RAMAIAH
Institute of Technology

CURRICULUM

for the Academic year 2019 – 2020

INFORMATION SCIENCE AND ENGINEERING

I & II Semester M. Tech (Software Engineering)

RAMAIAH INSTITUTE OF TECHNOLOGY
(Autonomous Institute, Affiliated to VTU)
Bangalore – 560054.

About the Institute

Ramaiah Institute of Technology (RIT)(formerly known as M.S. Ramaiah Institute of Technology) is a self-financing institution established in Bangalore in the year 1962 by the industrialist and philanthropist, Late Dr. M S Ramaiah. The institute is accredited with "A" grade by NAAC in 2014 and all engineering departments offering bachelor degree programs have been accredited by NBA. RIT is one of the few institutes with prescribed faculty student ratio and achieves excellent academic results. The institute was a participant of the Technical Education Quality Improvement Program (TEQIP), an initiative of the Government of India. All the departments have competent faculty, with 100% of them being postgraduates or doctorates. Some of the distinguished features of RIT are: State of the art laboratories, individual computing facility to all faculty members. All research departments are active with sponsored projects and more than 304 scholars are pursuing PhD. The Centre for Advanced Training and Continuing Education (CATCE), and Entrepreneurship Development Cell (EDC) have been set up on campus. RIT has a strong Placement and Training department with a committed team, a good Mentoring/Proctorial system, a fully equipped Sports department, large air-conditioned library with over 1,35,427 books with subscription to more than 300 International and National Journals. The Digital Library subscribes to several online e-journals like IEEE, JET etc. RIT is a member of DELNET, and AICTE INDEST Consortium. RIT has a modern auditorium, several hi-tech conference halls and all are air-conditioned with video conferencing facilities. It has excellent hostel facilities for boys and girls. RIT Alumni have distinguished themselves by occupying high positions in India and abroad and are in touch with the institute through an active Alumni Association. RIT obtained Academic Autonomy for all its UG and PG programs in the year 2007. As per the National Institutional Ranking Framework, MHRD, Government of India, Ramaiah Institute of Technology has achieved 64th rank in 2019 among the top 100 engineering colleges across India.

About the Department

Information Science and Engineering department is established in the year 1992 with an objective of producing high-quality professionals to meet the demands of the emerging field of Information Science and Engineering. Department also started M.Tech program in Software Engineering in the year 2004 and has been recognized as R&D center by VTU in 2012. The department is accredited by the NBA in 2001, 2004, 2010 and reaccredited in 2015 under Tier-1. Department has highly qualified and motivated faculty members, and well equipped state of the art laboratories. All faculty members are involved in research and technical papers publications in reputed journals, conferences across the world. Strong collaboration with industries and high profile institutions is in place for curriculum updates, more hands on training, practical's, project based learning, expert lectures, partial course deliveries by industry experts and student interns to keep an healthy academic atmosphere. Department is successfully conducting seminars, conferences and workshops for students and academicians in the emerging areas of Information Technology. Some of the laboratories have also been set up in collaboration with industries such as Intel, Microsoft, Apple, SECO, Honeywell, EMC^2, NVIDIA and IBM. Also, an echo system is built to initiate start-ups at the department level along with the mentorship. All the above potential activities have lead to high profile placements, motivation to become an entrepreneur, and encouragement for higher learning.

VISION OF THE INSTITUTE

To be an Institution of International Eminence, renowned for imparting quality technical education, cutting edge research and innovation to meet global socio-economic needs

MISSION OF THE INSTITUTE

MSRIT shall meet the global socio-economic needs through

- Imparting quality technical education by nurturing a conducive learning environment through continuous improvement and customization
- Establishing research clusters in emerging areas in collaboration with globally reputed organizations
- Establishing innovative skills development, techno-entrepreneurial activities and consultancy for socio-economic needs

QUALITY POLICY

We at M. S. Ramaiah Institute of Technology strive to deliver comprehensive, continually enhanced, global quality technical and management education through an established Quality Management System complemented by the synergistic interaction of the stake holders concerned

VISION OF THE DEPARTMENT

To evolve as an outstanding education and research center of Information Technology to create high quality Engineering Professionals for the betterment of Society.

MISSION OF THE DEPARTMENT

- To provide a conducive environment that offers well balanced Information Technology education and research.
- To provide training and practical experience in fundamentals and emerging technologies.
- To nurture creativity for overall personality development.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: Contribute in the area of Software Engineering development, maintenance and research in social-technical system

PEO2: Exhibit the Software Engineering skills for analysis, design and testing using modern tools and technologies within or outside discipline.

PEO3: Act according to professional ethics and communicate effectively with various stakeholders by demonstrating leadership qualities.

PROGRAM OUTCOMES (POs)

PO1: An ability to independently carry out research/investigation and development work to solve practical problems.

