

B.Tech IV-I Semester (CSE)

S. No.	Course Code	Subject	L	T	P	C
1.	15A52601	Management Science	3	1	-	3
2.	15A05701	Grid & Cloud Computing	3	1	-	3
3.	15A05702	Information Security	3	1	-	3
4.	15A05703	Mobile Application Development	3	1	-	3
5.		CBCC-II	3	1	-	3
	15A05704	a. Software Architecture				
	15A05705	b. Computer Graphics				
	15A05706	c. Machine Learning				
6.		CBCC-III	3	1	-	3
	15A05707	a. Software Project Management				
	15A05708	b. Distributed Systems				
	15A05709	c. Real Time Systems				
7.	15A05710	Grid & Cloud Computing Laboratory	-	-	4	2
8.	15A05711	Mobile Application Development Laboratory	-	-	4	2
Total:			18	06	08	22

UPIQPBANK.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

15A52601 MANAGEMENT SCIENCE

Course Objectives: *The objective of the course is to equip the student the fundamental knowledge of management science and its application for effective management of human resource, materials and operation of an organization. It also aims to expose the students about the latest and contemporary developments in the field of management.*

UNIT –I: Introduction to Management: Concept-Nature and Importance of Management, Functions-Evaluation of Scientific Management, Modern management-Motivation Theories-Leadership Styles-Decision Making Process-Designing Organization Structure-Principles and Types of Organization.

UNIT- II: Operations Management: Plant location and Layout, Methods of production, Work-Study-Statistical Quality Control through Control Charts, Objectives of Inventory Management, Need for Inventory Control-EOQ&ABC Analysis(Simple Problems)**Marketing Management:** Meaning, Nature, Functions of Marketing, Marketing Mix, Channels of distribution- Advertisement and sales promotion-Marketing strategies-Product Life Cycle.

UNIT -III: Human Resource Management (HRM): Significant and Basic functions of HRM-Human Resource Planning(HRP), Job evaluation, Recruitment and Selection, Placement and Induction-Wage and Salary administration. Employee Training and development-Methods-Performance Appraisal-Employee Grievances-techniques of handling Grievances.

UNIT –IV: Strategic Management: Vision, Mission, Goals and Strategy- Corporate Planning Process-Environmental Scanning-SWOT analysis-Different Steps in Strateg Formulation, Implementation and Evaluation. **Project Management:** Network Analysis-PERT, CPM, Identifying Critical Path-Probability-Project Cost Analysis, Project Crashing (Simple Problems).

UNIT-V: Contemporary Management Practices: Basic concepts of MIS-Materials Requirement Planning(MRP),Just-In-Time(JIT)System, Total Quality Management(TQM)-Six Sigma and Capability Maturity Models(CMM) evies, Supply Chain Management, Enterprise Resource Planning(ERP),Performance Management, Business Process Outsourcing(BPO), Business Process Re-Engineering and Bench Marking, Balance Score Card.

Course Outcomes: This course enables the student to know the principles and applications of management knowledge and exposure to the latest developments in the field. This helps to take effective and efficient management decisions on physical and human resources of an organization. Beside the knowledge of Management Science facilitates for his/her personal and professional development.

TEXT BOOKS:

1. A.R Aryasri: Management Science, TMH, 2013
2. Kumar /Rao/Chalill 'Introduction to Management Science' Cengage, Delhi, 2012.

REFERENCE BOOKS:

1. A.K.Gupta "Engineering Management", S.CHAND, New Delhi, 2016.
2. Stoner, Freeman, Gilbert, Management, Pearson Education, New Delhi, 2012.
3. Kotler Philip & Keller Kevin Lane: Marketing Management, PHI, 2013.
5. Koontz & Weihrich: Essentials of Management, 6/e, TMH, 2005.
6. Kanishka Bedi, Production and Operations Management, Oxford University Press, 2004.
7. Memoria & S.V.Gauker, Personnel Management, Himalaya, 25/e, 2005
8. Parnell: Strategic Management, Biztantra, 2003.
9. L.S.Srinath: PERT/CPM, Affiliated East-West Press, 2005.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

15A05701 GRID AND CLOUD COMPUTING
Course Objectives:

The student should be made to:

- Understand how Grid computing helps in solving large scale scientific problems.
- Gain knowledge on the concept of virtualization that is fundamental to cloud computing. Learn how to program the grid and the cloud.
- Understand the security issues in the grid and the cloud environment.

Course Outcomes:

The student should be able to

- Apply the security models in the grid and the cloud environment.
- Use the grid and cloud tool kits.
- Apply the concept of virtualization.
- Apply grid computing techniques to solve large scale scientific problems

UNIT I INTRODUCTION

Evolution of Distributed computing: Scalable computing over the Internet – Technologies for network based systems – clusters of cooperative computers - Grid computing Infrastructures – cloud computing - service oriented architecture – Introduction to Grid Architecture and standards – Elements of Grid – Overview of Grid Architecture.

UNIT II GRID SERVICES

Introduction to Open Grid Services Architecture (OGSA) – Motivation – Functionality Requirements – Practical & Detailed view of OGSA/OGSI – Data intensive grid service models – OGSA services.

UNIT III VIRTUALIZATION

Cloud deployment models: public, private, hybrid, community – Categories of cloud computing: Everything as a service: Infrastructure, platform, software - Pros and Cons of cloud computing – Implementation levels of virtualization – virtualization structure – virtualization of CPU, Memory and I/O devices – virtual clusters and Resource Management – Virtualization for data center automation.

