

List down golden rules of user interface design.

Strive for Consistency:

- Keep interface elements like color, fonts, layouts, and interaction patterns consistent throughout.

Cater to Universal Usability:

- Speed up interaction for experienced users by providing keyboard shortcuts, swipe gestures, or command palettes.

Offer Informative Feedback:

- The system must acknowledge every user action with immediate feedback, like loading spinners, success/error messages, or progress bars.

Design Dialogs to Yield Closure:

- Group actions into meaningful sequences with clear beginnings, middles, and ends to provide satisfying completion.

Prevent Errors:

- Proactively prevent errors through input validations, constraints, and confirmations for destructive actions.

Permit Easy Reversal of Actions:

- Provide readily accessible undo, redo, or cancel options to let users explore without fear of making mistakes.

Keep Users in Control:

- Users should feel in charge, not the system; avoid unexpected pop-ups, auto-navigation, or forced actions.

Reduce Short-Term Memory Load:

- Minimize cognitive burden by making information, instructions, and options visibly available rather than relying on recall.

What is feasibility study? What are the types of feasibility study?

Feasibility Study is an evaluation process that analyses whether a proposed software project is practical and viable before actual development begins.

Types of Feasibility Study:

- Technical Feasibility - Assesses available hardware, software, and technical resources
- Economic Feasibility - Analyses cost-benefit ratio and financial viability
- Operational Feasibility - Evaluates ease of operation and maintenance post-deployment
- Legal Feasibility - Investigates compliance with legal requirements and regulations
- Schedule Feasibility - Determines if the project can be completed within given timeframes

Write short note on CMM levels.

The Capability Maturity Model (CMM) is a framework developed to improve and assess the processes of a software development organization. It defines a five-level evolutionary path.

- Initial (Level 1): Processes are unpredictable and poorly controlled. Success depends on individual heroics.
- Repeatable (Level 2): Basic project management processes are established. Success can be repeated for similar projects.
- Defined (Level 3): Processes are standardized, documented, and integrated into an organization-wide standard.
- Managed (Level 4): Management Processes are measured and controlled using detailed metrics.
- Optimizing (Level 5): The focus is on continuous process improvement through quantitative feedback and innovative ideas.

The goal of the CMM is to provides a framework for organizations to assess their current process maturity and identify areas for improvement to achieve higher quality software development.

Differentiate between Prescriptive and Evolutionary models.

Aspect	Prescriptive Models	Evolutionary Models
Approach	Sequential, predefined phases	Iterative and incremental development
Requirements	Requirements fully defined upfront	Requirements evolve over iterations
Flexibility	Less flexible to changes	Highly adaptable to changing requirements
Examples	Waterfall, V-Model	Spiral, Prototyping
Feedback	Limited early feedback	Continuous user feedback
Risk	Higher risk due to late testing	Lower risk through iterative refinement

Describe advantages and limitations of large sized software projects.

Advantages:

- Enhanced productivity through specialized teams
- Ability to handle complex, large-scale problems
- Faster development through parallel work streams
- Comprehensive testing capabilities

Limitations:

- Communication Challenges arise due to complex coordination between team members
- Difficulty in maintaining accountability
- Increased Complexity causes higher management overhead
- Cost Overruns are likely as budget management becomes difficult
- Schedule Delays occur frequently due to coordination delays as they affect timelines

Describe the characteristics and nature of software and explain the layered structure of software engineering.

Software Characteristics:

- Intangible - Cannot be physically touched
- Logical - Consists of instructions and algorithms
- Complex - Involves intricate relationships between components
- Maintainable - Requires continuous updates and modifications

Layered Structure of Software Engineering:

Tools
Methods
Processes
Quality Focus

Four-Layered Technology:

- Quality Focus (Foundation) - Continuous process improvement, integrity, maintainability, and usability
- Process Layer - Framework binding all layers; defines activities, actions, and tasks for software development
- Methods Layer - Technical approaches including communication, planning, modelling, construction, and deployment
- Tools Layer - Automated support for process and methods including CASE tools and development environments

Write short note on Scrum.

Scrum is an agile framework for managing software development projects that emphasizes teamwork, iterative development, and continuous improvement.

Scrum Roles:

- Scrum Master - Facilitates and oversees the process and removes impediments
- Product Owner - Defines requirements and priorities
- Development Team - Self-organizing team that delivers the product

Scrum Events:

- Sprint - Time-boxed iteration (2-4 weeks)
- Sprint Planning - Plan work for upcoming sprint
- Daily Scrum - 15-minute daily meeting
- Sprint Review - Demonstrate completed work
- Sprint Retrospective - Reflect and improve processes

Analysis Model Elements:

Four Primary Elements:

1. Scenario-based Elements

- Comprised of use cases and user stories, represent system from the user's perspective

2. Class-based Elements

- Defines objects, attributes, and relationships, class diagrams and collaboration diagrams

3. Behavioural Elements

- Consists of State diagrams to show system state changes, sequences and event responses

4. Flow-oriented Elements

- Consists of Data Flow Diagrams (DFD) and Control Flow Diagrams (CFD), to show data transformation through system

Core Components:

- Data Dictionary - Repository of all data objects
- Entity Relationship Diagram (ERD) - Depicts data relationships
- Process Specifications (PSPEC) - Describes function details

Differentiate between Agile and Evolutionary models.

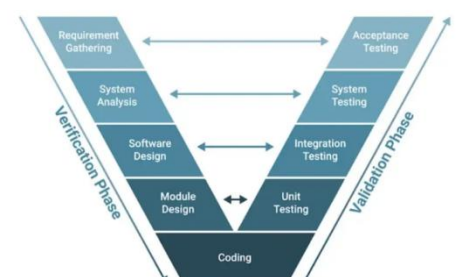
Aspect	Agile Models	Evolutionary Models
Philosophy	Values and principles focused	Iterative refinement focused
Time Management	Fixed timed-box iterations	Variable iterations
Customer Involvement	Continuous	Periodic
Documentation	Less	Moderate
Team Structure	Self-organizing, cross-functional	Hierarchical

What is V model? Draw its diagram.

V-Model is a sequential software development model that extends the waterfall model by associating each development phase with a corresponding testing phase.

Key Characteristics:

- Also known as Verification and Validation Model
- Left side represents Verification phases (development)
- Right side represents Validation phases (testing)
- Coding phase connects both sides at the bottom



What are 3Ps in Software Project spectrum?

People:

- This refers to the human resources involved in the project, including the development team, project managers, testers, designers, and stakeholders.

Process:

- This defines the set of practices, methodologies, and tools used to manage the project. Examples include Agile, Scrum, Waterfall, DevOps, and the specific steps for development, testing, and deployment.

Product:

- This encompasses the project's goals, features, functionalities, and requirements of what is to be delivered. It defines the scope of the final software product.