PROGRAM EDUCATIONAL OBJECTIVES:

The Software Engineering program educational objectives are to produce post graduates who demonstrate:

- an ability to design, synthesize, and analyze, software systems of increasing size and complexity at various abstraction levels, from the individual component to the entire system architecture
- an ability to define, assess, and tailor software quality practices, and software processes and methodologies for appropriate application on software development projects in a variety of domain areas
- an ability to be an effective member of a multi-disciplinary software-intensive product development team with an awareness of individual professional and ethical responsibilities
- an ability to communicate, to varied stakeholder audiences, technical concepts in a complete, concise, and correct manner in a format appropriate for the audience
- an ability to critically analyze issues in industry and research trends and analyze and use various state-of-the-art practices and tools

PROGRAM OUTCOMES

Students will be able to:

a. Use of formal methods, use of finite state machines, software quality measurement and analysis, requirements gathering, software system design.
b. Identifying project constraints and solutions, problem decomposition, requirements elicitation, design trade-off analysis
c. Design to constraints and requirements; scope management for time constraints; addressing aspects such as maintainability, testability, or evolvability; software security.
d. Design and analysis of usability tests, system debugging, performance testing and analysis, project metric tracking.
e. Function effectively on teams to accomplish a common goal
f. Communicate effectively to varied stakeholder audiences, technical concepts in a complete, concise, and correct manner in a format appropriate for the audience
g. Analyze the local and global impact of computing on individuals, organizations, and society
h. Recognize the need for and possess an ability to engage in lifelong learning and continuing professional development
i. Use effective, proper and state-of-the-art software engineering tools and technologies.
j. Critically analyze existing literature in an area of specialization and develop innovative and research oriented methodologies to tackle gaps identified
### SEMESTER I

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW8101</td>
<td>Software Architecture and Design</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>CP8151</td>
<td>Advanced Data Structures and Algorithms</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>CP8152</td>
<td>Object Oriented Systems Engineering</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>CP8153</td>
<td>Open Source Systems and Networking</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>MA8154</td>
<td>Advanced Mathematics for Computing</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
</tbody>
</table>

#### PRACTICAL

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW8111</td>
<td>Professional Practice</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

**TOTAL** 15 1 6 19

### SEMESTER II

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW8201</td>
<td>Software Quality Assurance and Testing</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>SW8202</td>
<td>Integrated Software Project Management</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>CP8251</td>
<td>Virtualization Techniques</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Elective I</td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Elective II</td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

#### PRACTICAL

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW8211</td>
<td>Case Study</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>SW8212</td>
<td>Technical Seminar</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

**TOTAL** 15 0 8 19

### SEMESTER III

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP8351</td>
<td>Security Principles and Practices</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

#### PRACTICAL

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW8311</td>
<td>Project Work Phase I</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>6</td>
</tr>
</tbody>
</table>

**TOTAL** 12 0 12 18
SEMESTER IV

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW8411</td>
<td>Project Work Phase II</td>
<td>0</td>
<td>0</td>
<td>24</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>24</td>
<td>12</td>
</tr>
</tbody>
</table>

TOTAL NO OF CREDITS: 68

LIST OF ELECTIVES

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>COURSE TITLE</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW8001</td>
<td>Agile Software Engineering</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8002</td>
<td>Business Intelligence</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8003</td>
<td>Enterprise Application Integration</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8004</td>
<td>Enterprise Resource Planning Concepts</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8005</td>
<td>Formal Methods in Software Engineering</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8006</td>
<td>IT Systems Management</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8007</td>
<td>Knowledge Management</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8008</td>
<td>Managing Human Resource</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8009</td>
<td>Principles of Supply Chain Management</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8010</td>
<td>Software Agents</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8011</td>
<td>Software Engineering Process Models</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8012</td>
<td>Software Reliability and Metrics</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8013</td>
<td>Software Requirements Management</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8014</td>
<td>Software Test Automation</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8015</td>
<td>User Interface Design</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8016</td>
<td>Web Design and Management</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>IF8252</td>
<td>Cloud Computing Technologies</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>CP8074</td>
<td>Real Time Systems Design</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>SW8071</td>
<td>Software Verification and Validation</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>
OBJECTIVES:
- To understand the architecture and quality attributes of the software.
- To understand the design fundamentals and methodologies of the software.
- To learn the tools of architectural design for the current trends.

UNIT I INTRODUCTION TO ARCHITECTURE

UNIT II DESIGN FUNDAMENTALS

UNIT III DESIGN METHODOLOGIES

UNIT IV ARCHITECTURAL DESIGN

UNIT V CASE STUDIES
Tools for Architectural design – Case Studies.

TOTAL: 45 PERIODS

OUTCOMES:
Upon Completion of the course, the students will be able to
- To reconstruct the software architecture that can be used for an application of your choice.
- Able to design the software using design fundamentals and methodologies.
- To create a good software by using the styles, architectural design space.

