1701CP201-SOFTWARE PROCESS AND PROJECT MANAGEMENT

L T P C
3 0 0 3

OBJECTIVES:

- To understand overall SDLC and adopt suitable processes
- To elicite, analyze, prioritize, and manage both functional and quality requirements
- To estimate efforts required, plan, and track the plans
- To understand and apply configuration and quality management techniques
- To evaluate, manage, and design processes

(A mini-project can be chosen by the instructor and use it as a context for the tutorials)

UNIT I DEVELOPMENT LIFE CYCLE PROCESSES **9**

Overview of software development life cycle – introduction to processes – Personal Software Process (PSP) – Team software process (TSP) – Unified processes – agile processes – choosing the right process Tutorial: Software development using PSP

UNIT II REQUIREMENTS MANAGEMENT **9**

Functional requirements and quality attributes – elicitation techniques – Quality Attribute Workshops (QAW) – analysis, prioritization, and trade-off – Architecture Centric Development Method (ACDM) – requirements documentation and specification – change management – traceability of requirements Tutorial: Conduct QAW, elicit, analyze, prioritize, and document requirements using ACDM

UNIT III ESTIMATION, PLANNING, AND TRACKING **9**

Identifying and prioritizing risks – risk mitigation plans – estimation techniques – use case points – function points – COCOMO II – top-down estimation – bottom-up estimation – work breakdown structure – macro and micro plans – planning poker – wideband delphi – documenting the plan – tracking the plan – earned value method (EVM) Tutorial: Estimation, planning, and tracking exercises

UNIT IV CONFIGURATION AND QUALITY MANAGEMENT **9**

identifying artifacts to be configured – naming conventions and version control – configuration control – quality assurance techniques – peer reviews – Fegan inspection – unit, integration, system, and acceptance testing – test data and test cases – bug tracking – causal analysis Tutorial: version control exercises, development of test cases, causal analysis of defects

UNIT V SOFTWARE PROCESS DEFINITION AND MANAGEMENT **9**

Process elements – process architecture – relationship between elements – process modeling – process definition techniques – ETVX (entry-task-validation-exit) – process baselining – process assessment and improvement – CMMI – Six Sigma Tutorial: process measurement exercises, process definition using ETVX

TOTAL 45+15=60 PERIODS

OUTCOMES:

Upon Completion of the course, the students will be able to

CO1 Explain software development life cycle
CO2 Adopt a suitable process for software development

CO3 Elicit functional and quality requirements
CO4 Analyze, prioritize, and manage requirements
CO5 Perform trade-off among conflicting requirements

REFERENCES:

1. Pankaj Jalote, "Software Project Management in Practice", Pearson, 2002.
2. Chris F. Kemerer, "Software Project Management – Readings and Cases", McGraw Hill, 1997.
3. Watts S. Humphrey, "PSP: A self-improvement process for software engineers", Addison- Wesley, 2005.
4. Watts S. Humphrey, "Introduction to the Team Software Process", Addison-Wesley, 2000.
5. Orit Hazzan and Yael Dubinsky, "Agile software engineering", Springer, 2008.
6. James R. Persse, "Process Improvement Essentials", O'Reilly, 2006.
7. Roger S. Pressman, "Software Engineering – A Practitioner's Approach", Seventh Edition, McGraw Hill, 2010.

1702CP202-**SECURITY IN COMPUTING** **L T P C**
3 0 0 3

OBJECTIVE

To understand the basics of cryptography, learn to find the vulnerabilities in programs and to overcome them, know the different kinds of security threats in networks, databases and the different solutions available, and learn about the models and standards for security.

UNIT I ELEMENTARY CRYPTOGRAPHY **9**

Terminology and Background – Substitution Ciphers – Transpositions – Making Good Encryption Algorithms- Data Encryption Standard- AES Encryption Algorithm – Public Key Encryption – Cryptographic Hash Functions – Key Exchange – Digital Signatures – Certificates

UNIT II PROGRAM SECURITY **9**

Secure programs – Non-malicious Program Errors – Viruses – Targeted Malicious code – Controls Against Program Threat – Control of Access to General Objects – User Authentication – Good Coding Practices – Open Web Application Security Project Top 10 Flaws – Common Weakness Enumeration Top 25 Most Dangerous Software Errors

UNIT III SECURITY IN NETWORKS **9**

Threats in networks – Encryption – Virtual Private Networks – PKI – SSH – SSL – IPSec – Content Integrity – Access Controls – Wireless Security – Honeypots – Traffic Flow Security – Firewalls – Intrusion Detection Systems – Secure e-mail.

UNIT IV SECURITY IN DATABASES **9**

Security requirements of database systems – Reliability and Integrity in databases – Two Phase Update – Redundancy/Internal Consistency – Recovery – Concurrency/Consistency – Monitors – Sensitive Data – Types of disclosures – Inference.