PO2: An ability to write and present a substantial technical report/document.

PO3: Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program.

PO4: An ability to analyze, design, verify, validate, implement, apply and maintain software systems

PO5: A recognition of the need for, and an ability to apply, professional and ethical responsibilities

Curriculum Course Credits Distribution

Semester	Humanities & Social Sciences (HSS)	Basic Sciences / Lab (BS)	Engineering Sciences/ Lab (ES)	Professional Courses - Core (Hard core, soft core, Lab) (PC-C)	Professional Courses- Electives (PC-E)	Other Electives (OE)	Project Work/Internship (PW /IN)	Extra & Co-curricular activities (EAC)	Total credits in a semester
First				14	8				22
Second				14	8				22
Third				1	4		17		22
Fourth				2			20		22
Total									**88**

SCHEME OF TEACHING

I SEMESTER

Sl. No	Subject Code	Subject	Credits				Contact Hours
			L	T	P	Total	
1	MSWE11	Advanced Software Engineering	4	0	0	04	04
2	MSWE12	Software Architecture and Design Patterns	4	0	0	04	04
3	MSWE13	Machine Learning	4	0	0	04	04
4	MSWEL1	Enterprise Application Development – I lab	0	1	1	02	04
6	MSWEAX	Elective – A	4	0	0	04	04
7	MSWEBX	Elective – B	4	0	0	04	04
		Total	**20**	**1**	**1**	**22**	**24**

Elective - A

MSWEA1	Cryptography and Information Security
MSWEA2	Advances in Databases
MSWEA3	Mobile Computing

Elective - B

MSWEB1	Probability, Statistics and Queuing Theory
MSWEB2	Advanced Algorithms
MSWEB3	Web Services

SCHEME OF TEACHING

II SEMESTER

Sl. No	Subject Code	Subject	Credits				Contact Hours
			L	T	P	Total	
1	MSWE21	Software Metrics and Quality Engineering	4	0	0	04	04
2	MSWE22	Software Project Management	4	0	0	04	04
3	MSWE23	Cloud Computing	4	0	0	04	04
4	MSWEL2	Enterprise Application Development – II Lab	0	1	1	02	04
6	MSWECX	Elective – C	4	0	0	04	04
7	MSWEDX	Elective – D	4	0	0	04	04
Total			**20**	**1**	**1**	**22**	**24**

Elective - C

MSWEC1	Internet of Things
MSWEC2	Deep Learning
MSWEC3	Advances in Operating Systems

Elective - D

MSWED1	Cognitive Computing
MSWED2	System Performance and Analysis
MSWED3	Blockchain and Cyber Security

I Semester

ADVANCED SOFTWARE ENGINEERING

Course Code: MSWE11 **Credit: 4:0:0**
Prerequisite: Software Engineering **Contact Hours: 56L**
Course Coordinator: Pushpalatha M N

Course Content:

Unit I

Specialized Process Models - Component-Based Development , The Formal Methods Model , Aspect-Oriented Software Development, Process and Team Models, Agile process, extreme programming, **SCRUM -** What is SCRUM, Overview of SCRUM, Key Aspects of SCRUM – Product Owner, Planning Poker, the Team, The Sprint, Scrum Master, Manager in Scrum and Product Backlog, Scrum Pre-Planning meeting.

Unit II

DevOps – Introduction, DevOps life cycle, Principles, benefits, Roles, Responsibilities and skills for a DevOps engineer, DevOps versus Agile, Continuous Integration and Deployment, Tools at various stages of DevOps. Research Issues in Software Engineering.
Formal modeling and verification - Formal methods concepts, Applying mathematical notation for formal specification, Formal Specification languages, The Cleanroom strategy, Functional specification, Cleanroom design, Cleanroom Testing.

Unit III

Design Process, Design concepts, Design Model, Architectural styles, architectural designs, architectural mapping using data flow, conducting component-level design, component-level design for WebApps, designing traditional components, **Webapp Design-** WebApp Design Quality, Design Goals , A Design Pyramid for WebApps , WebApp Interface Design , Aesthetic Design, Content Design ,Architecture Design, Navigation Design.

Unit IV

Software Quality and Testing: What Is Quality?, Software Quality , The Software Quality Dilemma, Achieving Software Quality , Cost Impact of Software Defects, Defect Amplification and Removal , Review Metrics and Their Use, Reviews: A Formality Spectrum , Informal Reviews , Formal Technical Reviews,

Testing object oriented applications- Broadening the view of testing, Testing OOA and OOD models, object oriented testing strategies, Object-oriented testing methods, Testing methods applicable at the class level, Interclass test case design, Testing for webapps- Testing concepts for webapps,The testing process-An overview, Content testing, User Interface Testing, Component-level testing, Navigation Testing, Configuration Testing, Security Testing and Performance testing.