REFERENCES:
OBJECTIVES:
- To extend the students' knowledge of algorithms and data structures, and to enhance their expertise in algorithmic analysis and algorithm design techniques.
- Expected to learn a variety of useful algorithms and techniques and extrapolate from them in order to then apply those algorithms and techniques to solve problems.

UNIT I FUNDAMENTALS

UNIT II HEAP STRUCTURES
Min/Max heaps - Dears - Leftist Heaps - Binomial Heaps - Fibonacci Heaps - Skew Heaps - Lazy-Binomial Heaps.

UNIT III SEARCH STRUCTURES

UNIT IV GEOMETRIC ALGORITHMS

UNIT V PARALLEL ALGORITHMS
Flynn's Classifications - List Ranking - Prefix computation - Array Max - Sorting on EREW PRAM - Sorting on Mesh and Butterfly - Prefix sum on Mesh and Butterfly - Sum on mesh and butterfly - Matrix Multiplication - Data Distribution on EREW, Mesh and Butterfly.

TOTAL: 45 + 30 : 75 PERIODS

OUTCOMES:
- Basic ability to analyze algorithms and to determine algorithm correctness and time efficiency class.
- Master a variety of advanced data structures and their implementations.
- Master different algorithm design techniques in computational geometry and in parallel algorithms.
- Ability to apply and implement learned algorithm design techniques and data structures to solve problems.

REFERENCES:
5. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein
OBJECTIVES:
- To understand the importance of object oriented software engineering.
- To study the various lifecycle models for developing software’s.
- To analyze and design software using tools.
- To develop efficient software, deploy and maintain after production.

UNIT I
CLASSICAL PARADIGM
System Concepts – Project Organization – Communication – Project Management

UNIT II
PROCESS MODELS

UNIT III
ANALYSIS:
Requirements Elicitation – Use Cases – Unified Modeling Language, Tools – Analysis Object Model (Domain Model) – Analysis Dynamic Models – Non-functional requirements – Analysis Patterns

UNIT IV
DESIGN

UNIT V
IMPLEMENTATION, DEPLOYMENT AND MAINTENANCE

TOTAL:45 + 30 =75 PERIODS

OUTCOMES:
- To prepare object oriented design for small/medium scale problem.
- To evaluate the appropriate life cycle model for the system under consideration.
- To apply the various tools and patterns while developing software.
- Testing the software against usability, deployment, maintenance.

REFERENCES:

CP8153
OPEN SOURCE SYSTEMS AND NETWORKING

OBJECTIVES:
- To understand the basic issues in open source kernels
- To appreciate the different aspects of processes
- To understand the role played by files and devices
- To understand the basic issues in open source networking
- To appreciate the different aspects of internetworking
UNIT I FOUNDATION

UNIT II PROCESSES

UNIT III FILES AND DEVICES
Virtual File System – I/O architecture and device drivers – block device drivers – page cache – accessing files

UNIT IV NETWORKING
Introduction, critical data structures, user space to kernel interface – System initialization: notification chains, device initialization, PCI layer, network interface cards, component initialization, device registration and initialization - Transmission and reception: interrupts and network drivers, frame reception and transmission, protocol handlers – Bridging: concepts, spanning tree protocol, implementation.

UNIT V INTERNETWORKING

TOTAL: 45 PERIODS

OUTCOMES:
Upon completion of the course, the students will be able to
• Identify the different features of open source kernels
• Install and use available open source kernel
• Modify existing open source kernels in terms of functionality or features used
• Identify different features of open source networking
• Modify and use existing open source networking modules

REFERENCES:

MA8154 ADVANCED MATHEMATICS FOR COMPUTING L T P C
3 1 0 4

OBJECTIVES:
To understand the basics of random variables and standard distributions
• To understand the arrival process and various queueing and server models
• To appreciate the use of simulation techniques
• To apply testing of hypothesis to infer outcome of experiments
• To apply mathematical linear programming techniques to solve constrained problems.
UNIT I  RANDOM VARIABLES

UNIT II  QUEUEING MODELS

UNIT III  SIMULATION
Discrete Event Simulation – Monte – Carlo Simulation – Stochastic Simulation – Applications to Queuing systems.

UNIT IV  TESTING OF HYPOTHESIS
Sampling distributions – Estimation of parameters - Statistical hypothesis – Tests based on Normal, t, Chi-square and F distributions for mean, variance and proportion.

UNIT V  LINEAR PROGRAMMING

TOTAL: 60 PERIODS

OUTCOMES:
Upon completion of the course, the student will be able to

• Identify the type of random variable and distribution for a given operational conditions/scene
• Study and Design appropriate queuing model for a given problem/system situation
• To understand and simulate appropriate application/distribution problems
• Differentiate/infer the merit of sampling tests.
• Formulate and find optimal solution in the real life optimizing/allocation/assignment problems involving conditions and resource constraints.