UNIT IV PROGRAMMING MODEL

Open source grid middleware packages – Globus Toolkit (GT4) Architecture , Configuration – Usage of Globus – Main components and Programming model - Introduction to Hadoop Framework - Mapreduce, Input splitting, map and reduce functions, specifying input and output parameters, configuring and running a job – Design of Hadoop file system, HDFS concepts, command line and java interface, dataflow of File read & File write.

UNIT V SECURITY

Trust models for Grid security environment – Authentication and Authorization methods – Grid security infrastructure – Cloud Infrastructure security: network, host and application level – aspects of data security, provider data and its security, Identity and access management architecture, IAM practices in the cloud, SaaS, PaaS, IaaS availability in the cloud, Key privacy issues in the cloud.

TEXT BOOK:

1. Kai Hwang, Geoffery C. Fox and Jack J. Dongarra, "Distributed and Cloud Computing: Clusters, Grids, Clouds and the Future of Internet", First Edition, Morgan Kaufman Publisher, an Imprint of Elsevier, 2012.

REFERENCES:

1. Jason Venner, "Pro Hadoop- Build Scalable, Distributed Applications in the Cloud", A Press, 2009
2. Tom White, "Hadoop The Definitive Guide", First Edition. O'Reilly, 2009.
3. Bart Jacob (Editor), "Introduction to Grid Computing", IBM Red Books, Vervante, 2005
4. Ian Foster, Carl Kesselman, "The Grid: Blueprint for a New Computing Infrastructure", 2nd Edition, Morgan Kaufmann.
5. Frederic Magoules and Jie Pan, "Introduction to Grid Computing" CRC Press, 2009.
6. Daniel Minoli, "A Networking Approach to Grid Computing", John Wiley Publication, 2005.
7. Barry Wilkinson, "Grid Computing: Techniques and Applications", Chapman and Hall, CRC, Taylor and Francis Group, 2010.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

15A05702 INFORMATION SECURITY
Course Objectives:

- Extensive, thorough and significant understanding of the concepts, issues, principles and theories of computer network security
- Identifying the suitable points for applying security features for network traffic
- Understanding the various cryptographic algorithms and implementation of the same at software level
- Understanding the various attacks, security mechanisms and services

Course Outcomes:

- Protect the network from both internal and external attacks
- Design of new security approaches
- Ability to choose the appropriate security algorithm based on the requirements.

Unit-I

Computer Security concepts, The OSI Security Architecture, Security attacks, Security services and Security mechanisms, A model for Network Security

Classical encryption techniques- symmetric cipher model, substitution ciphers, transposition ciphers, Steganography.

Modern Block Ciphers: Block ciphers principles, Data encryption standard (DES), Strength of DES, linear and differential cryptanalysis, block cipher modes of operations, AES, RC4.

Unit-II

Introduction to Number theory – Integer Arithmetic, Modular Arithmetic, Matrices, Linear Congruence, Algebraic Structures, $GF(2^n)$ Fields, Primes, Primality Testing, Factorization, Chinese remainder Theorem, Quadratic Congruence, Exponentiation and Logarithm.

Public-key cryptography - Principles of public-key cryptography, RSA Algorithm, Diffie-Hellman Key Exchange, ElGamal cryptographic system, Elliptic Curve Arithmetic, Elliptic curve cryptography

Unit-III

Cryptographic Hash functions: Applications of Cryptographic Hash functions, Requirements and security, Hash functions based on Cipher Block Chaining, Secure Hash Algorithm (SHA)

Message Authentication Codes: Message authentication Requirements, Message authentication functions, Requirements for Message authentication codes, security of MACs, HMAC, MACs based on Block Ciphers, Authenticated Encryption
Digital Signatures-RSA with SHA & DSS

Unit-IV

Key Management and distribution: Symmetric key distribution using Symmetric Encryption, Symmetric key distribution using Asymmetric, Distribution of Public keys, X.509 Certificates, Public key Infrastructure.

User Authentication: Remote user Authentication Principles, Remote user Authentication using Symmetric Encryption, Kerberos, Remote user Authentication using Asymmetric Encryption, Federated Identity Management, Electronic mail security: Pretty Good Privacy (PGP), S/MIME.

Unit-V

Security at the Transport Layer(SSL and TLS) : SSL Architecture, Four Protocols, SSL Message Formats, Transport Layer Security, HTTPS, SSH

Security at the Network layer (IPSec): Two modes, Two Security Protocols, Security Association, Security Policy, Internet Key Exchange.

System Security: Description of the system, users, Trust and Trusted Systems, Buffer Overflow and Malicious Software, Malicious Programs, worms, viruses, Intrusion Detection System(IDS), Firewalls

Text books:

1. "Cryptography and Network Security", Behrouz A. Frouzan and Debdeep Mukhopadhyay, Mc Graw Hill Education, 2nd edition, 2013.
2. "Cryptography and Network Security: Principals and Practice", William Stallings, Pearson Education , Fifth Edition, 2013.

References:

1. "Network Security and Cryptography", Bernard Menezes , Cengage Learning.
2. "Cryptography and Security", C.K. Shymala, N. Harini and Dr. T.R. Padmanabhan, Wiley-India.
3. "Applied Cryptography, Bruce Schneier, 2nd edition, John Wiley & Sons.
4. "Cryptography and Network Security", Atul Kahate, TMH.
5. 'Introduction to Cryptography", Buchmann, Springer.
6. 'Number Theory in the Spirit of Ramanujan", Bruce C.Berndt, University Press
7. "Introduction to Analytic Number Theory", Tom M.Apostol, University Press

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

15A05703 MOBILE APPLICATION DEVELOPMENT
Course Objectives:

- To understand fundamentals of android operating systems.
- Illustrate the various components, layouts and views in creating android applications
- To understand fundamentals of android programming.

