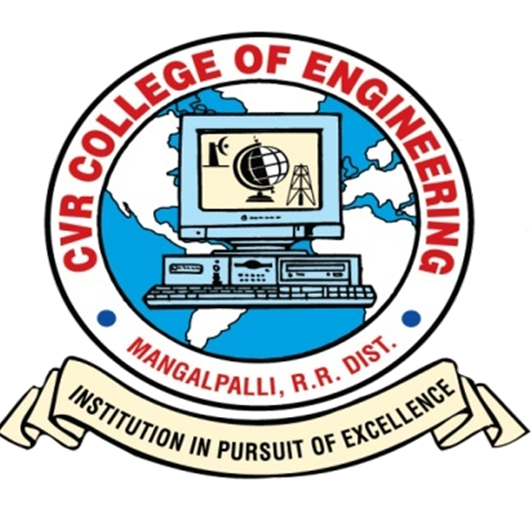
****

**CVR COLLEGE OF ENGINEERING**

**(An UGC Autonomous Institute, Accredited by NBA, NAAC with ‘A’ Grade)**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**Outcome Based Education Teaching Model Sheet**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Dept:** | Department of Computer Science and Engineering | | | |
| **Name of the Faculty & Designation** | P MADHAVI, Assistant Professor | | | |
| **Service Dept(s):** | CSE | | | |
| **CourseCode:** | 35351 | Year of Study: | 3rd Yr 2nd Sem | Section: D |
| **CourseTitle:** | Software Engineering | | | |
| **Compulsory/Elective:** | Compulsory | | | |
| **CoursePrerequisites:** | Scripting language, IDEs, Databases, Networking, SDLC | | | |
| **Prerequisite**  **Knowledge:** | Software Development Life Cycle, Product, Project metrices | | | |
| **Duration:** | One Semester | Credit Units: | 4 | |
| **Class/Laboratory**  **Schedule:** | 4 hours of lecture and 3 hours Case tools lab | | | |
| **Laboratory/Software**  **Usage:** | Case tools lab | | | |
| **CourseDescription:** | Software engineering is a discipline that allows us to apply engineering and computer science concepts in the development and maintenance of reliable, usable, and dependable software. | | | |
| **CourseObjectives:** | 1. To help the student differentiate between the programming approach and the software engineering approach and introduce the issues while building large programs.  2. To introduce basic concepts of software engineering through – project, product, process models, personal software process, team software process, umbrella activities.  3. To elaborate techniques and processes for software requirements, design methodologies, coding and testing methodologies, software metrics and quality.  4. To make the students understand how the applications of software engineering principles would improve the quality of software and decrease the cost of software development and maintenance. | | | |
| **CourseOutcomes**  **(CO):** | At the end of the course, the student should be able to  **CO1**: Understand underlying principles of software engineering, software myths and Software Process models.  **CO2**: Understand requirements engineering process and related system models.  **CO3**: Understand software design process, design quality, design models and create architectural designs, component designs and UI designs. **CO4**: Develop a strategic approach to testing and use debugging Techniques.  **CO5**: Analyze the importance of software metrics and apply risk management strategies., their technical features, and what kinds of applications they can support | | | |
| **Texts& References:**  ***(\* recommended textbook(s))*** | **TEXTBOOKS:**  Roger S. Pressman, Software Engineering, A practitioner’s Approach, 6th edition, McGraw Hill International Edition, 2005.  Ian Sommerville, Software Engineering, 10th edition, Pearson Education, 2017  **REFERENCES:**  1. Pankaj Jalote, Software Engineering A Precise Approach, 3rd Edition Wiley India, 2010.  2. Waman S Jawadekar, Software Engineering A Primer, Tata McGraw-Hill, 2010. Rajib Mall, Fundamentals of Software Engineering, 3rd Edition, PHI, 2009 | | | |
| **StudentAssessment:** | **• Assignments:10%**  **• Twomid-termexaminations:20%**  **• One finalexamination:70%** | | | |
| **CourseOutcome**  **Assessment:** | • Assignments, MidsandExternal examinations  • CourseEnd Survey | | | |