REFERENCES:

SW8111  PROFESSIONAL PRACTICE

THE OBJECTIVES OF PROFESSIONAL PRACTICE:
To Facilitate Research, Analysis, and Problem Solving.
To Interview people who know the context of the Problem and the Solution
To Explore various possible alternative solutions
To Estimate Risk
THE OUTCOMES OF PROFESSIONAL PRACTICE:
Formulating a Problem
Describing the Background of the Problem
Assessing the needs of the People
Framing a Policy
Predicting Business Opportunity
Understanding System Implications.

TOTAL: 30 PERIODS

SW8201 SOFTWARE QUALITY ASSURANCE AND TESTING  
L T P C  3 0 2 4

OBJECTIVES:
- To Know what is software and the usage of different types of software’s.
- To know the Quality Metrics of various Software’s.
- Plans, methods and process are executed to get a good Quality software.
- Knowing the methodologies in making Software.
- To test the product finally to check the product Quality.

UNIT I INTRODUCTION  

UNIT II TESTING METHODOLOGIES  
Basics of software testing – test generation from requirements – finite state models – combinatorial designs - test selection, minimization and prioritization for regression testing – test adequacy, assessment and enhancement

UNIT III TEST STRATEGIES  
Testing strategies – white box and black box approach – integration testing – system and acceptance testing – performance testing – regression testing - internationalization testing – ad-hoc testing – website testing – usability testing – accessibility testing Test plan – management – execution and reporting – software test automation – automated testing tools

UNIT IV SOFTWARE QUALITY  

UNIT V SQA IN PROJECT MANAGEMENT  
Project progress control – costs – quality management standards – project process standards – management and its role in SQA – SQA unit

TOTAL: 45+30 = 75 PERIODS

OUTCOMES:
- To analyze the product Quality by various testing methods.
- To use various testing methods for the appropriate applications.
- To assess Quality standards.
REFERENCES:

SW8202 INTEGRATED SOFTWARE PROJECT MANAGEMENT

OBJECTIVES:
• To understand the basic concept of project management
• To appreciate the various costing and life cycle management
• To understand the role played by risk in software project
• To appreciate the use of metrics for software project management
• To appreciate the challenges in people management

UNIT I PROJECT MANAGEMENT CONCEPTS

UNIT II COSTING & LIFECYCLE MANAGEMENT

UNIT III RISK MANAGEMENT

UNIT IV METRICS
UNIT V  PEOPLE MANAGEMENT

TOTAL: 45PERIODS

OUTCOMES:
Upon Completion of the course the students will be able to
- Identify the various elements of software management process framework
- Use available open source estimation tools for cost estimation
- Identify existing risk and perform risk assessment
- Design a software metric for software project management
- Modify the art of interviewing people for a given scenario

REFERENCES:

CP8251  VIRTUALIZATION TECHNIQUES

OBJECTIVES:
- To understand the need of virtualization
- To explore the types of virtualization
- To understand the concepts of virtualization and virtual machines
- To understand the practical virtualization solutions and enterprise solutions
- To understand the concepts of cloud computing
- To have an introduction to cloud programming giving emphasis to Hadoop MapReduce
- To understand the security issues in cloud computing

UNIT I  OVERVIEW OF VIRTUALIZATION
UNIT II  SERVER AND NETWORK VIRTUALIZATION  9+6

UNIT III  STORAGE, DESKTOP AND APPLICATION VIRTUALIZATION  9+6

UNIT IV  APPLYING VIRTUALIZATION  9+6

UNIT V  CLOUD COMPUTING  9+6

TOTAL: 45 + 30 = 75 PERIODS

OUTCOMES:
Upon completion of the course, the students will be able to
• Deploy legacy OSs on virtual machines
• Understand the intricacies of server, storage, network, desktop and application virtualizations
• Design new models for virtualization
• Design and develop cloud applications on virtual machine platforms
• Design new models for Big data processing in cloud

REFERENCES:

SW8211 CASE STUDY

The case study approach is to engage students in critical thinking for real-world situations. As Students, they turn basic knowledge into principles that can be applied across cases. By placing them in real situations, and requiring them to make decisions, students learn to connect their knowledge with analytical skills.

A Case Study includes Research, Analysis, and Problem Solving. Interviewing people who know the place or the situation is a vital step. There is no single solution. A case study includes:

- Introduction
- Background
- People
- Policy
- Business Opportunity
- System Implications

TOTAL: 30 PERIODS

SW8212 TECHNICAL SEMINAR

THE OBJECTIVES OF TECHNICAL SEMINAR ARE:
1. To elicit pro-active participation of the students through
2. To entrust assignment to present
3. To inculcate presentation and leadership skills among students
4. To involving students to learn actively
5. To offer opportunities of interaction with peer students and staff

THE OUTCOMES OF THE TECHNICAL SEMINAR ARE:
1. Good Communications Skills.
2. Knowing the Audience.
3. Choosing the Topic.
4. Setting the Goals for the Talk.
5. Talking to the Audience.
6. Knowing the Content of the Talk.
7. Preparation of the Slides.
8. Answering Questions.

