating objects.
2. Architecture

Is the fundamental organization of a system.

1. 4+1 view

Logical View

Physical View

Process View

Deployment View

Use case View

1. GRASP
2. Design priciples:

Decomposition

Abstract

Separation of concerns

Information hiding and localization

Modularity

SOLID Single Principle

Open/Closed Principle

Liskov Substitution Principle

Interface Segregation

Dependency Inversion

1. GoF

Adaptor

Factory Method

Observer

Strategy

Façade

Singleton

Composite

1. Relationship of class

Association

Aggregation: is a part of

Generalization: is a kind of

1. Diagrams

SSD : system operation