UNIT V SECURITY MODELS AND STANDARDS **9**

Secure SDLC – Secure Application Testing – Security architecture models – Trusted Computing Base – Bell-LaPadula Confidentiality Model – Biba Integrity Model – Graham-Denning Access Control Model – Harrison-Ruzzo-Ulman Model – Secure Frameworks – COSO – CobiT – Compliances – PCI DSS – Security Standards - ISO 27000 family of standards – NIST.

TOTAL: 45 PERIODS

TEXT BOOKS:

1. Charles P. Pfleeger, Shari Lawrence Pfleeger, "Security in Computing", Fourth Edition, Pearson Education, 2007.
2. Matt Bishop, "Introduction to Computer Security", Addison-Wesley, 2004.
3. Michael Whitman, Herbert J. Mattord, "Management of Information Security", Third Edition, Course Technology, 2010.

OUTCOMES:

The students will be able to

CO1 Apply various cryptographic algorithms
CO2 Find the vulnerabilities in programs and to overcome them
CO3 Analyze different kinds of security threats

CO4 Demonstrate the requirement of databases security and solutions to attain it
CO5 Implement Models and standards of security

REFERENCES:

1. William Stallings, "Cryptography and Network Security : Principles and Practices", Fifth Edition, Prentice Hall, 2010.
2. Michael Howard, David LeBlanc, John Viega, "24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them", First Edition, Mc Graw Hill Osborne Media, 2009.
3. Matt Bishop, "Computer Security: Art and Science", First Edition, Addison-Wesley, 2002.
4. https://www.owasp.org/index.php/Top_10_2010
5. https://www.pcisecuritystandards.org/security_standards/pci_dss.shtml
6. http://cwe.mitre.org/top25/index.html

1702CP203-INTERNET OF THINGS

L T P C
3 0 2 4

OBJECTIVES :

- To understand the fundamentals of Internet of Things.
- To build a small low cost embedded system using Arduino / Raspberry Pi or equivalent boards.
- To apply the concept of Internet of Things in the real world scenario

UNIT I FUNDAMENTALS OF IOT **9**

Introduction-Characteristics-Physical design - Protocols – Logical design – Enabling technologies – IoT Levels – Domain Specific IoTs – IoT vs M2M.

UNIT II IOT DESIGN METHODOLOGY **9**

IoT systems management–IoT Design Methodology–Specifications Integration and Application Development.

UNIT III BUILDING IOT WITH RASPBERRY PI **9**

Physical device – Raspberry Pi Interfaces – Programming – APIs / Packages – Web services

UNIT IV BUILDING IOT WITH GALILEO/ARDUINO **9**

Intel Galileo Gen2 with Arduino- Interfaces - Arduino IDE – Programming - APIs and Hacks

UNIT V CASE STUDIES and ADVANCED TOPICS **9**

Various Real time applications of IoT- Connecting IoT to cloud – Cloud Storage for Iot – Data Analytics for IoT – Software & Management Tools for IoT

TOTAL: 45 PERIODS

OUTCOMES:

Upon the completion of the course the student should be able to

CO1 Design a Domain specific IoTs and analyze its characteristics
CO2 Design a portable IoT using Arduino/ equivalent boards and relevant protocols.
CO3 Develop web services to access/control IoT devices.
CO4 Deploy an IoT application and connect to the cloud.
CO5 Analyze applications of IoT in real time scenario

REFERENCES:

1. Arshdeep Bahga, Vijay Madisetti, "Internet of Things – A hands-on approach", Universities Press, 2015.

2. Manoel Carlos Ramon, "Intel® Galileo and Intel® Galileo Gen 2: API Features and Arduino Projects for Linux Programmers", Apress, 2014.

3. Marco Schwartz, "Internet of Things with the Arduino Yun", Packt Publishing, 2014.

1702CP204-**ADVANCED DATABASES** **L T P C**
3 0 0 3

OBJECTIVES:

- To learn the modeling and design of databases.
- To acquire knowledge on parallel and distributed databases and its applications.
- To study the usage and applications of Object Oriented database
- To understand the principles of intelligent databases.
- To understand the usage of advanced data models.
- To learn emerging databases such as XML, Cloud and Big Data.
- To acquire inquisitive attitude towards research topics in databases.

UNIT I PARALLEL AND DISTRIBUTED DATABASES **9**

Database System Architectures: Centralized and Client-Server Architectures – Server System Architectures – Parallel Systems- Distributed Systems – Parallel Databases: I/O Parallelism – Inter and Intra Query Parallelism – Inter and Intra operation Parallelism – Design of Parallel Systems- Distributed Database Concepts - Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control – Distributed Query Processing – Case Studies

UNIT II OBJECT AND OBJECT RELATIONAL DATABASES **9**

Concepts for Object Databases: Object Identity – Object structure – Type Constructors – Encapsulation of Operations – Methods – Persistence – Type and Class Hierarchies – Inheritance – Complex Objects – Object Database Standards, Languages and Design: ODMG Model – ODL – OQL – Object Relational and Extended – Relational Systems: Object Relational features in SQL/Oracle – Case Studies.