TOTAL: 30 PERIODS
OBJECTIVES:
- To understand the mathematical foundations of security principles
- To appreciate the different aspects of encryption techniques
- To understand the role played by authentication in security
- To appreciate the current trends security practices

UNIT I INTRODUCTION AND MATHEMATICAL FOUNDATION 9
An illustrative communication game – safeguard versus attack – Probability and Information Theory - Algebraic foundations – Number theory.

UNIT II ENCRYPTION – SYMMETRIC TECHNIQUES 9

UNIT III ENCRYPTION – ASYMMETRIC TECHNIQUES AND DATA TECHNIQUES 9

UNIT IV AUTHENTICATION 9

UNIT V SECURITY PRACTICES 9

TOTAL: 45 PERIODS

OUTCOMES:
Upon Completion of the course, the students will be able to
- Use the mathematical foundations in security principles
- Identify the features of encryption and authentication
- Use available security practices

REFERENCES:
OBJECTIVES:
- To provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software
- A good understanding of software design and a set of software technologies and APIs
- Detailed examination and demonstration of Agile development and testing techniques
- Benefits and pitfalls of working in an Agile team
- Understand Agile development and testing

UNIT I AGILE METHODOLOGY
Theories for Agile management – agile software development – traditional model vs. agile model - classification of agile methods – agile manifesto and principles – agile project management – agile team interactions – ethics in agile teams - agility in design, testing – agile documentations – agile drivers, capabilities and values

UNIT II AGILE PROCESSES

UNIT III AGILITY AND KNOWLEDGE MANAGEMENT

UNIT IV AGILITY AND REQUIREMENTS ENGINEERING

UNIT V AGILITY AND QUALITY ASSURANCE
Agile product development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile approach to Quality Assurance - Test Driven Development – Agile approach in Global Software Development

TOTAL: 45 PERIODS

OUTCOMES:
- The importance of interacting with business stakeholders in determining the requirements for a software system.
- Iterative software development processes: how to plan them, how to execute them.
- The impact of social aspects on software development success.
- Techniques and tools for improving team collaboration and software quality.
- Software process improvement as an ongoing task for development teams.
- How agile approaches can be scaled up to the enterprise level.
REFERENCES:


SW8002 BUSINESS INTELLIGENCE

OBJECTIVES:

- To identify the enormous opportunities that currently exists in providing business intelligence services
- Gain a practical understanding of the key data mining methods of classification, prediction, data reduction and exploration
- To understand and help develop the strategies of modern enterprise decision makers
- Knowledge in many scientific and technological fields including data warehouses, data mining, content analytics, business process management, visual analytics, and requires competences in information systems, web science, decision science, software engineering, and innovation and entrepreneurship

UNIT I INTRODUCTION


UNIT II BI BIG PICTURE


UNIT III BI ARCHITECTURE


UNIT IV BI TECHNOLOGIES


UNIT V FUTURE OF BI


TOTAL: 45 PERIODS
OUTCOMES:
- Be aware of the business intelligence potential of today’s data rich environment
- Know how to decide when to use which technique
- Understand how to implement major techniques using Excel add-ins
- Gain the intellectual capital required to provide business analytics services

REFERENCES:
5. Swain Scheps, Business Intelligence For Dummies, John Wiley & Sons, 2011
8. Elizabeth Vitt, Michael Luckeivich, Stacia Misner, Business Intelligence, O'Reilly Media, Inc., 2010

SW8003 ENTERPRISE APPLICATION INTEGRATION

OBJECTIVES:
- Create a Business Data Catalog Application Definition file
- discuss the most important topics and technologies related to enterprise application integration
- To expose the students to practice implementation strategies.
- To familiarize the student with the Integration facilities for an application.
- To appreciate the current trends in Enterprise Application Integration.

UNIT I INTRODUCTION

UNIT II ENTERPRISE INTEGRATION ARCHITECTURE

UNIT III SERVICE AND INFORMATION INTEGRATION ARCHITECTURE
UNIT IV  PROCESS AND APPLICATION INTEGRATION ARCHITECTURE  9

UNIT V  CASE STUDY  9
TOTAL: 45 PERIODS

OUTCOMES:
- Learn how to use object-oriented concepts in developing web applications
- Understand the enterprise application integration
- Deploy the components of EIA with solutions
- Better collaboration needed in application integration.

