GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD, GUJARAT
COURSE CURRICULUM

Course Title: Fundamentals of Software Designing
(Code: 3340703)

Diploma Programmes in which this course is offered	Semester in which offered
Computer Engineering	**4th Sem**

1. RATIONALE

The students of this course will be able to describe evaluation of software and they can design the software with various life cycle models with scheduling. Through requirement analysis and specification phase, client requirements will be easily understood and systematic organization of requirements will be kept in specification document. It gives systematic and cost effective techniques to software development. It also gives the modern approach - Object Oriented Analysis from the requirement specifications. It is needed to develop high quality software. Hence students will be able to apply these concepts during project development in forthcoming semester.

2. COMPETENCY

The course content should be taught and implemented with the aim to develop different types of skills so that students are able to acquire following competency:

- Gather & Analyze data, design user interface, prepare system in terms of natural objects

3. Course Outcomes:

- Describe the Software Development Life Cycle (SDLC).
- Distinguish various Software Process Models (Approach of Software Development).
- Analyze gather and prepare Software Requirement Specification for given project.
- Draw use case diagrams for given modules and design user interface
- Apply code standard and Identify Software Testing Techniques.

4. Teaching and Examination Scheme

Teaching Scheme (In Hours)			Total Credits (L+T+P)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
L	**T**	**P**	**C**	**ESE**	**PA**	**ESE**	**PA**	**100**
3	2	0	5	70	30	00	00	

Legends: L-Lecture; **T** – Tutorial/Teacher Guided Theory Practice; **P** - Practical; **C** – Credit **ESE** - End Semester Examination; **PA** - Progressive Assessment.

5. COURSE DETAILS

Unit	Major Learning Outcomes	Topics and Sub-topics
Unit – I Introduction of Software Engineering	1a. Identify the need of Software Engineering 1b. Compare Programs and Products 1c. Explain evaluation of Software	1.1 The Software Engineering Discipline-Its Evolution and Impact 1.2 Software Development Projects: Programs versus Products 1.3 Emergence of Software Engineering: Early Computer Programming, High-Level Language Programming, Control Flow-Based Design, Data Structure-Oriented Design, Data Flow-Oriented Design, Object-Oriented Design
Unit – II Software life cycle model	2a. Describe the importance of the stages in the software life cycle. 2b. Select an appropriate Life Cycle Model for a Project 2c. Design object oriented software and understand software development methodologies. 2d. Differentiate various life cycle models.	2.1 Need of Life Cycle Model : Need of a Document of Life Cycle Model, Phases Entry and Exit Criteria 2.2 Types of Life Cycle Model: Classical Waterfall Model (Phases, Shortcomings, Utility), Iterative Waterfall Model (Phase Containment of Errors, Shortcomings), Prototyping Model, Spiral Model (Risk Handling, Phases, Pros and Cons, as a Meta Model) 2.3 Evolutionary Model : Life Cycle Activities 2.4 Comparison of Different Life Cycle Model or a Project
	2e. Schedule the task of given projects.	2.7Scheduling : Work Breakdown Structure, Activity Networks and Critical Path Method, Gantt Charts,

Unit	Major Learning Outcomes	Topics and Sub-topics
		PERT Charts, Project Monitoring and Control
Unit – III Designing Software Requirement Specification	3a. Gather and analyze requirements of various Projects. 3b. Prepare SRS document of various projects. 3c. Explain merits and demerits of the Formal methods	3.1Requirements Gathering and Analysis : Studying the existing documentation, Interview, Task analysis, scenario analysis, form analysis 3.2Software Requirements Specification (SRS): Characteristics of a Good SRS Document, Examples of Bad SRS Document, Important Categories of Customer Requirements, Functional Requirements (Identify, Document , Traceability, Organization of the SRS, Techniques for Representing Complex Logic) 3.3 Formal Style Specification: Formal Techniques, Model versus Property-Oriented Methods, Operational Semantics
	3d. Transform the SRS document into design document 3e. Identify characteristics of good software design. 3f. Explain various types of cohesion and coupling	3.4 Outcome of a Design Process: Classification of Design Activities & Methodologies, Analysis versus Design 3.5 Characteristics of Good Software Design 3.6Cohesion and Coupling : Coupling, Cohesion, Functional Independence, Classification of Cohesiveness & Coupling
	3h. Construct Data Flow Diagram of various applications	3.7 Overview of Structured Analysis & Structured Design Methodology 3.8 Data Flow Diagrams (DFDs) : Primitive Symbols Concepts to Construct DFD Models, Developing the DFD Model of a System, Shortcomings of the DFD Model
Unit – IV Object Modeling using UML	4a. Describe various UML notations. 4b. Draw Use case diagram & Class Diagram for given project.	4.1 UML Diagrams 4.2 Use Case Model: Representation of Use Cases , Need to Develop the Use Case Diagram, Identify the Use Cases of a System, Essential versus Real Use Case, Factoring of Commonality among Use Cases,