Course Outcomes:

- Create data sharing with different applications and sending and intercepting SMS.
- Develop applications using services and publishing android applications.
- To demonstrate their skills of using Android software development tools

Unit 1: Introduction to Android:

The Android 4.1 jelly Bean SDK, Understanding the Android Software Stack, installing the Android SDK, Creating Android Virtual Devices, Creating the First Android Project, Using the Text view Control, Using the Android Emulator, The Android Debug Bridge(ADB), Launching Android Applications on a Handset.

Unit 2: Basic Widgets:

Understanding the Role of Android Application Components, Understanding the Utility of Android API, Overview of the Android Project Files, Understanding Activities, Role of the Android Manifest File, Creating the User Interface, Commonly Used Layouts and Controls, Event Handling, Displaying Messages Through Toast, Creating and Starting an Activity, Using the Edit Text Control, Choosing Options with Checkbox, Choosing Mutually Exclusive Items Using Radio Buttons

Unit 3: Building Blocks for Android Application Design:

Introduction to Layouts, Linear Layout, Relative Layout, Absolute Layout, Using Image View, Frame Layout, Table Layout, Grid Layout, Adapting to Screen orientation.

Utilizing Resources and Media Resources, Creating Values Resources, Using Drawable Resources, Switching States with Toggle Buttons, Creating an Images Switcher Application, Scrolling Through Scroll View, playing Audio, Playing Video, Displaying Progress with Progress Bar, Using Assets.

Unit 4: Using Selection widgets and Debugging:

Using List View, Using the Spinner control, Using the GridView Control, Creating an Image Gallery Using the ViewPager Control, Using the Debugging Tool: Dalvik Debug Monitor Service(DDMS), Debugging Application, Using the Debug Perspective.

Displaying And Fetching Information Using Dialogs and Fragments: What Are Dialogs?, Selecting the Date and Time in One Application, Fragments, Creating Fragments with java Code, Creating Special Fragments

Unit 5: Building Menus and Storing Data:

Creating Interface Menus and Action Bars, Menus and Their Types, Creating Menus Through XML, Creating Menus Through Coding, Applying a Context Menu to a List View, Using the Action Bar, Replacing a Menu with the Action Bar, Creating a Tabbed Action Bar, Creating a Drop-Down List Action Bar

Using Databases:

Using the SQLiteOpenHelper class, Accessing Databases with the ADB, Creating a Data Entry Form,

Communicating with SMS and Emails:

Understanding Broadcast Receivers, Using the Notification System, Sending SMS Messages with Java Code, Receiving SMS Messages, Sending Email, Working With Telephony Manager.

Text Books

1. Android Programming by B.M Harwani, Pearson Education, 2013.

References Text Books:

1. Android application Development for Java Programmers, James C Sheusi, Cengage Learning
2. Android In Action by w.Frank Ableson, Robi Sen, Chris King, C. Enrique Ortiz., Dreamtech.
3. Professional Android 4 applications development, Reto Meier, Wiley India, 2012.
4. Beginning Android 4 applications development, Wei- Meng Lee, Wiley India, 2013

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

15A05704
**SOFTWARE ARCHITECTURE
(CBCC-II)**
Course Objectives:

- Introduction to the fundamentals of software architecture.
- Software architecture and quality requirements of a software system
- Fundamental principles and guidelines for software architecture design, architectural styles, patterns, and frameworks.
- Methods, techniques, and tools for describing software architecture and documenting design rationale.
- Software architecture design and evaluation processes.

Course Outcomes:

- The student will be able to:
- Design and motivate software architecture for large scale software systems
- Recognize major software architectural styles, design patterns, and frameworks
- Describe a software architecture using various documentation approaches and architectural description languages
- Generate architectural alternatives for a problem and select among them
- Use well-understood paradigms for designing new systems

UNIT I: ENVISIONING ARCHITECTURE

What is software Architecture-What is Software Architecture, Other Points of View, Architectural Patterns, Reference Models, and Reference Architectures, Importance of Software Architecture, Architectural Structures and views.

ENVISIONING ARCHITECTURE:

Architecture Business Cycle- Architectures influences, Software Processes and the Architecture Business Cycle, Making of "Good" Architecture.

UNIT II: DESIGNING THE ARCHITECTURE WITH STYLES

Designing the Architecture: Architecture in the Life Cycle, Designing the Architecture, Formatting the Team Structure, Creating a Skeletal System.

Architecture Styles: Architectural Styles, Pipes and Filters, Data Abstraction and Object-Oriented Organization, Event-Based, Implicit Invocation, Layered Systems, Repositories, Interpreters.2013-2014

UNIT III: CREATING AN ARCHITECTURE-I

Creating an Architecture: Understanding Quality Attributes – Functionality and Architecture, Architecture and Quality Attributes, System Quality Attributes, Quality Attribute. Scenarios in Practice, Other System Quality Attributes, Business Qualities, Architecture Qualities.

Achieving Qualities: Introducing Tactics, Availability Tactics, Modifiability Tactics, Performance Tactics, Security Tactics, Testability Tactics, Usability Tactics.

UNIT IV: CREATING AN ARCHITECTURE-II

Documenting Software Architectures: Use of Architectural Documentation, Views, Choosing the Relevant Views, Documenting a view, Documentation across Views.