Unit V

Project Scheduling, Basic Concepts, Project Scheduling, Defining a Task Set for the Software Project ,Scheduling , Time-Line Charts , Earned Value Analysis.
Risk Management : Reactive versus Proactive Risk Strategies, Software Risks, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring, and Management, Free and Open Source Software Engineering concepts and research issues.

References:

1. Roger S Pressman, Software Engineering, 7th edition, TMH publication.
2. Ian Sommerville, Software Engineering, 9th edition, Pearson Education.
3. Rumbaugh, Object –Oriented Modeling and Design, Pearson Education.
4. Fabio et al, Free and open source software development and research Opportunities for software engineering, 15th Brazilian Symposium on Software Engineering, 2011.

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Identify various software process models. (PO-1, 3, 4)
2. Describe DevOps and explain Formal modeling and verification methods. (PO-3, 4)
3. Design a high quality system or product using appropriate techniques (PO-3, 4)
4. Verify the quality of software and test the software using appropriate methods (PO-3, 4)
5. Manage software projects by proper scheduling and managing risks and Identify Research issues in free and open source software engineering. (PO-1, 3, 4)

SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

Course Code: MSWE12 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator: Naresh E

Course Content:

Unit I

Introduction to software architecture, Common architectural styles including Pipes and Filters, OO, Event based invocations, Layered systems, Repositories, Table driven interpreters and heterogeneous architectures. Some case studies in software architecture.

Unit II

Software design patterns, OO design principles, Creational patterns: Abstract Factory, Builder, Factory Method, Prototype, Singleton.

Unit III

Concepts and Applications of Structural Patterns: Adapter, Bridge, Composite, Decorator, Facade, Flyweight, Proxy, Case studies.

Unit IV

Chain of responsibility, Command, Interpreter , Iterator, Mediator. Overview, Applications, case studies are dealt with respect to specified design patterns.

Unit V

Behavioural Patterns: Memento, Observer, State, Strategy, Template Method, Visitor. UML modelling for different problem scenarios are illustrated.

References:

1. Len Bass, Paul Clement, Rick Kazman, “Software Architectures in Practice”, 3rd Edition, Pearson, 2013.
2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, “Pattern Oriented Software Architecture: A System of Patterns”, John Wiley and Sons, Volume 1, Reprinted February 2001.
3. Alan Shalloway, James R Trott, Design Patterns Explained, A New Perspective on Object Oriented Design, 2nd Edition, Addison Wesley 14.

4. Mary Shaw and David Garlan: Software Architecture-Perspectives on an Emerging Discipline, PHI Learning, 2007.
5. James W Cooper, Java Design Patterns, A Tutorial, Addison Wesley.
6. Eric Freeman, Elisabeth Freeman, Head First Design Patterns, O'reilly Publications.

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Classify some of the challenging design issues that software engineers face and the trade-offs associated with the solutions to these. (PO-1, 3, 4).
2. Describe the principles behind software patterns and be able to apply a number of the fundamental patterns (PO-1, 3, 4).
3. Summarize the need for software architecture and the principles of the classic architectural styles (PO-1, 3, 4).
4. Outline the major approaches to integrate patterns into software design (PO-1, 3, 4).
5. Demonstrate practical competence in the application and construction of software by applying appropriate architecture and patterns (PO-1, 3, 4).

MACHINE LEARNING

Course Code: MSWE13 **Credit: 4:0:0**
Prerequisite: Linear Algebra **Contact Hours: 56L**
Course Coordinator: Dr. Vijaya Kumar B P

Course Contents:

Unit I

Introduction: Introduction to Machine Learning, Why Machine Learning, Introduction to Statistics and Probability, Introduction to Julia and Python, **Learning Theory:** Regularization Bias-Variance Decomposition and Tradeoff, Concentration Inequalities Generalization and Uniform Convergence VC-dimension

Unit II

Learning Evaluation Metrics, Dimensionality Reduction Algorithms, Model selection and curse of dimensionality, Decision Tree, Random Forest, PCA, Ensemble Learning, Case studies.

Unit III

Supervised Learning: Regression, Decision Tree, Random Forest, KNN, Logistic Regression etc. Bayesian Learning methods, Support Vector Machine - Kernel Method, Gaussian Processes, Case studies.

Unit IV

ANN, Multilayer Perceptrons, Introduction to Deep Learning- CNN; Image, Text classification, **Unsupervised Learning** methods, K-means, KSOM-NN, Case studies for above topics.

Unit V

Reinforcement learning - Hidden Markov Models, Time series; Autoregressive models, Case studies Current developments, issues and future directions in Machine Learning.