|  |  |  |
| --- | --- | --- |
|  |  Lecture |  Service learning |
| Guestlectures | Internship |
| Casestudy |  Field study |
| Pedagogical  Methods | Roleplaying  Studentpresentation | Company visits   e-learning |
|  |  Project | Independentstudy |
|  |  Simulation game | Others:Web Technologies Laboratory |
|  |  Exercisesand problems |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Percentage  Content of |  | | | | | | | | | | |
| **Math** | **Basic**  **Science** | | **Engineering**  **Science** | **Engineering Designand Synthesis** | | **Complementary**  **Studies** | | **Computer**  **Studies** | | **Total** |
| 5 | 0 | | 0 | 20 | | 25 | | 50 | | 100 |
|  | | | | | | | | | | |
| Hours per week |  | | | | | | | | | | |
| **Lecture** | | **Tutorial** | | | **Laboratory** | | **Other** | | **Total** | |
| 4 | | - | | | 3- | |  | | 7 | |
|  | | | | | | | | | | |

**CourseContent**

|  |  |  |  |
| --- | --- | --- | --- |
| **WeekNo.** | **Topics** | **Assignment test no.** | **CO No.** |
| 1 | The Nature of Software: Defining Software, Software Application Domains, Legacy Software, The Changing Nature of Software. Software Engineering: Defining the Discipline the Software Process, Software Engineering Practice, Software Development Myths. | 1 | 1 |
| 2,3 | Process Models: Prescriptive Process Models, Specialized Process Models, Component-Based Development, The Unified Process, Personal and Team Process Models, Agile Development: What Is Agility? Agility and the Cost of Change, What Is an Agile Process? Extreme Programming, Other Agile Process Models.. | 1,2 |
| 4,5 | Understanding Requirements: Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Analysis Model, Negotiating Requirements, Requirements Monitoring, Validating Requirements, Avoiding Common Mistakes. Scenario-Based Methods: Requirements Analysis, Scenario-Based Modeling, UML Models | 1,2 |
| 6,7 | That Supplement the Use Case. Class-Based Methods: Identifying Analysis Classes, Specifying Attributes, Defining Operations, Class-Responsibility-Collaborator Modeling, Associations and Dependencies, Analysis Packages. Behavioral Methods: Creating a Behavioral Model, Identifying Events with the Use Case, State Representations, Patterns for requirement modeling  Design Concepts: Design within the Context of Software Engineering, The Design Process, Design Concepts, The Design Model. Architectural Design: Software Architecture | 1,3 |
| 8,9 | Architectural Genres, Architectural Styles, Architectural Considerations, Architectural Decisions, Architectural Design, Architectural Reviews, Lessons Learned, Pattern-based Architecture Review, Architecture Conformance Checking, Agility and Architecture. Component-Level Design: Definition of Component, Designing Class-Based Components, Conducting Component-Level Design, | 2 | 3,4 |
| 10,11 | Designing Traditional Components, Component-Based Development. User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps.  Software Testing Strategies: A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Context, Validation Testing, System Testing, | 3,4 |
| 12,13 | The Art of Debugging. Testing Conventional Applications: Software Testing Fundamentals, Internal and External Views of Testing, White-Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing. Quality Concepts: Definition of Quality, Software Quality, The Software Quality Dilemma, Achieving Software Quality. | 4,5 |
| 14,15 | Product Metrics: A Framework for Product Metrics, Metrics for the Requirements Model, Metrics for the Design Model, Metrics for Source Code, Metrics for Testing, Metrics for Maintenance. Process and Project Metrics: Metrics in the Process and Project Domains, Software Measurement, Metrics for Software Quality, Integrating Metrics within the Software Process, Metrics for Small Organizations. Risk Management: Reactive versus Proactive Risk Strategies, Software Risks, Risk Identification, Risk Refinement, Risk Mitigation, Monitoring, and Management, The RMMM Plan. | 4,5 |

**Mapping Of COs to Pos:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **COs** | **POs** | | | | | | | | | | | |  | **PSOs** | | |
| **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** | **PSO3** | **PSO4** |
| **CO 1** | 2 |  |  |  |  |  |  | 1 |  |  |  | 2 |  |  |  |  |
| **CO 2** | 2 | 2 | 2 |  |  | 1 |  |  |  |  |  | 2 |  |  |  |  |
| **CO 3** |  | 2 | 2 |  | 2 |  |  |  |  |  | 2 |  |  |  |  |  |
| **CO 4** | 2 | 2 |  |  |  |  |  |  |  |  |  | 2 |  |  |  |  |
| **CO 5** |  |  |  |  |  | 2 |  | 2 | 2 |  | 1 |  |  |  | 1 | 1 |