Reconstructing Software Architecture: Introduction, Information Extraction, Database Construction, View Fusion, and Reconstruction.

UNIT V: ANALYZING ARCHITECTURES

The ATAM: Participants in the ATAM, Outputs of The ATAM, Phases Of the ATAM. The CBAM: Decision-Making Context, The Basis for the CBAM, Implementing the CBAM.

The World Wide Web:A Case study in Interoperability- Relationship to the Architecture Business Cycle, Requirements and Qualities, Architecture Solution, Achieving Quality Goals.

TEXT BOOKS:

1. Software Architectures in Practice , Len Bass, Paul Clements, Rick Kazman, 2nd Edition, Pearson Publication.
2. Software Architecture , Mary Shaw and David Garlan, First Edition, PHI Publication, 1996.

REFERENCES BOOKS:

1. **Software Design: From Programming to Architecture**, Eric Braude, Wiley, 2004.
2. N. Domains of Concern in Software Architectures and Architecture Description Languages. Medvidovic and D. S. Rosenblum. USENIX.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

**15A05705 COMPUTER GRAPHICS
(CBCS-II)**
Course Objectives:

- To provide students with an understanding of the algorithms and theories that form the basis of computer graphics and modeling.
- To give students skills necessary in the production of 2D & 3D models.

Course Outcomes:

- Acquire familiarity with the relevant mathematics of computer graphics.
- Be able to design basic graphics application programs, including animation
- Be able to design applications that display graphic images to given specifications

UNIT I

Introduction, Application areas of Computer Graphics, overview of graphics systems, video-display devices, raster-scan systems, random scan systems, graphics monitors and work stations and input devices

Output primitives: Points and lines, line drawing algorithms, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms.

UNIT II

2-D Geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems.

2-D Viewing : The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland and Cyrus-beck line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm.

UNIT III

3-D Object representation: Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-spline curves, Bezier and B-spline surfaces. Basic illumination models, polygon rendering methods.

UNIT IV

3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations, 3-D viewing: Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping.

UNIT V

Visible surface detection methods: Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP-tree methods, area sub-division and octree methods
Computer animation: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications

TEXT BOOKS:

1. "Computer Graphics C version", Donald Hearn and M. Pauline Baker, Pearson education.
2. "Computer Graphics Principles & practice", second edition in C, Foley, VanDam, Feiner and Hughes, Pearson Education.

REFERENCE BOOKS:

1. "Computer Graphics Second edition", Zhiqiang Xiang, Roy Plastock, Schaum's outlines, Tata Mc Graw hill edition.
2. "Procedural elements for Computer Graphics", David F Rogers, Tata Mc Graw hill, 2nd edition.
3. "Principles of Interactive Computer Graphics", Neuman and Sproul, TMH.
4. "Principles of Computer Graphics", Shalini, Govil-Pai, Springer.
5. "Computer Graphics", Steven Harrington, TMH.
6. Computer Graphics, F.S.Hill, S.M.Kelley, PHI.
7. Computer Graphics, P.Shirley, Steve Marschner & Others, Cengage Learning.
8. Computer Graphics & Animation, M.C.Trivedi, Jaico Publishing House.
9. An Integrated Introduction to Computer Graphics and Geometric Modelling, R.Goldman, CRC Press, Taylor & Francis Group.
10. Computer Graphics, Rajesh K.Maurya, Wiley India.

B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

15A05706

MACHINE LEARNING (CBCC-II)

Course Objectives:

- To understand the basic theory underlying machine learning.
- To be able to formulate machine learning problems corresponding to different applications.
- To understand a range of machine learning algorithms along with their strengths and weaknesses.
- To be able to apply machine learning algorithms to solve problems of moderate complexity.

Course Outcomes:

- Ability to understand what is learning and why it is essential to the design of intelligent machines.
- Ability to design and implement various machine learning algorithms in a wide range of real-world applications.
- Acquire knowledge deep learning and be able to implement deep learning models for language, vision, speech, decision making, and more

Unit 1:

What is Machine Learning?, Examples of machine learning applications, supervised Learning: learning a class from examples, Vapnik- Chervonenkis dimension, probably approximately correct learning, noise, learning multiple classes, regression, model selection and generalization, dimensions of a supervised machine learning algorithm. Decision Tree Learning: Introduction, Decisions Tree representation, Appropriate problems for decision tree learning, the basic decision tree learning algorithm, Hypothesis space search in decision tree learning, Inductive bias in decision tree learning, issues in decision tree learning, Artificial Neural Networks: Introduction, Neural Network Representation – Problems – Perceptrons – Multilayer Networks and Back Propagation Algorithm, Remarks on the BACKPROPGRATION Algorithm, An illustrative Example: Face Recognition, Advanced Topics in Artificial Neural Networks.