Text Book:

1. Tom M Mitchell, “Machine Learning”, McGraw-Hill, Inc. New York.

References:

1. Christopher Bishop "Pattern Recognition and Machine Learning", CBS Publishers & Distributors-New Delhi
2. Ethem Alpaydin "Introduction To Machine Learning" 3rd Edition, PHI Pvt. Ltd, 2010
3. Simon Haykin, "Nueral networks and Learning Machines", 3rd Edition, PHI
4. James A. Anderson, "An introduction to Neural Networks", PHI

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Identify and interpret the concepts and issues of various learning models. (PO-1, 3, 4).
2. Evaluate the learning models generated from data, and interpret the dimensionality issues and their modeling. (PO-1, 3, 4).
3. Illustrate the supervised learning methods and compare different models for classification and regression with expected accuracy (PO-1, 3, 4).
4. Design and develop ANN perceptron, nonparametric approaches for solving learning problems and Unsupervised Learning methods. (PO-1, 3, 4).
5. Describe the Reinforcement learning technique, and apply the algorithms to a real-world problem for optimization (PO-1, 2, 3, 4).

ENTERPRISE APPLICATION DEVELOPMENT-I LAB

Course Code: MSWEL1 **Credit: 0:1:1**
Prerequisite: NIL **Contact Hours: 28T+28P**
Course Coordinator: Dr. Krishnaraj P M

Lab Exercises:

Development of Enterprise Application using J2EE (Java 2 Enterprise Edition)/ .Net / Open Source Based Framework which includes the following activities.

1. Application of Architectural Frameworks & Design Patterns.
2. Client Tier Development.
3. Web Tier Development.
4. Business Tier Development.
5. System Integration with Enterprise Information System Tier.
6. Packaging and Deployment.

The students have to work in groups of three and develop any enterprise applications like the ones listed below:

1. HR Management
   a. Recruitment System.
   b. Staff Appraisal System.
   c. Grievance Redressal System.
2. Finance Management
   a. Billing.
   b. Payroll Processing.
3. Stock Management
   a. Sales and Purchase.
   b. Marketing of Products.
4. Production Management (Increase the QTY and QLTY)
   a. Cost Management.
   b. Quality Assurance and Delivery of Finished Products.
5. Any other problem with prior approval of the faculty.

CIE Evaluation:

Regular lab sessions based on adherence to project plan - 20 Marks
Final Demonstration - 15 Marks
Project Report - 5 Marks
Implementation of changes suggested during examination - 10 Marks
Total - 50 Marks

SEE Evaluation:
Validity of Project Plan, Software Architecture and QA Plans - 20 Marks
Final Demonstration - 30 Marks
Project Report - 10 Marks
Viva - Voce - 10 Marks
Implementation of changes suggested during examination - 30 Marks
Total - 100 Marks

References:

1. Martin Fowler, Patterns of Enterprise Application Architecture (Addison Wesley Signature Series), 1st Edition, 2010 (Reprint).
2. Inderjeet Singh, Beth Stearns,Mark Johnson, and the Enterprise Team, Designing Enterprise Applications with the J2EETM Platform, Second Edition, Addison Wesley,2011 (Reprint).
3. John Kanalakis," Developing .NET Enterprise Applications, 1st Edition, Apress, 2003.
4. Yakov Fain, "Enterprise Development with Flex", O'Reilly Series, 1st Edition, 2010.
5. Jansch, "PHP/Architect's Guide to Enterprise PHP Development", Musketeers.me, LLC, 2008.
6. Steven Holzner, PHP: The Complete Reference, McGraw Hill Education; 1 edition (30 November 2007).
7. http://docs.oracle.com/javaee/6/firstcup/doc/gcrky.html
8. http://www.oracle.com/technetwork/developer-tools/jdev/j2eedev-084379.html
9. http://j2eetutorials.50webs.com/
10. http://www.webagesolutions.com/knowledgebase/waskb/waskb017/

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Implement the full stack development framework for web based applications. (PO-1, 3, 4)
2. Develop web applications using modern IDEs. (PO-1, 3, 4)
3. Apply advanced technologies used in application development. (PO-1, 3, 4)
4. Present the outcomes of the project in written and oral forms. (PO-2)
5. Evaluate the tools used in modern web application development. (PO-1, 3, 4)

CRYPTOGRAPHY AND INFORMATION SECURITY

Course Code: MSWEA1 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator: Dr Sumana M

Course Contents:

Unit I

The state of threats against computers, and networked Systems-Overview of computer security solutions and why they Fail-Vulnerability assessment, firewalls, **VPN's** –Overview of Intrusion Detection and Intrusion Prevention Network and Host-based IDS

Unit II

A General IDS model and taxonomy, Signature-based Solutions, Snort, Snort rules, Evaluation of IDS, Cost sensitive IDS **Anomaly Detection Systems and Algorithms**-Network Behavior Based Anomaly Detectors (rate based)-Host-based Anomaly Detectors-Software Vulnerabilities , State transition, Immunology, Payload Anomaly Detection.