Unit	Major Learning Outcomes	Topics and Sub-topics
		Use Case Packaging 4.3 Class Diagrams
	4c. Design a good graphical user interface. 4d. Differentiate Graphical and Text based Interface. 4e. List various types of User Interface. 4f. List types of various widgets.	4.4 Characteristics of a Good User Interface 4.5 Graphical User Interface vs. Text based User Interface 4.6Types of User Interface: Command Language based Interface, Menu based Interface, Direct Manipulation Interface 4.7 Fundamentals of Component based GUI development: Window system, types of widgets
Unit – V Coding and Testing	5a. Discuss standards and guidelines of coding 5b. List coding techniques. 5c. List various testing techniques.	5.1 Coding : Coding standards and Guidelines 5.2 Code Review : Code Walkthrough, Code Inspection, Clean Room Testing 5.3 Software Documentation : Internal and External Documentation 5.4 Testing : Unit Testing, Black Box Testing, White Box Testing

6. SUGGESTED SPECIFICATION TABLE WITH HOURS & MARKS (THEORY)

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	**Introduction of Software Engineering**	04	02	05	00	07
II	**Software life cycle model**	12	03	14	04	21
III	**Designing Software Requirement Specification**	12	00	14	07	21
IV	**Object Modeling using UML**	10	00	07	07	14
V	**Coding and Testing**	04	03	04	00	07
	Total	**42**	**08**	**44**	**18**	**70**

Legends: R = Remembrance; U = Understanding; A = Application and above levels (Revised Bloom's taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

7. SUGGESTED LIST OF EXERCISES/PRACTICALS

S. No.	Unit No.	Practical Exercises (Outcomes' in Psychomotor Domain)	Hrs. required
1	I	Draw control flow graphs for simple programs	02
2	II	Draw classical and iterative waterfall model	02
3	II	Draw spiral model of software development	02
4	II	Draw Gantt chart for the given project	02
5	III	Develop an SRS for the Railway Reservation System	04
6	III	Develop an SRS for Online Examination System	02
7	III	Develop an SRS for Online Shopping	04
8	III	Develop SRS for Library Management System	02
9	III	Draw 2nd level DFD for Railway Reservation System	04
10	III	Draw 2nd level DFD for Online Shopping System	04
11	IV	Draw UML Diagram for Library Management System	02
12	IV	Draw UML Diagram for ATM	02
13	IV	Draw UML Diagram for Order processing Management	04
14	IV	Draw UML Diagram for Online Shopping	04
Total			**40**

8. SUGGESTED LIST OF STUDENT ACTIVITIES

Following is the list of proposed student activities like:

i. Prepare charts for various models, SDLC life cycles, UML notations etc.
ii. Prepare SRS documents based on case study.
iii. Discuss various case studies available on internet.

9. SPECIAL INSTRUCTIONAL STRATEGIES (if any)

i. Concepts will be introduced through lecture sessions.
ii. Practices through case studies of real life example.
iii. Integrate one small example to develop a VB.net based application.
iv. Seminar on various case studies

10. SUGGESTED LEARNING RESOURCES

A) List of Books

S. No.	Title of Book	Author	Publication
1.	Fundamentals of Software Engineering, Third Edition	Rajib, Mall	PHI Publication
2.	Software Engineering, Seventh Edition	Roger S. Pressman	McGraw Hill
3.	Structured System analysis and Design	Madhulika Jain	Bph Publication
4.	Object Oriented Modeling and design with UML, second edition	Michael R Blaha and James R Rambaugh	Pearson Prentice Hall

B) List of Major Equipment/ Instrument with Broad Specifications

- Computer System with latest configuration and memory

C) List of Software/Learning Websites

i. Software: E-draw, Visio, Enterprise Architect, Visual Paradigm, Using Creately's web based UML tools, draw.io etc
ii. http://forum.jntuworld.com/showthread.php?3841-SOFTWARE-ENGINEERING-(SE)-Notes-All-8-Units
iii. Ppts: www.facweb.iitkgp.ernet.in/~spp/LECT1.ppt
iv. Ppts: http://www.phindia.com/rajibmall/chapters/
v. http://msdn.microsoft.com/en-us/library/vstudio/dd409432.aspx

11. COURSE CURRICULUM DEVELOPMENT COMMITTEE

Faculty Members from Polytechnics

1. Prof. R. M. Shaikh, H.O.D Computer Department, K. D. Polytechnic, Patan
2. Prof. K. N. Raval, H.O.D Computer Department, R. C. Technical Institute, Ahmedabad
3. Prof. M. P. Mehta ,Sr. Lecturer Computer Engineering Department, K. D. Polytechnic, Patan
4. Prof. A. S. Galathiya ,Lecturer Computer Engineering Department, R. C. Technical Institute, Ahmedabad

Coordinator and Faculty Members from NITTTR Bhopal

1. Prof. (Mrs.) Susan S. Mathew
2. Dr. Joshua Earnest