

Information System Development and Management:

1. Business Process
2. Business Model
3. Various Degrees of Organizational Change
 1. *Incremental*
 2. *Far-Reaching*
4. Four kinds of Structural Organizational Change
 1. **Automation**
 2. Rationalization of Procedures
 3. **Business Process Reengineering (BPR)**
 - i. Not an Incremental Improvements
 - ii. Identifying Business Processes that need Improvement
 1. Strategic Analysis
 2. Pain Points
 - iii. After Identifying a Process to be improved identify the following
 1. Inputs and Outputs
 2. Flow of Products and Services
 3. Network of Activities and Buffers in the Process
 4. Identify all Resources
 5. Information Structure and Flow
 6. Process Owners
 7. Process Actors and Decision Makers
5. **Digital Transformation**
6. Workflow Management and Document Management
7. **Change Management**
 1. **Change Agent**
 2. *Force Field Analysis* (*Analysis of forces that are for and against the change*)
8. Business Process Management (BPM)
9. Quality
 1. Right Quality (Generally Higher Quality is the Right Quality)
 2. Quality Control
 3. Quality Assurance
 4. **Total Quality Management**
 - i. Six Sigma
 1. 3.4 defects per million opportunities
 - ii. Incremental Improvements
10. Benchmarking
11. Best Practices
12. **Systems Development**
 1. **Systems Analysis**
 - i. **Two of the approaches in Systems Analysis**
 1. **Critical Success Factors**
 2. **Enterprise Analysis / Business Systems Planning**
 - ii. Requirement Identification

iii. Feasibility Study

1. PESTEL

- a. Political
- b. Economical
- c. Social
- d. Technical
- e. Environmental
- f. Legal

2. Systems Design

- i. Design Specifications
- ii. Software Designing Methods
 - 1. Structured Methodologies
 - a. Structure Charts
 - 2. Object Oriented Design
 - 3. Aspect Oriented Design
 - 4. Component Based Design
 - 5. Prototyping
 - 6. Joint Application Design (JAD) *{Enhances user participation}*

3. Programming/Implementation

4. Testing

i. Different Levels of Testing

- 1. Unit Testing / Program Testing
- 2. System Testing
- 3. Acceptance Testing / User Acceptance Testing (UAT)
- 4. Other terms used
 - a. Regression Testing (Testing whether previous functionalities are working after enhancements unless purposely modified or removed)
 - b. Black Box Testing
 - c. White-Box Testing
 - d. Test Design, Test Execution
 - e. Integration Test
 - f. Compatibility Testing
 - g. Exploratory testing (Exploratory testing is a hands-on approach in which testers are involved in minimum planning and maximum test execution)
 - h. Smoke Test (non-exhaustive set of tests that aim at ensuring that the most important functions work. The results of this testing is used to decide if a build is stable enough to proceed with further testing)

ii. Testing Purpose

- 1. Right Functionality
- 2. Right Performance
- 3. Robustness
 - a. Load Testing
 - b. System Crashing
 - c. Right Security
- 4. Adhering to the standards
- 5. Right User-friendliness

iii. Ice-Berg problem

- iv. Test Plan
 - v. Test Automation
 - 1. Test Automation Pyramid
- 5. Conversion/Installation
 - i. Conversion Strategies
 - 1. Parallel Strategy
 - 2. Direct Cutover
 - 3. Pilot Study
 - 4. Phased Approach
 - ii. Points to Note
 - 1. Training
 - 2. Documentation
 - 3. Updating the Database with Basic Data
 - 4. Updating the Database with Up-to-date information.
- 6. Production and Maintenance
 - i. Production
 - ii. Maintenance
 - 1. Preventive Maintenance
 - 2. Corrective Maintenance
 - 3. Perfective Maintenance
 - iii. *SLA (Service Level Agreements)*
- 13. Tools for Modeling and Designing Systems
 - 1. Data Flow Diagram
 - 2. UML (Unified Modeling Language)
 - i. Activity Diagram
 - ii. Use Case Diagram
 - iii. Class Diagram
 - 1. Entity Relationship Diagram (ER Diagram)
 - 2. Structure Charts
 - 3. CASE Tools (Computer Aided Software Engineering Tools)
 - 4. Process Model Diagrams
 - 5. ...
- 14. System Life Cycle
- 15. Software Development Methodologies
 - 1. Waterfall model
 - 2. Cowboy Coding
 - 3. Pair-programming
 - 4. Rapid Application Development (RAD)
 - 5. Dynamic Systems Development Method (DSDM)
 - 6. *Rational Unified Process (RUP)*
 - 7. *Cleanroom Software Engineering*
 - 8. Iterative processes
 - 9. Distributed Development
 - 10. Virtual Teams (Geographically Dispersed Team)
 - 11. Lean (also known as Lean Manufacturing)

{Derived from Toyota Production System}

{ Basically, lean is centered on preserving value with less work}

12. **Test Driven Development (TDD)**

i. Steps:

1. *Start with the Problem*
2. *Give the simplest solution (without you getting **Paralyzed with Analysis**) and get it to work.*
3. *Improve your quick & dirty solution with a better design.*

13. Extreme Programming (XP)

14. **Agile Methodologies**

i. *Agile Manifesto (below, while there is value in the items on the right, we value the items on the left more)*

1. *Individuals and interactions over processes and tools*
2. *Working software over comprehensive documentation*
3. *Customer collaboration over contract negotiation*
4. *Responding to change over following a plan*

ii. Scrum (also an Agile methodology)

1. *Product backlog*
 - a. *User Stories*
 - i. *Tasks*
 - ii. *Story Points*
2. *Sprints (Also known as Iterations; sprint is a period of 2-6 weeks)*
 - a. *Sprint Planning Meetings*
 - b. *Daily Stand up meetings*
 - c. *Burn-down Charts*
 - d. *Sprint Demos*
 - e. *Retrospectives (also a meeting)*
 - f. *Sprint Velocity*
 - g. *Planning Poker*
3. *Team empowerment*

15. **Scaled Agile**

16. **DevOps** *(Developers and Operations Teams - It is a culture where Developers, System Administrators and Quality Assurers work in close collaboration in order to make frequent deliveries)*

1. **ALM** *(Applications Lifecycle Management)*

16. Customization

17. **Outsourcing** *(delegating to an independent outside entity (or company))*

18. **Insourcing** *(delegating to an independent internal entity (or company) within the same organization)*

19. **Subcontracting**

20. Request for Proposal (RFP)

21. Request for Information (RFI)

22. Request for Change (RFC)

23. Information Lifecycle Management (ILM)

24. **Blockchain**

1. **Bitcoin** *(a Digital Currency)*

25. Stakeholder Matrix (Interest vs Power quadrants)