REFERENCES:
2. Martin Fowler Patterns of Enterprise Application Architecture (Addison-Wesley Signature Series) 2002

SW8004 ENTERPRISE RESOURCE PLANNING CONCEPTS L T P C 3 0 0 3

OBJECTIVES:
- To enrich students with concepts and knowledge of ERP.
- To focus on illustrating procurement, production, and sales business processes using ERP software.
- To bridge the gap between the need of business process knowledge and its application to the business environment.
- To learn ERP Implementation Success & Failure for an application.

UNIT I  INTRODUCTION TO ERP  9

UNIT II  ERP IMPLEMENTATION  9

UNIT III  BUSINESS MODULES  9

UNIT IV  ERP MARKET  9
UNIT V   ERP – APPLICATIONS
Lean manufacturing and ERP - Turbo Charge the ERP System – EIA Study of ERP selection process – Big Bang ERP implementation – Impact of ERP systems on organizational effectiveness – Knowledge management for enterprise systems

OUTCOMES:
- To make the student a manager of computer service offerings across business processes of an organization.
- Understand the structure of an ERP system
- To prepare them to become knowledgeable ERP user professionals suitable to Industry and Information Technology Companies.

REFERENCES:

SW8005   FORMAL METHODS IN SOFTWARE ENGINEERING
L T P C
3 0 0 3

OBJECTIVE:
- Students will learn to translate the informal diagrammatic and textual notations into formal interpretation sufficient for particular forms of automated reasoning. To understand the formal semantics and tools

UNIT I   INTRODUCTION

UNIT II   FORMAL SPECIFICATION STYLE
Model-Oriented – Specifications – Concurrency-Based Specifications – Example Specification Languages.

UNIT III   VDM
Introduction to VDM – Basic Types – Quote Types – Compound Types – Optional Types – Functions – Operations – Additional Constructs – Modules.

UNIT IV   THE Z NOTATION
The Interchange Language – User-Defined Identifiers – Data Types – Basic Types – Compound Types – Schemas – Additional Constructs.
UNIT V  FORMAL SEMANTICS AND TOOLS

TOTAL: 45 PERIODS

OUTCOMES:
- Ability to apply knowledge of logic, formal methods and software engineering methods.
- To use VDM and Z notation in the context of software engineering
- To assess the software against semantics and tools.

REFERENCES:

SW8006 IT SYSTEMS MANAGEMENT L T P C
3 0 0 3

OBJECTIVES:
- To study the importance of change, configuration and operations in Systems Management.
- To manage the people, tools and process based on the change.
- It provides frameworks and management principles that current or aspiring managers can employ to cope with the challenges inherent in the implementation of rapidly advancing technology.
- Formulate a project proposal that includes a broad research issue anchored in the management of information systems literature.

UNIT I INTRODUCTION

UNIT II CHANGE MANAGEMENT
Change Management Processes - Identifying the need for change, Making a business case and measuring return on investment, Managing change – people, tools, processes.

UNIT III CONFIGURATION MANAGEMENT

UNIT IV OPERATIONS MANAGEMENT I

UNIT V OPERATIONS MANAGEMENT II

OUTCOMES:
- Analyze the need of system management.
- Ability to predict the need for change, and manage the same.
- To evaluate the risk ahead and plan for the business continuity.
- Write up a report and a project report of high quality that draws upon and synthesizes a number of sources and fulfills the objectives in the project proposal.
- Apply critical thinking to academic and practice sources.
REFERENCES:

SW8007 KNOWLEDGE MANAGEMENT

OBJECTIVES:
- To learn knowledge engineering basics
- To Knowing the knowledge models
- To know the techniques of knowledge management and implementation

UNIT I INTRODUCTION

UNIT II KNOWLEDGE MODELS

UNIT III TECHNIQUES OF KNOWLEDGE MANAGEMENT
Knowledge Elicitation Techniques – Modeling Communication Aspects – Knowledge Management and Organizational Learning.

UNIT IV KNOWLEDGE SYSTEM IMPLEMENTATION

UNIT V ADVANCED KNOWLEDGE MANAGEMENT

TOTAL: 45 PERIODS

OUTCOMES:
- Learning knowledge engineering basics
- Knowing the knowledge models
- knowing the techniques of knowledge management and implementation

TEXT BOOKS:
REFERENCES:
2. http://www.epistemics.co.uk

SW8008 MANAGING HUMAN RESOURCE

OBJECTIVES:
To know the importance of resources for a task.
- To compare all the resources with Human resources so the employee constraints are checked to meet the completion of the task.
- Training types are studied for employees
- To forecasting Human Resource requirement.
- To know the selection procedures.

UNIT I PERSPECTIVES IN HUMAN RESOURCE MANAGEMENT

UNIT II THE CONCEPT OF BEST FIT EMPLOYEE

UNIT III TRAINING AND EXECUTIVE DEVELOPMENT
Types of training, methods, purpose, benefits and resistance. Executive development programmes – common practices - benefits – self development – knowledge management.