|  |  |  |
| --- | --- | --- |
| **PO No** | **Program Outcomes (POs)** | **Contribution of COs** |
| **1** | **Engineering knowledge**: Apply the knowledgeof mathematics, science, and engineering. | 1 |
| **2** | **Problem analysis**: Identify, formulate, review and analyze complex engineering problems. | 2 |
| **3** | **Design/development of solutions**: Design solutionsforcomplexengineeringproblems. | 3 |
| **4** | **Conduct investigationsofcomplexproblems**: Useresearch-basedknowledgeandresearch methods. | 3 |
| **5** | **Modern toolusage**: Use the computer/IT tools relevant to the discipline. | 3 |
| **6** | **The engineer and society**: Understand the impact of engineering solutions to the society. | 1 |
| **7** | **Environmentandsustainability**: Understandtheimpactoftheengineeringsolutionsin societal and environmental contexts. |  |
| **8** | **Ethics**:Understand professional and ethicalresponsibility. |  |
| **9** | **Individualand team work**:Functionin a multidisciplinaryteam. |  |
| **10** | **Communication**:Communicateeffectively. |  |
| **11** | **Project management and finance**: Understand the importance ofmanagementprinciplestomanageprojects. |  |
| **12** | **Life-long learning**: Recognize theneed and have the for life-longlearning. | 2 |
| **PSO1** | **Software Development Skills**: Analyze a problem, design an algorithm, define the computing requirements and implement it through logical and programming skills. | 3 |
| **PSO2** | **Professional Skills**: Architect, evolve and integrate a working model, leading to secure software product development to meet the evolving needs of the industry and open source environments. | 2 |
| **PSO3** | **Interdisciplinary Skills**: Understand software engineering practices and hardware integration for developing solutions to real world problems over multi-disciplinary domains. | 2 |
| **PSO4** | **Industry Readiness:** Use theoretical and practical concepts of various domains to realize new ideas and innovations for pursuing research, entrepreneurship, employment and higher studies. | 3 |

**CO Calculations:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CVR College of Engineering** | | | | | |
| **(Autonomous)** | | | | | |
| **Mangapally (V), Ibrahimpatan (M), Hyderabad-501510** | | | | | |
|  | | | | | |
| III B-Tech I Sem – SE | | | | | |
| **COMPUTER SCIENCE AND ENGINEERING DEPARTMENT.** | | | | | |
| **AY:2020-21** | | **Subject: DWDM** | | | **Year/Sem: IV/I** |
|  | **Internal Examinations** | | | | **Average of Internal Examinations** |
| **CO's** | **Mid-1** | **Assignement-1** | **Mid-2** | **Assignement-2** |  |
| **CO1** | **3.00** | **3.00** |  |  | **3.00** |
| **CO2** | **3.00** | **3.00** |  |  | **3.00** |
| **CO3** | **3.00** | **3.00** | **3.00** | **3.00** | **3.00** |
| **CO4** |  |  | **3.00** | **3.00** | **3.00** |
| **CO5** |  |  | **3.00** | **3.00** | **3.00** |
| **Average of Attainemnts** | | | | | **3.00** |
| **Weightage as per curriculum** | | | | | **30%** |

**Result Analysis of Software Engineering**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Total Number of students appeared** | **S**  **(Outstanding)** | **A+**  **(Excellent)** | **A**  **(Very Good)** | **B+**  **(Good)** | **B**  **(Above Average)** | **C**  **(Average)** | **P**  **(Pass)** | **F**  **Fail** |
| 58 | 3 | 18 | 18 | 10 | 5 | 1 | 1 | 2 |

**Corrective Measures to improve Course Outcomes (COs) Attainment:**

1. Remedial coaching classes have been conducted for slow learners.
2. Regular Class tests are to be conducted to improve the performance of students in end exams.
3. More Practice programs need to be given to students in Servlets and JSPs leading to better insights into the concepts.
4. Demonstration of creative web sites creation using ICT Videos

**For Other Subjects, following measures may also be suggested:**

1. Group discussions need to be conducted during class hours to promote active learning and student involvement in teaching learning process.
2. By introducing Open book examination, students analytical skills can be improved.
3. Any other suggestion as may be deemed appropriate.