Unit 2:

Evaluating Hypotheses: Motivation, Estimating hypothesis accuracy, basics of sampling theory, a general approach for deriving confidence intervals, differences in error of two hypothesis, comparing learning algorithms, Bayesian Learning: Introduction, Bayes Theorem, Bayes Theorem and Concept Learning, Maximum Likelihood and least squared error hypothesis, Maximum Likelihood hypothesis for predicting probabilities,

Minimum Description Length Principle, Bayes Optimal Classifier, Gibbs Algorithm, Naïve Bayes Classifier, Bayesian Belief Network, EM Algorithm

Unit 3:

Dimensionality Reduction: Introduction, Subset selection, principle component analysis, feature embedding, factor analysis, singular value decomposition and matrix factorization, multidimensional scaling, linear discriminant analysis, canonical correlation analysis, Isomap, Locally linear embedding, laplacian eigenmaps, Clustering: Introduction, Mixture densities, K- Means clustering, Expectations-Maximization algorithm, Mixture of latent variable models, supervised learning after clustering, spectral clustering, Hierarchical clustering, Choosing the number of clusters, Nonparametric Methods: Introduction, Non Parametric density estimation, generalization to multivariate data, nonparametric classification, condensed nearest neighbor, Distance based classification, outlier detection, Nonparametric regression: smoothing models, how to choose the smoothing parameter

Unit 4:

Linear Discrimination: Introduction, Generalizing the linear model, geometry of the linear discrimination, pair wise separation, parametric discrimination revisited, gradient descent, logistic discrimination, discrimination by regression, learning to rank, Multilayer Perceptrons: Introduction, the perceptron, training a perceptron, learning Boolean functions, multilayer perceptrons, MLP as a universal approximator, Back propagation algorithm, Training procedures, Tuning the network size, Bayesian view of learning, dimensionality reduction, learning time, deep learning

Unit 5:

Kernel Machines: Introduction, Optimal separating hyperplane, the non separable case: Soft Margin Hyperplane, ν -SVM, kernel Trick, Vectorial kernels, defining kernels, multiple kernel learning, multicast kernel machines, kernel machines for regression, kernel machines for ranking, one-class kernel machines, large margin nearest neighbor classifier, kernel dimensionality reduction, Graphical models: Introduction, Canonical cases for conditional independence, generative models, d separation, belief propagation, undirected Graphs: Markov Random files, Learning the structure of a graphical model, influence diagrams.

Text Books:

- 1) Machine Learning by Tom M. Mitchell, Mc Graw Hill Education, Indian Edition, 2016.
- 2) Introduction to Machine learning, Ethem Alpaydin, PHI, 3rd Edition, 2014

References Books:

- 1) Machine Learning: An Algorithmic Perspective, Stephen Marsland, Taylor & Francis, CRC Press Book

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

**15A05707 SOFTWARE PROJECT MANAGEMENT
(CBCC-III)**
Course Objectives:

The main goal of software development projects is to create a software system with a predetermined functionality and quality in a given time frame and with given costs. For achieving this goal, models are required for determining target values and for continuously controlling these values. This course focuses on principles, techniques, methods & tools for model-based management of software projects, assurance of product quality and process adherence (quality assurance), as well as experience-based creation & improvement of models (process management). The goals of the course can be characterized as follows:

- Understanding the specific roles within a software organization as related to project and process management
- Describe the principles, techniques, methods & tools for model-based management of software projects, assurance of product quality and process adherence (quality assurance), as well as experience-based creation & improvement of models (process management).
- Understanding the basic infrastructure competences (e.g., process modeling and measurement)
- Understanding the basic steps of project planning, project management, quality assurance, and process management and their relationships

Course Outcomes:

- Describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project.
- Compare and differentiate organization structures and project structures
- Implement a project to manage project schedule, expenses and resources with the application of suitable project management tools

UNIT I

Conventional Software Management: The waterfall model, conventional software Management performance. Evolution of Software Economics: Software Economics, pragmatic software cost estimation

UNIT II

Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

The old way and the new: The principles of conventional software engineering, principles of modern software management, transitioning to an iterative process

UNIT III

Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases.

Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts. Model based software architectures: A Management perspective and technical perspective.

UNIT IV

Work Flows of the process: Software process workflows, Inter Trans workflows. Checkpoints of the Process: Major Mile Stones, Minor Milestones, Periodic status assessments. Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Interaction planning process, Pragmatic planning.

Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations.

Process Automation: Automation Building Blocks, The Project Environment

UNIT V

Project Control and Process instrumentation: The server care Metrics, Management indicators, quality indicators, life cycle expectations pragmatic Software Metrics, Metrics automation. Tailoring the Process: Process discriminates, Example.

Future Software Project Management: Modern Project Profiles Next generation Software economics, modern Process transitions.

Case Study: The Command Center Processing and Display System-Replacement (CCPDS-R)

Text Books:

1. Software Project Management, Walker Royce, Pearson Education.
2. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition, Tata Mc-Graw Hill

Reference Books :

1. Applied Software Project Management, Andrew Stellman & Jennifer Greene, O'Reilly, 2006
2. Head First PMP, Jennifer Greene & Andrew Stellman, O'Reilly, 2007
3. Software Engineering Project Management, Richard H. Thayer & Edward Yourdon, second edition, Wiley India, 2004.
4. Agile Project Management, Jim Highsmith, Pearson education, 2004
5. The art of Project management, Scott Berkun, O'Reilly, 2005.
6. Software Project Management in Practice, Pankaj Jalote, Pearson Education, 2002

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

15A05708
**DISTRIBUTED SYSTEMS
(CBCC-III)**
Course Objectives:

The student should be made to:

- Understand the issues involved in studying process and resource management.
- Understand in detail the system level and support required for distributed system.
- Introduce the idea of peer to peer services and file system.
- Understand foundations of Distributed Systems.

Course Outcomes:

Student should be able to:

- Design process and resource management systems.
- Apply remote method invocation and objects.
- Apply network virtualization.
- Discuss trends in Distributed Systems.

UNIT I
INTRODUCTION

Examples of Distributed Systems – Trends in Distributed Systems – Focus on resource sharing – Challenges. Case study: World Wide Web.

