GUJARAT TECHNOLOGICAL UNIVERSITY

DIPLOMA IN COMPUTER ENGINEERING

Semester: 4

Subject Name System analysis and Design

Sr. No.	Course content
1.	**1. Introduction** 1.1. The Software Engineering Discipline - evolution and Impact 1.2. Programs vs. Software Products 1.3. Software process and software engineering methods 1.4. Attributes of good software Professional and ethical responsibility
2.	**2. System Development Life Cycle** 2.1 Introduction to system development life cycle (SDLC). 2.2 Various phases- 2.2.1 Study 2.2.2 Analysis 2.2.3 Design 2.2.4 Development 2.2.5 Implementation and Maintenance 2.3 System Documentation considerations. 2.3.1 Principles of system Documentation 2.3.2 Types of documentation and their importance. 2.3.3 Enforcing documentation discipline in an Organization.
3.	**3. Software Life cycle Models** 3.1. Introduction of Life Cycle Model 3.2. Life Cycle Models 3.2.1. CLASSICAL WATERFALL MODEL 3.2.2. ITERATIVE WATERFALL MODEL 3.2.3. PROTOTYPE MODEL 3.2.4. EVOLUTIONARY MODEL 3.2.5. SPIRAL MODEL 3.3. Comparison between different life cycle models
4.	**4. System requirement specification and System Planning** 4.1. Requirements gathering and analysis 4.2. Software requirement specification (SRS) 4.2.1. Content of SRS Document 4.2.2. Functional requirement 4.2.3. Traceability 4.2.4. Characteristics of good SRS Document 4.2.5. Techniques of representing Complex Logic 4.3. Formal System Development Techniques 4.3.1. What is a formal technique? 4.3.2. Model vs. Property oriented methods

	4.3.3. Operational Semantics 4.3.4. Merits and Limitations of Formal Methods 4.4. Data and fact gathering techniques. 4.4.1. Interviews 4.4.2. Group Communication 4.4.3. Presentation 4.4.4. Site visits. 4.5. Feasibility study and its importance
5.	**5. System Design and Modeling** 5.1. Process modeling 5.2. Logical and Physical Design. 5.3. Design representation. 5.4. System Flowchart and structured charts. 5.5. Data Flow Diagram. 5.6. Common diagramming concentration and guidelines using DFD ERD for data modeling and system Analysis.
6.	**6. Object Modeling using UML AND User Interface Design** 6.1. Use Case Model 6.1.1. Actors 6.1.2. Use cases 6.1.3. Use case diagram 6.2. Sequence Models 6.3. Activity Models 6.3.1. Activities 6.3.2. Branches 6.3.3. Initiation and termination 6.3.4. Concurrent Activities 6.3.5. Executable Activity Diagram 6.4. Classification of forms. 6.5. Input/Output forms design. 6.6. User-interface Design. 6.7. Graphical Interface.
7.	**7. Modular and structured Design** 7.1. Module Specification. 7.2. Module Coupling and cohesion. 7.3. Top-Down and bottom-up design.
8.	**8. System Implementation and Maintenance** 8.1. Planning Consideration. 8.2. Conversion methods, Procedure and controls. 8.3. System Acceptance criteria. 8.4. System evaluation and performance. 8.5. Testing and validation. 8.6. System Quality control and Assurance. 8.7. Maintenance activities and issues

LABORATORY EXPERIENCES :

Case studies

1. Pay roll
2. Customer order processing & invoicing.
3. Account Receivable.
4. Account Payable.
5. Financial Accounting System.
6. Inventory Management.
7. Purchase system.
8. Library Automation.
9. Student registration system.
10. Vehicle registration system.
11. Laboratory system.

A standard layout is to be adopted for all Case studies

1. Key Information needs.
   1.1 Analysis of Existing or New System.
   1.2 Create Manual Reports.
2. Transaction processing and management control.
3. Report design and data collection methods, routing, frequency etc:-
   3.1 Input
   3.2 Output
   3.3 Control Reports.
4. Data Flow Diagram (DFD) of System.
5. Computer System Design:-
   5.1 Master Files.
   5.2 Transaction Files.
   5.3 Checks & Controls.

Reference Books:

1. Structured system analysis and design. Madhulika Jain
2. Fundamentals of Software Engineering , second edition, Rajib Mall PHI
3. Object Oriented Modeling and design with UML, second edition, By Michael R Blaha and James R Rambaugh
4. Software engineering for students forth ed. Dougless Bell Pearson
5. System Analysis and Design , Elias M. Awad, GP.
6. Software engg. Deepak Jain Oxford