**MODULE 5**

**Analysis** **Process** - systematic approach used to understand, evaluate, and interpret information

**Data Flow Diagrams (DFDs)** – visually represent the flow of data within a system.

**Data Dictionaries** - creating a structured repository

**Process Specifications And Structured Decision-Making Techniques** - clearly defining the logic and rules

**HIPO (Hierarchy plus Input-Process-Output)** - represent the functional requirements of a system

**Problem Analysis Tools** - like Pareto Analysis and Fishbone Diagrams (also known as Ishikawa diagrams)

**Pareto** **Analysis** - based on the Pareto Principle (80/20 rule)

**Fishbone Diagram** - identify the root causes of a problem.

**Entity-Relationship Diagram (ERD)** - crucial step in database design

**Designing Forms and Reports** - crucial aspect of user interface design

**Forms** - structured document or interface used to collect data from users.

Designing Forms:

1. **Purpose and Clarity** - Clearly define the purpose of the form

2. **Logical Layout** - Organize form fields in a logical order

3. **Field Design** - Use appropriate input controls for each type of data

4. **Validation and Feedback** - Implement real-time validation

5. **Accessibility** - Ensure that forms are accessible to all users

6. **Minimize User Input** - Reduce the amount of information

Components:

**Input Fields**: Areas where users can enter data.

**Labels**: Descriptions that indicate what information is required in each field.

**Instructions**: Guidance on how to fill out the form, if necessary.

**Buttons**: Options for submitting the form or resetting it.

Forms Format:

**Layout > Field Design > Instructions > Visual Elements > Submission Options**

**Report** - structured document that presents information, data, and analysis

Designing Reports

1. **Purpose and Audience** - Clearly define the purpose of the report and the target audience

2. **Structure and Organization** - Organize the report in a logical structure

3. **Visual Elements** - Use visual elements such as charts, graphs

4. **Clarity and Conciseness** - the language used in reports is clear and concise

5. **Consistency** - Maintain consistency in formatting

6. **Interactivity** - consider incorporating interactive elements into reports

Reports Format:

**Structure > Sections > Visual Aids > Formatting > Conclusion and Recommendations > Appendices and References**

**Assessing usability** - critical process in the design and evaluation of systems

Key Components of Usability

**1. Effectiveness:**

**Task Success Rate:** Measure the percentage of users who successfully complete a given task.

**Error Rate:** Track the number of errors users make while attempting to complete tasks.

**2. Efficiency:**

**Time on Task**: Record the time it takes for users to complete specific tasks

**Resource Utilization**: Assess the amount of effort required to complete tasks.

**3. Satisfaction:**

**User Satisfaction Surveys:** Use questionnaires to gather user feedback on their experience

**Net Promoter Score (NPS):** Measure the likelihood of users

**4. Learnability:**

**Time to Learn:** Evaluate how long it takes for new users to become proficient with the system.

**Retention:** Assess how well users retain knowledge and skills over time.

**5. Accessibility:**

**Inclusivity:** Ensure that the system is usable by people with varying abilities and disabilities

**System implementation** - where the designed system is put into operation.

**Coding** - where the actual source code for the system is written. This phase translates the design specifications into a functional software application.

**Testing** - software system meets its requirements and functions correctly.

key types of testing:

**Unit testing** - involves testing individual components or modules

**Integration testing** - focuses on verifying the interactions.

**System testing** - process of testing the complete and integrated software system

**Installation** - where the completed software system is deployed into a production environment.

**Direct installation** - also known as "big bang" installation

**Parallel installation** - running both the old and new systems

**Phased installation** - implementing the new system in stages

**Documentation** - provides detailed information about the system, its functionality, and how to use

**System Documentation** - describes the architecture, components, and functionalities

**User documentation** - help end-users understand and effectively use the software system.

* It provides instructions, guidelines, and support materials tailored to the needs of the users.

**User training** - educating end-users on how to use the new system effectively.

**User support** - ongoing assistance provided to users

**System maintenance** - continued functionality, performance, and security of a software system

Types of System Maintenance:

**Corrective Maintenance** - fixing defects or bugs that are identified after the system has been deployed.

**Adaptive Maintenance** - modifying the system to accommodate changes

**Perfective Maintenance** - focuses on improving the system's performance

**Preventive maintenance** - to prevent potential issues before they occur.

Factors Influencing Maintenance Costs:

**1. System Complexity** - More complex systems require more resources

**2. Frequency of Changes** - require frequent updates or changes

**3. User Base Size** - A larger user base can lead to increased support requests

**4. Technology Stack** - The choice of programming languages, frameworks, and tools

**5. Regulatory Compliance** - Systems that must comply with industry