UNIT II
COMMUNICATION IN DISTRIBUTED SYSTEM

System Model – Inter process Communication - the API for internet protocols – External data representation and Multicast communication. Network virtualization: Overlay networks. Case study: MPI Remote Method Invocation And Objects: Remote Invocation – Introduction - Request-reply protocols - Remote procedure call - Remote method invocation. Case study: Java RMI - Group communication - Publish-subscribe systems - Message queues - Shared memory approaches - Distributed objects - Case study: Enterprise Java Beans -from objects to components.

UNIT III**PEER TO PEER SERVICES AND FILE SYSTEM**

Peer-to-peer Systems – Introduction - Napster and its legacy - Peer-to-peer – Middleware - Routing overlays. Overlay case studies: Pastry, Tapestry- Distributed File Systems –Introduction - File service architecture – Andrew File system. File System: Features-File model -File accessing models - File sharing semantics Naming: Identifiers, Addresses, Name Resolution – Name Space Implementation – Name Caches – LDAP.

UNIT IV**SYNCHRONIZATION AND REPLICATION**

Introduction - Clocks, events and process states - Synchronizing physical clocks- Logical time and logical clocks - Global states – Coordination and Agreement – Introduction - Distributed mutual exclusion – Elections – Transactions and Concurrency Control– Transactions -Nested transactions – Locks – Optimistic concurrency control - Timestamp ordering – Atomic Commit protocols -Distributed deadlocks – Replication – Case study – Coda.

UNIT V**PROCESS & RESOURCE MANAGEMENT**

Process Management: Process Migration: Features, Mechanism - Threads: Models, Issues, Implementation. Resource Management: Introduction- Features of Scheduling Algorithms –Task Assignment Approach – Load Balancing Approach – Load Sharing Approach.

TEXT BOOK:

1. George Coulouris, Jean Dollimore and Tim Kindberg, "Distributed Systems Concepts and Design", Fifth Edition, Pearson Education, 2012.

B. Tech IV-I Sem. (CSE)

L	T	P	C
3	1	0	3

15A05709

REAL TIME SYSTEMS (CBCC-III)

Course Objectives:

- Acquire skills necessary to design and develop embedded applications by means of real-time operating systems
- Understand embedded real-time operating systems

Course Outcomes:

- Characterize real-time systems and describe their functions
- Analyze, design and implement a real-time system
- Apply formal methods to the analysis and design of real-time systems
- Apply formal methods for scheduling real-time systems
- Characterize and describe reliability and fault tolerance issues and approaches.

Unit-1

Typical Real time Applications: Digital control, High-level control, Signal processing, other Real-time Applications.

Hard versus Soft Real-Time Systems: Jobs and processors, Release time, dead lines and Timing constraints, Hard and soft timing constraints, Hard Real time systems, Soft Real-time Systems.

A Reference Model of Real Time Systems: Processors and resources, Temporal parameters of Real time workload, periodic task model, precedence constraints and data dependency, Functional parameter, Resource Parameters of Jobs and Parameters of Resources, Scheduling Hierarchy.

Commonly used Approaches to real time Scheduling: Clock-Driven Approach, Weighted Round-Robin Approach, Priority driven Approach, Dynamic vs Static Systems, Effective release time and deadlines, Optimality of the EDF and LST algorithms, Nonoptimality of the EDF and LST algorithms, Challenges in validating timing constraints in priority driven System, Off line vs On line scheduling, summary.

Unit-2

Clock-Driven Scheduling: Notations and Assumptions, static, Timer-Driven scheduler, General Structure of the Cyclic Scheduler, Improving the average response time of Aperiodic Jobs, Scheduling sporadic Jobs, Practical considerations and generalizations, Algorithm for generating Static Schedules, Pros and cons of Clock-driven scheduling, summary.

Unit-3

Priority-Driven Scheduling of periodic Tasks : Static Assumption, Fixed-priority vs Dynamic-priority Algorithms, Maximum Schedulable Utilization, Optimality of the RM and DM Algorithms, A Schedulability test for Fixed-priority tasks with Short Response time, A Schedulability test for Fixed-priority tasks with arbitrary Response time, Sufficient Schedulability conditions for the RM and DM Algorithms, summary.

Unit-4

Scheduling Aperiodic and Sporadic Jobs in Priority Driven Systems: Assumptions and approaches, Diferrable servers, Sporadic Servers, Constant utilization, total bandwidth and weighted fair –Queueing servers, Slack stealing in Dead-line Driven System, Stack stealing in Fixed-priority systems, Scheduling of sporadic jobs, Real-time performance for jobs with soft timing constraints, A two-level scheme for Integrated scheduling.

Unit-5

Resources and Resource access control: Assumptions on Resources and their usage, Effects of Resource contention and resource access control, Non Preemptive critical section, Basic Priority inheritance protocol, Basic Priority ceiling protocol, Stack –based, Priority ceiling protocol, Use of priority ceiling protocol in Dynamic priority systems, pre-emption ceiling protocol, Controlling accesses to Multiple unit Resources, Controlling concurrent accesses to data objects.

Multiprocessor Scheduling, Resource access control, and Synchronization:

Model of Multiprocessor and Distributed Systems, Task assignment, Multiprocessor Priority ceiling protocol, Elements of Scheduling Algorithms for End-to-End Periodic Tasks, Schedulability of Fixed-priority End-to-End periodic Tasks, End to End tasks in heterogeneous Systems, Predictability and validation of Dynamic Multiprocessor Systems, Summary.

Text Book:

1. "Real-Time Systems" by Jane W.S Liu, Pearson Edition, 2006.