UNIT III INTELLIGENT DATABASES **9**

Active Databases: Syntax and Semantics (Starburst, Oracle, DB2)- Taxonomy- Applications- Design Principles for Active Rules- Temporal Databases: Overview of Temporal Databases- TSQL2- Deductive Databases: Logic of Query Languages – Datalog- Recursive Rules-Syntax and Semantics of Datalog Languages- Implementation of Rules and Recursion- Recursive Queries in SQL- Spatial Databases- Spatial Data Types- Spatial Relationships- Spatial Data Structures- Spatial Access Methods- Spatial DB Implementation.

UNIT IV ADVANCED DATA MODELS **9**

Mobile Databases: Location and Handoff Management - Effect of Mobility on Data Management - Location Dependent Data Distribution - Mobile Transaction Models -Concurrency Control - Transaction Commit Protocols- Multimedia Databases- Information Retrieval- Data Warehousing- Data Mining- Text Mining.

UNIT V EMERGING TECHNOLOGIES **9**

XML Databases: XML-Related Technologies-XML Schema- XML Query Languages- Storing XML in Databases-XML and SQL- Native XML Databases- Web Databases- Geographic Information Systems- Biological Data Management- Cloud Based Databases: Data Storage Systems on the Cloud- Cloud Storage Architectures-Cloud Data Models- Query Languages- Introduction to Big Data-Storage-Analysis.

TOTAL: 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to

CO1 Select the appropriate high performance database like parallel and distributed database
CO2 Model and represent the real world data using object oriented database
CO3 Embed the rule set in the database to implement intelligent databases
CO4 Represent the data using XML database for better interoperability
CO5 Handle Big data and store in a transparent manner in the cloud

REFERENCES:

1. R. Elmasri, S.B. Navathe, "Fundamentals of Database Systems", Fifth Edition, Pearson Education/Addison Wesley, 2007.
2. Thomas Cannolly and Carolyn Begg, "Database Systems, A Practical Approach to Design, Implementation and Management", Third Edition, Pearson Education, 2007.
3. Henry F Korth, Abraham Silberschatz, S. Sudharshan, "Database SystemConcepts", Fifth Edition, McGraw Hill, 2006.
4. C.J.Date, A.Kannan and S.Swamynathan, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2006.
5. Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems", McGraw Hill, Third Edition 2004.

1702CP205-**MACHINE LEARNING TECHNIQUES** **L T P C**
3 0 0 3

OBJECTIVES:

- To understand the machine learning theory
- To implement linear and non-linear learning models
- To implement distance-based clustering techniques
- To build tree and rule based models
- To apply reinforcement learning techniques

UNIT I FOUNDATIONS OF LEARNING **9**

Components of learning – learning models – geometric models – probabilistic models – logic models – grouping and grading – learning versus design – types of learning – supervised – unsupervised – reinforcement – theory of learning – feasibility of learning – error and noise – training versus testing – theory of generalization – generalization bound – approximationgeneralization tradeoff – bias and variance – learning curve

UNIT II LINEAR MODELS **9**

Linear classification – univariate linear regression – multivariate linear regression – regularized regression – Logistic regression – perceptrons – multilayer neural networks – learning neural networks structures – support vector machines – soft margin SVM – going beyond linearity – generalization and overfitting – regularization – validation

UNIT III DISTANCE-BASED MODELS **9**

Nearest neighbor models – K-means – clustering around medoids – silhouttes – hierarchical clustering – k-d trees – locality sensitive hashing – non-parametric regression – ensemble learning – bagging and random forests – boosting – meta learning

UNIT IV TREE AND RULE MODELS **9**

Decision trees – learning decision trees – ranking and probability estimation trees – regression trees – clustering trees – learning ordered rule lists – learning unordered rule lists – descriptive rule learning – association rule mining – first-order rule learning

UNIT V REINFORCEMENT LEARNING **9**

Passive reinforcement learning – direct utility estimation – adaptive dynamic programming – temporal-difference learning – active reinforcement learning – exploration – learning an actionutility function – Generalization in reinforcement learning – policy search – applications in game playing – applications in robot control

TOTAL : 45 PERIODS

OUTCOMES:

Upon Completion of the course, the students will be able to

CO1 Explain theory underlying machine learning
CO2 Construct algorithms to learn linear and non-linear models
CO3 Implement data clustering algorithms
CO4 Construct algorithms to learn tree and rule-based models
CO5 Apply reinforcement learning techniques

REFERENCES:

1. Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin, "Learning from Data", AMLBook Publishers, 2012.
2. P. Flach, "Machine Learning: The art and science of algorithms that make sense of data", Cambridge University Press, 2012.
3. K. P. Murphy, "Machine Learning: A probabilistic perspective", MIT Press, 2012.
4. C. M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2007.
5. D. Barber, "Bayesian Reasoning and Machine Learning", Cambridge University Press, 2012.
6. M. Mohri, A. Rostamizadeh, and A. Talwalkar, "Foundations of Machine Learning", MIT Press, 2012.
7. T. M. Mitchell, "Machine Learning", McGraw Hill, 1997.
8. S. Russel and P. Norvig, "Artificial Intelligence: A Modern Approach", Third Edition, Prentice Hall, 2009.