UNIT IV SUSTAINING EMPLOYEE INTEREST

UNIT V PERFORMANCE EVALUATION AND CONTROL PROCESS

TOTAL: 45 PERIODS

OUTCOMES:
To apply recruitment techniques in employee selection process.
- To apply appropriate training process
- To motivate employees by sustaining employee Interests.explain the development, management and exploitation of information systems, discuss their impact on organizations and evaluate.
REFERENCES:

SW8009 PRINCIPLES OF SUPPLY CHAIN MANAGEMENT

OBJECTIVES:
To learn about the E-business environment driven by the Automation Software in quick movement of supply of products.
- To study the fundamentals of supply chain management comprised of Inventory management and warehousing etc as co parts of entire business.
- Learning the cost management for the supply of products.
- The main objectives of Supply chain management are to improve the overall organization performance and customer satisfaction by improving product or service delivery to consumer.
- The higher the supply chain profitability or surplus, the more successful is the supply chain.

UNIT I FUNDAMENTALS OF SUPPLY CHAIN MANAGEMENT
Supply chain networks, Integrated supply chain planning, Decision phases in supply chain, process view of a supply chain, supply chain flows, Overview of supply chain models and modeling systems, Supply chain planning: Strategic, operational and tactical, Understanding supply chain through process mapping and process flow chart.

UNIT II SCM STRATEGIES, PERFORMANCE
Supply chain strategies, achieving strategic fit, value chain, Supply chain drivers and obstacles, Strategic Alliances and Outsourcing, purchasing aspects of supply chain, Supply chain performance measurement: The balanced score card approach, Performance Metrics. Planning demand and supply: Demand forecasting in supply chain, Aggregate planning in supply chain, Predictable variability.

UNIT III PLANNING AND MANAGING INVENTORIES
Introduction to Supply Chain Inventory Management. Inventory theory models: Economic Order Quantity Models, Reorder Point Models and Multi echelon Inventory Systems, Relevant deterministic and stochastic inventory models and Vendor managed inventory models.

UNIT IV DISTRIBUTION MANAGEMENT
Role of transportation in a supply chain - direct shipment, warehousing, cross-docking; push vs. pull systems; transportation decisions (mode selection, fleet size), market channel structure, vehicle routing problem. Facilities decisions in a supply chain. Mathematical foundations of distribution management, Supply chain facility layout and capacity planning.
UNIT V        STRATEGIC COST MANAGEMENT IN SUPPLY CHAIN  9
The financial impacts, Volume leveraging and cross docking, global logistics and material positioning,
global supplier development, target pricing, cost management enablers, Measuring service levels in
supply chains, Customer Satisfaction/Value/Profitability/Differential Advantage.

TOTAL: 45 PERIODS

OUTCOMES:

- Learning about the e-business for supply of products managed with the appropriate
  methodologies and management techniques
- Knowing Supply Chain Management is consists of all parties (Including Manufacturer, Marketer,
  Suppliers, transporters, Warehouses, Retailers and even customers) directly or indirectly involved
  in fulfillment of a customer
- The supply chains of tomorrow must deliver varying degrees of six outcomes — the traditional
cost-related benefit plus responsiveness, security, sustainability, resilience and innovation —
depending on key customers’ needs
- Knowing automated back office software systems
- Knowing basic business process

REFERENCES:
2003.
2. Sunil Chopra and Peter Meindel. Supply Chain Management: Strategy, Planning, and Operation,

SW8010 SOFTWARE AGENTS  L T P C
3 0 0 3

OBJECTIVES:
- To overview of the agent systems and software agents.
- To understand the basic concepts of intelligent software agents.
- To enable the students to design and build a multiagent system.
- To have a basic understanding about software agent technology and to be familiar with some of
  the communicating languages, standardization and applications.
- To learn the use of software agents to represent and share information to coordinate activities of
  the agents for the purpose of group problem solving

UNIT I          AGENTS – OVERVIEW  9
Agent Definition – Agent Programming Paradigms – Agent Vs Object – Aglet – Mobile Agents – Agent
Frameworks – Agent Reasoning.

UNIT II         JAVA AGENTS  9
Distributed Computing – Aglets Programming – Jini Architecture – Actors and Agents – Typed and
proactive messages.
UNIT III MULTIAGENT SYSTEMS

UNIT IV INTELLIGENT SOFTWARE AGENTS
Interface Agents – Agent Communication Languages – Agent Knowledge Representation – Agent Adaptability – Belief Desire Intension – Mobile Agent Applications.

UNIT V AGENTS AND SECURITY

TOTAL : 45 PERIODS

OUTCOMES:
- Create / develop an agent based system for a particular task.
- Design an application that uses different security issues for intelligent agents.
- Effectively apply agent-based technologies in the development and application of distributed information systems that use software agents.