Reference Text Book:

1. Real-Time Systems: Scheduling, Analysis, and Verification, Cheng, A. M. K.: Wiley, 2002.
2. Z.: Scheduling in Real-Time Systems, by Cottet, F., Delacroix, J., Kaiser, C., Mammeri John Wiley & Sons, 2002.
3. Real-Time Systems, C. M., Shin, K. G. McGraw-Hill, Krishna 1997.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
B. Tech IV-I Sem. (CSE)

L	T	P	C
0	0	4	2

15A05710 GRID AND CLOUD COMPUTING LABORATORY
Course Objectives:

- The student should be made to:
- Be familiar with developing web services/Applications in grid framework.
- Be exposed to tool kits for grid and cloud environment.
- Learn to use Hadoop
- Learn to run virtual machines of different configuration.

Course Outcomes:

The student should be able to
 Design and Implement applications on the Cloud.
 Design and implement applications on the Grid.
 Use the grid and cloud tool kits.

GRID COMPUTING PROGRAMS USING GRIDSIM

- 1 Program to creates one Grid resource with three machines
 - 2 Program to create one or more Grid users. A Grid user contains one or more Gridlets
 - 3 Program to shows how two GridSim entities interact with each other ; main(ie example3) class creates Gridlets and sends them to the other GridSim entities, i.e. Test class
 - 4 Program shows how a grid user submits its Gridlets or tasks to one grid resource entity
 - 5 Program to show how a grid user submits its Gridlets or task to many grid resource entities
 - 6 Program to show how to create one or more grid users and submits its Gridlets or task to many grid resource entities
 - 7 Program to creates one Grid resource with three machines
- Grid computing programs using Use Globus Toolkit or equivalent:**

- 1 Develop a new Web Service for Calculator.
- 2 Develop new OGSA-compliant Web Service.
- 3 Using Apache Axis develop a Grid Service.
- 4 Develop applications using Java or C/C++ Grid APIs
- 5 Develop secured applications using basic security mechanisms available in Globus Toolkit.
- 6 Develop a Grid portal, where user can submit a job and get the result. Implement it with and without GRAM concept.

CLOUD COMPUTING

Programs on SaaS

- 1 Create an word document of your class time table and store locally and on the cloud with doc,and pdf format . (use www.zoho.com and docs.google.com)
- 2 Create a spread sheet which contains employee salary information and calculate gross and total sal using the formula
 $DA=10\% \text{ OF BASIC}$
 $HRA=30\% \text{ OF BASIC}$
 $PF=10\% \text{ OF BASIC IF BASIC} \leq 3000$
 $12\% \text{ OF BASIC IF BASIC} > 3000$
 $TAX=10\% \text{ OF BASIC IF BASIC} \leq 1500$
 $=11\% \text{ OF BASIC IF BASIC} > 1500 \text{ AND BASIC} \leq 2500$
 $=12\% \text{ OF BASIC IF BASIC} > 2500$
 (use www.zoho.com and docs.google.com)
 $NET_SALARY=BASIC_SALARY+DA+HRA-PF-TAX$
- 3 Prepare a ppt on cloud computing –introduction , models, services ,and architecture
Ppt should contain explanations, images and at least 20 pages
 (use www.zoho.com and docs.google.com)
- 4 Create your resume in a neat format using google and zoho cloud

Programs on PaaS

- 1 Write a Google app engine program to generate n even numbers and deploy it to google cloud
- 2 Google app engine program multiply two matrices
- 3 Google app engine program to validate user ; create a database login(username, password) in mysql and deploy to cloud
- 4 Write a Google app engine program to display nth largest no from the given list of numbers and deploy it into google cloud
- 5 Google app engine program to validate the user Use mysql to store user info and deploy on to the cloud
- 6 Implement Prog 1-5 using Microsoft Azure

CASE STUDY- cloud computing

Sr. No.	Title of Experiment	Aim of the Experiment	Demonstration Equipments/ Components to be required	Type of Experiment/ Demonstration (Lab/Classroom)
1	Case Study of Amazon	To understand the services of Amazon elastic cloud.	Computers with Internet Connection	Experiment: Student perform practical under supervision of faculty and Lab technician.
2	Case Study of Azure	To understand the services of Microsoft azure.	Computers with Internet Connection	Experiment: Student perform practical under supervision of faculty and Lab technician.
3	Case Study of Hadoop	To understand the services of hadoop.	Computers with Internet Connection	Experiment: Student perform practical under supervision of faculty and Lab technician.
4	Case Study of Aneka	To understand the services of aneka elastic cloud.	Computers with Internet Connection	Experiment: Student perform practical under supervision of faculty and Lab technician.
5	Case Study of Google Apps	To understand the services of google apps engine.	Computers with Internet Connection	Experiment: Student perform practical under supervision of faculty and Lab technician.
6	Google apps business solution for data access and data upload	To understand the business solution application of Google apps.	Computers with Internet Connection	Experiment: Student perform practical under supervision of faculty and Lab technician.

7

Control panel software manager
Application of hypervisors

To understand the application of hypervisors.

Computers with Internet Connection

Experiment: Student perform practical under supervision of faculty and Lab technician.



UPIQPBANK.

B. Tech IV-I Sem. (CSE)

L	T	P	C
0	0	4	2

15A05711 MOBILE APPLICATION DEVELOPMENT LABORATORY

Course Objectives:

- To understand fundamentals of android operating systems.
- Illustrate the various components, layouts and views in creating android applications
- To understand fundamentals of android programming.