Unit III

Case Studies Of Cryptography: Denial of service attacks, IP spoofing attacks, Secure inter branch payment transactions, Conventional Encryption and Message Confidentiality, Conventional Encryption Principles, Conventional Encryption Algorithms, Location of Encryption Devices, Key Distribution. Public Key Cryptography and Message Authentication: Approaches to Message Authentication, SHA-1, MD5, Public-Key Cryptography Principles, RSA, Digital, Signatures, Key Management.

Unit IV

DoS attacks. Web server and application vulnerabilities, SQL injection attacks, Vulnerability Analysis and Reverse Engineering, Buffer overflow attacks. Client-side browser exploits, Exploiting Windows Access Control Model for Local Elevation Privilege. Exploiting vulnerabilities in Mobile Application.

Unit V

Ethical hacking process, Hackers behavior & mindset, Maintaining Anonymity, Hacking Methodology, Information Gathering, Active and Passive Sniffing, Physical security vulnerabilities and countermeasures. Internal and External testing. Preparation of Ethical Hacking and Penetration Test Reports and Documents.

References:

1. Forouzan, B.A., Cryptography & Network Security. Tata McGraw-Hill Education , 2010.
2. Baloch, R., Ethical Hacking and Penetration Testing Guide, CRC Press, 2015.
3. Kahate, A. Cryptography and Network Security. McGraw-Hill Higher Ed., 2009.
4. Godbole, N., Information Systems Security: Security Management, Metrics, Frameworks and Best Practices. 1st Ed. John Wiley & Sons India, 2009.

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Identify threats on systems and access the vulnerabilities (PO-3,4).
2. Classify the various forms of intrusion detection systems (PO-3, 4).
3. Describe the crypto graphical techniques and their usage (PO-1, 3, 4).
4. Demonstrate different types of attacks and ways to control them. (PO-1, 3, 4).
5. Understand Ethical hacking and its approaches. (PO-1, 3, 4, 5).

ADVANCES IN DATABASES

Course Code: MSWEA2 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator: Dr Sumana M

Course Content:

Unit I

Review of Relational Data Model and Relational Database Constraints: Relational model concepts; Relational model constraints and relational database schemas; Update operations, anomalies, dealing with constraint violations, Types and violations. Overview of Object-Oriented Concepts – Objects, Basic properties. Advantages, examples, Abstract data types, Encapsulation, class hierarchies, polymorphism, examples.

Unit II

Object and Object-Relational Databases: Overview of OOP; Complex objects; Identity, structure etc. Object model of ODMG, Object definition Language ODL; Object Query Language OQL; Conceptual design of Object database. Overview of object relational features of SQL; Object-relational features of Oracle; Implementation and related issues for extended type systems; syntax and demo examples, The nested relational model. Overview of C++ language binding;

Unit III

Parallel and Distributed Databases: Architectures for parallel databases; Parallel query evaluation; Parallelizing individual operations; Parallel query optimizations; Introduction to distributed databases; Distributed DBMS architectures; Storing data in a Distributed DBMS; Distributed catalog management; Distributed Query processing; Updating distributed data; Distributed transactions; Distributed Concurrency control and Recovery.

Unit IV

Data Warehousing, Decision Support and Data Mining: Introduction to decision support; OLAP, multidimensional model; Window queries in SQL; Finding answers quickly; Implementation techniques for OLAP; Data Warehousing; Views and Decision support, View materialization, Maintaining materialized views. Introduction to Data Mining; Counting co-occurrences; Mining for rules; Tree-structured rules; ROC and CMC Curves; Clustering; Similarity search over sequences; Incremental mining and data streams; Additional data mining tasks.

Unit V

Enhanced Data Models for Some Advanced Applications: Active database concepts and triggers; Temporal, Spatial, and Deductive Databases – Basic concepts. More Recent Applications: Mobile databases; Multimedia databases; Geographical Information Systems; Genome data management.

References:

1. Elmasri and Navathe, Fundamentals of Database Systems, Pearson Education, 2013.
2. Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, 3rd Edition, McGraw-Hill, 2013.

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Select the appropriate high performance database like parallel and distributed database. (PO-1, 3,4)
2. Infer and represent the real world data using object oriented database. (PO-1, 3,4)
3. Interpret rule set in the database to implement data warehousing of mining. (PO-1, 3,4)
4. Discover and design database for recent applications database for better interoperability. (PO-1, 3,4)

MOBILE COMPUTING

Course Code: MSWEA3 **Credit: 4:0:0**
Prerequisite: Computer Networks, DBMS **Contact Hours: 56L**
Course Coordinator: Dr Naidila Sadashiv

Course Content:

Unit I

Introduction: Challenges in mobile computing, coping with uncertainties, resource poorness, bandwidth, etc. Cellular architecture, co-channel interference, frequency, reuse, capacity increase by cell splitting. Evolution of mobile system: CDMA, FDMA, TDMA, GSM. Wireless LAN: IEEE 802.11.