REFERENCES:

SW8011 SOFTWARE ENGINEERING PROCESS MODELS

OBJECTIVES:
- To learn the Software Engineering process models to make the project effectively.
- To learn different types of process models.
- To learn software development phases.
- To know set of methods in project reviews and inspections.
- To know work practices tools and techniques in developing software.

UNIT I PROCESS AND BASIC PROCESS MODELS

UNIT II ADVANCED PROCESS MODELS

UNIT III ADVANCED PROCESS MODELS – II
UNIT IV PROCESS IMPROVEMENT MODELS – I 8

UNIT V PROCESS IMPROVEMENT MODELS – II 8
Six Sigma – CMMI. TOTAL: 45 PERIODS

OUTCOMES:
Demonstrate the Software Process models for the Adaptive development of Software Knowing the process improvement for the product quality improvement.
- To categorizes and examines a number of methods for describing or modeling how software systems are developed.
- Knowing the software life cycle
- contemporary models of software development must account for software the interrelationships between software products and production processes, as well as for the roles played by tools, people and their workplaces

REFERENCES:

SW8012 SOFTWARE RELIABILITY AND METRICS L T P C
OBJECTIVES:
- To understand different definitions of software quality and how you might measure it
- To understand different notions of ‘defects’ and be able to classify them
- To understand the basic techniques of data collection and how to apply them
- To learn software metrics that define relevant metrics in a rigorous way.
- To gain confidence in ultra-high reliability

UNIT I INTRODUCTION TO SOFTWARE RELIABILITY 9
UNIT II SOFTWARE RELIABILITY MODELING 9
Concepts – General Model Characteristic – Historical Development of models – Model Classification scheme – Markovian models – General concepts – General Poisson Type Models – Binomial Type Models – Poisson Type models – Fault reduction factor for Poisson Type models.

UNIT III COMPARISON OF SOFTWARE RELIABILITY MODELS 9

UNIT IV FUNDAMENTALS OF MEASUREMENT 9

UNIT V PRODUCT METRICS 9

TOTAL : 45 PERIODS

OUTCOMES:
- To be able to perform some simple statistical analysis relevant to software measurement data.
- To understand from practical examples both the benefits and limitations of software metrics for quality control and assurance.

REFERENCES:

SW8013 SOFTWARE REQUIREMENTS MANAGEMENT L T P C 3 0 0 3

OBJECTIVES:
- To understand the basics of requirements engineering
- To appreciate the different techniques used for requirements elicitation
- To understand the role played by requirements analysis in requirement integration
- To appreciate the use of various methodologies for requirements development
- To appreciate the current trends in requirements prioritization and validation

UNIT I REQUIREMENTS ENGINEERING OVERVIEW 9
UNIT II REQUIREMENTS ELICITATION


UNIT III REQUIREMENTS ANALYSIS & MODELING


UNIT IV REQUIREMENTS DEVELOPMENT


UNIT V REQUIREMENTS PRIORITIZATION & VALIDATION


TOTAL : 45 PERIODS

OUTCOMES:

Upon Completion of the course the students will be able to

- Identify the different stakeholders
- Use available elicitation techniques and integrate the requirements
- Use existing documentation tools to capture the functionalities or features
- Design a prototype and evaluate it
- Implement a tracking tool that can be used for requirements tracking

REFERENCES:

OBJECTIVES:
- To understand the basics of test automation
- To appreciate the different aspects of test tool evaluation and test automation approach selection
- To understand the role played by test planning and design in test execution
- To appreciate the use of various testing tools for testing varied applications
- To understand test automation using case studies

UNIT I INTRODUCTION
9

UNIT II TEST FRAMEWORK AND AUTOMATION
9
Test Tool Evaluation and selection – organisations’ system engineering environment – tools that support the testing life cycle – test process analysis – test tool consideration Test framework – Test Library Management –selecting the test automation approach - test team management

UNIT III TEST PLANNING AND DESIGN
9
Test planning – Test program scope – Test requirements management – Test Events, Activities and Documentation – Test Environment – Evolving a Test plan
Test analysis and design – Test requirements analysis – Test program design – Test procedure design – Test development architecture – guidelines – automation infrastructure – test execution and review – test metrics

UNIT IV TESTING THE APPLICATIONS
9

UNIT V CASE STUDIES
9

OUTCOMES:
Upon Completion of the course the students will be able to
- Identify the different test tool proposal ways
- Use available testing tools to test some software applications
- Modify existing test metrics based on functionality or features used
- Design test cases and execute them
- Implement test scripts that can be used for automating test execution of an application of your choice

REFERENCES:

SW8015 USER INTERFACE DESIGN

OBJECTIVES:
- To understand how to study the tasks that the user needs to accomplish with the software system.
- To learn the constraints that affect the UI design.
- To study the importance of human-computer interaction.
- To identify the various facilities provided in WINDOWS including multimedia.