Course Outcomes:

- Create data sharing with different applications and sending and intercepting SMS.
- Develop applications using services and publishing android applications.
- To demonstrate their skills of using Android software development tools

1. Setting Up the Development Environment

1.1 Download/Install the SDK

For in-depth instructions, visit [Android Installation Documentation](#). Otherwise perform the following steps.

- Go to <http://developer.android.com/sdk/index.html>.
- Unpack to a convenient location - Remember the full path to this location, we will refer to it as <android_sdk_dir> for the rest of the lab.
 - <android_sdk_dir> would then be /home/<username>/android_dir.
- Add the path to the <android_sdk_dir>/tools directory to your system PATH
 - Windows:
 1. Right-click My Computer.
 2. Click Properties.
 3. Click Advanced tab.
 4. Click Environment Variables button.
 5. Double Click Path under System Variables.
 6. Add ; <android_sdk_dir>/tools;<android_sdk_dir>/platform-tools to the end of the Variable Values text field.
- Navigate to your <android_sdk_dir>/tools directory and type android. Add the appropriate components. See step 4 in <http://developer.android.com/sdk/installing.html>.
- Test your installation by running adb from the command line. If you did everything right, you should get a long list of help instructions.

1.2 Download/Install the Eclipse Plugin

- It is recommended that you use Eclipse 3.4 or later
 - Lab Machines - Fedora Eclipse based on 3.4.2
The version of Eclipse used by the lab machines is missing a vital component and requires adding an additional Eclipse plugin in order to use the Android plugin:
 1. Click the menu Help -> Software Updates.
 2. Click the tab Available Software -> Add Site button.
 3. Enter <http://download.eclipse.org/releases/ganymede> into the Location field.
 4. Click OK button.
 5. Enter WST Common UI into the search/text box at the top of the window (give it a second, it tries to search as you type and its kind of slow).
 6. Click the checkbox next to WST Common UI.
 7. Click the Install button.
 8. Click the Next button.
 9. Accept the terms, click Finish.
 10. Restart Eclipse.
 11. Follow the steps in the next bullet 3.4 Ganymede.
 - Eclipse 3.4 Ganymede:
 1. Click the menu Help -> Software Updates.
 2. Click Available Software tab -> Add Site button.
 3. Enter <https://dl-ssl.google.com/android/eclipse> into the "Location" field.
 4. Click OK button.
 5. Click the checkbox next to Developer Tools.
 6. Click the Install button.
 7. Click the Next button.
 8. Accept the terms, click Finish.
 9. Restart Eclipse.
 - Eclipse 3.5 Galileo:
 1. Click Help -> Install New Software .
 2. Click Add... button.
 3. Enter a name for the site into the Name field.
 4. Enter <https://dl-ssl/google.com/android/eclipse/> into the Location field.
 5. Click OK button.
 6. Click the checkbox next to Developer Tools.
 7. Click the Next button.
 8. Accept the terms, click Finish.
 9. Restart Eclipse.

- Point Eclipse to <android_sdk_dir>:
 1. Click the menu Window -> Preferences.
 2. Click Android from the Hierarchy view on the left hand side.
 3. Enter <android_sdk_dir> into the SDK Location field.
 4. Click the Apply button.
 5. Click the OK button.

1.3 Download/Install the SDK Platform Components

At the time of writing this lab there are eight different versions of the Android Platform available, ranging from 1.1 to 2.2. It is best practice to develop for the oldest platform available that still provides the functionality you need. This way you can be assured that your application will be supported by as many devices as possible. However, you will still want to download newer versions of the platforms so that you can test your applications against these as well. Due to the size of each platform component you will only be required to download and develop on one platform for the whole class. We will target the highest platform that the G1 phones support, Android 1.6 (API 4). Before we can begin developing we must download and install this platform:

- Select the menu Window -> "Android SDK and AVD Manager", or click on the black phone shaped icon in the toolbar.
- Select Available Packages on the left hand side.
- Expand the Google Android site in the "Site, Packages, and Archives" Tree.
- Check the following items:
 - SDK Platform Android 1.6, API 4 Revision 3
 - Google APIs by Google Inc., Android API 4, Revision 2
 - *NOTE: Those of you developing on Lab Machines should follow these instructions: <http://sites.google.com/site/androidhowto/how-to-1/set-up-the-sdk-on-lab-machines-linux>.*
- Click Install Selected.
- Accept the Terms for all packages and click Install Accepted.

We're now ready to develop our application.

2. Create "Hello World" Application

- 2.1 Create a new Android Project
- 2.2 Run "Hello World" on the Emulator
- 2.3 On a Physical Device
- 2.4 Greeting the User

3. Create Application by Using Widgets

- 3.1 Creating the Application by using the Activity class
 - (i) onCreate()
 - (ii) onStart()
 - (iii) onResume()

- (iv) onPause()
 - (v) onStop()
 - (vi) onDestroy()
 - (vii) onRestart()
- 3.2 Creating the Application by using Text Edit control.
 - 3.3 Creating the Application Choosing Options
 - (i) CheckBox
 - (ii) RadioButton
 - (iii) RadioGroup
 - (iv) Spinner
4. Create Application by Using Building Blocks for Android Application Design
- 4.1 Design the Application by using
 - (i) Linear Layout
 - (ii) Relative Layout
 - (iii) Absolute Layout
 - 4.2 Create the Application to play the Audio and Video clips.
5. Create Application by Using Building Menus and Storing Data
- 5.1 Design the Application for Menus and Action Bar
 - 5.2 Design the application to display the Drop-Down List Action Bar