Unit II

Mobility Management: Cellular architecture, Co-channel interference, Mobility: handoff, types of handoffs; location management, HLR-VLR scheme, Mobile IP, Dynamic host configuration protocol, Mobile transport layer-Traditional and classical TCP.

Unit III

Databases: Database Hoarding Techniques, Data Caching, Transactional Models, Query Processing. Data Dissemination and Broadcasting Systems: Communication Asymmetry, Classification of Data-Delivery Mechanisms, Data Dissemination Broadcast Models, Selective Tuning and Indexing Techniques.

Unit IV

Data Synchronization in Mobile Computing Systems: Synchronization, Synchronization software for mobile devices, Synchronization protocols, SyncML - Synchronization language for mobile computing, Sync4J (Funambol), Synchronized Multimedia Markup Language (SMIL). Mobile Devices: Server and Management: Mobile agent, Application server, Gateways, Portals, Service Discovery, Device management, Mobile files systems, security.

Unit V

Support for Mobility- File Systems, Mobile operating systems; Features, services and interfacing modules of: Windows, Android, iOS, Linux for Mobile devices.

Text Books:

1. RajKamal, Mobile Computing,, Oxford University Press, 2nd Edition, 2012.
2. Jochen Schiller, Mobile Communications, 2nd Edition, Pearson 2003.

References:

1. Reza B, Mobile Computing Principles, Cambridge University press 2005.
2. B. P. Vijay Kumar and P. Venkataram ,Prediction-based location management using multilayer neural networks , Journal of The Indian Institute of Science, Vol. 82, No. 1, 2002.
3. B. P. Vijay Kumar and P. Venkataram, A Neural Network–Based Connectivity Management for Mobile Computing Environment, International Journal of Wireless Information Networks, Vol. 10, No. 2, 2003.
4. S.Acharya, M. Franklin and S. Zdonik, Balancing Push and Pull for Data Broadcast, Proceedings of the ACM SIGMOD, Tuscon, AZ, May 1997.
5. S.Acharya, M. Franklin and S. Zdonik, Disseminating Updates on Broadcast Disks, Proceedings of the 22nd VLDB Conference, Mumbai (Bombay), India, Sept 1996.

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Discuss the challenges and issues in mobile computing, and describe the basic principles and techniques, and protocol standards in wireless networks.(PO- 1, 3,4).
2. Describe the concept of network and transport layer for mobile networks in respect to mobility management. (PO-1, 4)
3. Analyze the database handling, data dissemination, synchronization in respect to Mobile data base and computing.(PO-1,3,4,5)
4. Describe and illustrate the mobile services, agents and mobility support using different file systems and platforms.(PO-1,2,3)
5. Develop mobile applications by analyzing their characteristics and Requirements by selecting the appropriate computing models and software architectures. (PO-3,4, 5)

PROBABILITY, STATISTICS AND QUEUEING THEORY

Course Code: MSWEB1 **Credit: 4:0:0**
Prerequisite: UG Mathematics **Contact Hours: 56L**
Course Coordinator: Dr N L Ramesh

Course Content:

Unit I

Random Variables: Random Variables (Discrete and Continuous), Probability density function, Cumulative distribution function, Mean, Variance, Moment generating function.

Discrete Probability Distributions: Binomial distribution, Poisson distribution,

Unit II

Continuous probability distributions. Normal distribution, Exponential distribution, Gamma distribution and Uniform distribution. **Joint probability distribution**: Joint probability distribution (both discrete and continuous), Conditional probability, Conditional expectation, Simulation of random variable.

Unit III

Stochastic Processes: Introduction, Classification of stochastic processes, Discrete time processes, Stationary, Ergodicity, Autocorrelation, Power spectral density. **Markov Chain:** Probability Vectors, Stochastic matrices, Regular stochastic matrices, Markov chains, Higher transition probabilities, Stationary distribution of Regular Markov chains and absorbing states, Markov and Poisson processes.

Unit IV

Queuing theory: Introduction, M/M/1 With infinite and finite capacity, M/M/K systems, M/G/1 queuing system , Engineering applications

Unit V

Sampling and Statistical Inference : Sampling, Sampling distributions, Standard error, Weak law of large numbers (without proof), Central limit theorem, Basics of parametric estimation, Test of Hypothesis for means, Confidence limits for means, Z-test,Test of significance of means and difference of means for large samples, Student's t-distribution, F-distribution, Chi-square distribution as a test of goodness of fit.