UNIT I INTRODUCTION

UNIT II HUMAN COMPUTER INTERACTION

UNIT III WINDOWS

UNIT IV MULTIMEDIA

UNIT V EVALUATION

OUTCOMES:
- To design a more user friendly software.
- To utilize the existing functionalities provided and develop a better design.
- To predict the need of the end user and design the interface accordingly.
- To evaluate the design with respect to the benchmarked standards.
REFERENCES:

SW8016 WEB DESIGN AND MANAGEMENT

OBJECTIVES:
- To know the importance of web technologies for the real world applications
- To learn the web page design with the appropriate scripting languages in producing a quality web application
- To know the testing techniques to test the product
- Will gain the skills and project-based experience needed for entry into web design and development careers.
- Will be able to use a variety of strategies and tools to create websites.
- Will develop awareness and appreciation of the many ways that people access the web, and will be able to create standards-based websites that can be accessed by the full spectrum of web access technologies.

UNIT I SITE ORGANIZATION AND NAVIGATION

UNIT II ELEMENTS OF PAGE DESIGN

UNIT III SCRIPTING LANGUAGES

UNIT IV PRE-PRODUCTION MANAGEMENT

UNIT V PRODUCTION, MAINTENANCE AND EVALUATION

TOTAL: 45 PERIODS

OUTCOMES:
- Various scripting languages are studied for the development of web applications
- Web design standards are studied
- Provides students with an opportunity for "real world" experience designing and developing websites for local community organizations.
- Emphasizes standards-based and accessible design.
- Begins by building a foundation of design theory principles, and all web design techniques are taught with these principles in mind
REFERENCES:

IF8252 CLOUD COMPUTING TECHNOLOGIES L T P C
3 0 0 3

OBJECTIVES:
- To understand the concept of cloud and utility computing.
- To understand the various issues in cloud computing.
- To familiarize themselves with the state of the art in cloud.
- To appreciate the emergence of cloud as the next generation computing paradigm.
- To be able to set up a private cloud.

UNIT I INTRODUCTION

UNIT II VIRTUALIZATION

UNIT III CLOUD INFRASTRUCTURE

UNIT IV PROGRAMMING MODEL

UNIT V SECURITY IN THE CLOUD

TOTAL: 45 PERIODS
OUTCOMES:
Upon Completion of the course, the students should be able to:
- Articulate the main concepts, key technologies, strengths and limitations of cloud computing
- Identify the architecture, infrastructure and delivery models of cloud computing
- Explain the core issues of cloud computing such as security, privacy and interoperability
- Choose the appropriate technologies, algorithms and approaches for the related issues

REFERENCES:

CP8074 REAL TIME SYSTEM DESIGN

OBJECTIVE:
- To learn real time operating system concepts and the associated issues & techniques

UNIT I REAL TIME SPECIFICATION AND DESIGN TECHNIQUES
9

UNIT II REAL TIME SPECIFICATION AND DESIGN TECHNIQUES
9

UNIT III INTERTASK COMMUNICATION AND SYNCHRONIZATION
9

UNIT IV REAL TIME DATABASES
9
Real time Databases – Basic Definition, Real time Vs General Purpose Databases, Main Memory Databases, Transaction priorities, Transaction Aborts, Concurrency control issues, Disk Scheduling.

UNIT V EVALUATION TECHNIQUES

OUTCOME:
• Understanding principles of real time systems design; be aware of architectures and behaviors of real time operating systems, database and applications.

REFERENCES:

SW8071 SOFTWARE VERIFICATION AND VALIDATION

OBJECTIVES:
• To understand the principles of verification and validation
• To appreciate the different verification and validation techniques
• To understand the various stages of testing
• To appreciate the use of tools for verification and validation
• To appreciate the benefit of using metrics for verification and validation

UNIT I INTRODUCTION
Principles of verification and validation – software architecture frameworks – model driven architecture – UML – systems modeling language – verification, validation and accreditation –

UNIT II METHODS OF SOFTWARE VERIFICATION

UNIT III TESTING

TOTAL: 45 PERIODS

OUTCOME:
• Understanding principles of real time systems design; be aware of architectures and behaviors of real time operating systems, database and applications.

REFERENCES:
UNIT IV        TOOLS FOR SOFTWARE VERIFICATION

UNIT V        ADVANCED APPROACHES
Automatic approach for verification and validation – validating UML behavioral diagrams – probabilistic model checking of activity diagrams in SysML – metrics for verification and validation

TOTAL : 45 PERIODS

OUTCOMES:
Upon Completion of the course the students will be able to
- Identify the different techniques for verification and validation
- Use available traceability analysis tools on some sample requirements
- Modify existing coverage analysers in terms of functionality or features used
- Design system test cases for application of your choice
- Use test case generators and test management tools for sample application

REFERENCES:
3. ESA Board for Software Standardisation and Control (BSSC), Guide to software verification and Validation, European Space Agency ESA PSS-05-10 Issue 1 Revision 1, March 1995