Text Books:

1. B.S.Grewal - Higher Engineering Mathematics - Khanna Publishers – 44th edition-2017.
2. R.E. Walpole, R. H. Myers, R. S. L. Myers and K. Ye – Probability and Statistics for Engineers and Scientists – Pearson Education – Delhi – 9th edition – 2012.

References:

1. Sheldon M. Ross –Introduction to Probability models – Academic Press – 2009.
2. Kishor S. Trivedi – Probability & Statistics with Reliability, Queuing and Computer Science Applications – John Wiley & Sons – 2nd edition – 2008.
3. T.Veerarajan - Probability, Statistics and Random process

Course outcomes (COs):

At the end of the course, students will be able to-

1. Analyze the given random data and their discrete probability distributions. (PO-1, 3)
2. Analyze the given random data, their continuous probability distributions and calculate the marginal & conditional distributions of bivariate random variables. (PO-1, 3)
3. Determine the parameters of stationary random processes and use Markov chain in prediction of future events. (PO-1, 3)
4. Determine the parameters involved in queuing models. (PO-1, 3)
5. Perform test of hypothesis for a population parameter. (PO-1, 3)

ADVANCED ALGORITHMS

Course Code: MSWEB2 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator: Dr Mydhili K Nair

Course Contents:

Unit I

Introduction: The Role of Algorithms in Computing – Getting started- Growth of functions – Recurrences – Probabilistic Analysis and Randomized algorithms – Heap sort – Quick Sort – Sorting in Linear Time – Elementary data structures – Binary Search Trees – Red black Trees.

Unit II

Advanced Design and Analysis Techniques: Dynamic Programming – Greedy algorithms – Amortized analysis – B-Trees – Binomial Heaps – Fibonacci Heaps. **Graph Algorithms**: Elementary Graph Algorithms – Minimum Spanning Trees – Single source Shortest path – All pairs Shortest paths.

Unit III

Sorting Networks – Matrix operations – Linear Programming – Polynomials and the FFT – Number – Theoretic Algorithms – String Matching – Computational Geometry – NP-Completeness – Approximation algorithms.

Unit IV

Parallel Algorithm Introduction – PRAM Model - Pointer Jumping - Performance evaluation of PRAM Algorithms - Comparison of PRAM Models - Sorting Machine - Relevance of the PRAM Model - **Sorting Networks :** Odd-Even Merge Sort - Sorting on a One-Dimensional Network.

Unit V

Algorithms on a Ring of Processors : Matrix-Vector Multiplication - Matrix-Matrix multiplication - A First Look at Stencil Applications - LU Factorization - A Second Look at Stencil Applications - Implementing Logical Topologies - Distributed vs. Centralized Implementations, Summary of Algorithmic Principles - **Algorithms on Grids of Processors** Logical Two-Dimensional Grid Topologies - Communication on a Grid of Processors - Matrix Multiplication on a Grid of Processors - Two-Dimensional Block Cyclic Data Distribution.

References:

1. Thomas H. Cormen, Charles E. Leiserson, Introduction to Algorithms, 2nd Edition, PHI, 2009.
2. Henri Casanova, Arnaud Legrand, Yves Robert; Parallel Algorithms, CRC press.

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Demonstrate the sorting and searching algorithms as well as recurring functions learnt and analyze them.(PO -1,3,4)
2. Apply the advanced algorithm design and analysis techniques learnt as well as graphical algorithms studied to write simple programs.(PO-1,4)
3. Identify and solve the computational aspects involved in matrix operations, polynomials and FFT number, NP-Completeness, computational geometry etc. (PO-1,3,4).
4. Demonstrate the parallel programming algorithms learnt such as PRAM, pointer jumping etc as well as identify sorting networks techniques. (PO-1,3,4)
5. Apply the various algorithms for ring and grid processors to write simple programs. (PO-1,3,4)

WEB SERVICES

Course Code: MSWEB3 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator: Dr Mydhili K Nair

Course Content:

Unit I

What are Web Services? Why are Web Services Important; Web Services & Enterprises; Moving Forward; **Service-Oriented Architecture:** Service Orientation in daily life; Evolution of SOA; Drivers for SOA; Dimensions of SOA; Key Components of SOA; Perspectives of SOA; **Enterprise-Wide SOA:** Considerations for Enterprise-Wide SOA, Strawman Architecture for Enterprise-Wide SOA, Enterprise SOA Layers, Application Development Process, SOA Methodology for Enterprise.

Unit II

XML Fundamentals: XML: The Lingua Franca of Web Services; XML Documents; XML Namespaces; XML Schema; Processing XML; **Introduction to XML:** Document Type Definitions, Namespaces, XML Schemas, Displaying Raw XML Documents, Displaying XML Documents with CSS, XSLT Style sheets, XML Processors

Unit III

SOAP and WSDL: The SOAP Model; SOAP; SOAP messages; SOAP Encoding; SOAP RPC;
WSDL; Using SOAP and WSDL; **Mobile and Wireless:** Mobile Wireless Services, Challenges with Mobile, J2ME Web Services

Unit IV

UDDI-Universal Description, Discovery and Integration: UDDI at a Glance; The UDDI Business Registry; UDDI Under the Covers; Accessing UDDI; How UDDI is Playing Out. **Transactions:** ACID Transactions, Distributed Transactions and Two-Phase Commit, Dealing with Heuristic Outcomes, Scaling Transactions to Web Services

Unit V

Workflow: Business Process Management, Workflows & Workflow Management Systems, Workflow Management Systems Drawbacks, Web Services and

Workflow; BPEL: BPEL Stack, Activities, Service Linking, Partners and Service References, Message Properties and Property Aliases, Correlating Messages, Containers and Data Handling, Workflow Example: Online Shop(Customer Web Service Not Included)

References:

1. Sandeep Chatterjee, Jeames Webber, Developing Enterprise Web Services – An Architect's Guide, Pearson Education, First Indian Reprint 2004
2. Robert W Sebesta, Programming the World Wide Web, Pearson Education, 4th Edition, Second Impression 2009
3. Shankar Kambhampaty, Service Oriented Architecture for Enterprise Applications, Wiley India Pvt Ltd, First Indian Edition 2008.
4. James McGovern et al: Java web Services Architecture, Elsevier, 2003
5. Thomas Erl, Service Oriented Architecture: Concepts, Technology and Design, Pearson Eductatin, Second Impression, 2008
6. Ben Margolis, SOA for the business developer, Shroff Publishers and Distributors Pvt Ltd(SPD), First Indian Reprint November 2007

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Illustrate Service Oriented Architecture (SOA) concepts and discuss how Web Services are the implementation of SOA.(PO-1,3,4)
2. Demonstrate XML concepts of DTD, Schema, CSS etc and discuss the importance of XML as a meta-data exchange language of Web Services. (PO-1,3,4)
3. Identify the technical details associated with WSDL and SOAP, two aspects vital for Web Services implementation and discuss the implementation of web services for mobile and wireless.(PO-1,3,4)
4. Illustrate the association of Web Services and UDDI as well as discuss the complexities associated with Transactions.(PO-1,3,4)
5. Identify Business Process Workflows and illustrate the use of BPEL.(PO-1,3,4)

II Semester

SOFTWARE METRICS AND QUALITY ENGINEERING

Course Code: MSWE21 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator: Mr Naresh E

Course Content:

Unit I

Fundamentals of measurements, Measurements- what is it and why do it? – Measurement in everyday life in Software Engineering, Scope of software metrics, the basics of measurements - Representational theory of measurement, measurement and models, measurement scale and scale types, a goal framework for software measurement – Classifying software measures, determining what to measure, applying the framework, software measurement validation and in practice.

Unit II

Software metrics data collection – What is good data, how to define the data, how and when to collect data, how to store and extract data, Analysing software measurement data – Analysing the results of experiments, examples of simple analysis techniques, more advanced methods, overview of statistical tests. Measuring internal software attributes: size – Aspects of software size, length, reuse, functionality, complexity.

Unit III

Software Quality - Perspective and Expectations, A Perspective on Testing, Basic definitions, Test Scenarios, Test cases, Insights from a Venn diagram, identifying test cases, Error, fault and Failure taxonomies, testing throughout the SDLC, Levels of testing, Activities of Test engineer, Test/Debug life cycle, testing principles, the cost of bugs, what makes a good software tester? Testing Axioms. Examples: The triangle problem, The NextDate function, the commission problem, The SATM (Simple Automatic Teller Machine) problem, the currency converter.

Unit IV

Functional Testing: Boundary value analysis, Robustness testing, Worst-case testing, Special value testing, Examples, Random testing, Equivalence classes, Equivalence test cases for the triangle problem, Decision tables, Test cases for the triangle problem. Compatibility testing, Usability testing, website testing, Testing the documentation. Case studies.

Unit V

Coverage-based Testing: Statement coverage testing, Condition coverage testing, Path coverage, computing cyclomatic complexity, exploratory testing. Static Testing: Reviews, Types of reviews, Inspections, Inspection process, Inspection roles, benefits of inspection, Checklists. Test Planning, Test Management, Test Process, Test Reporting. What is Test Automation? Terms used in Automation, Skills needed for automation, what to automate.

References:

1. Norman E. Fenton and Shari Lawrence Pfleeger, Software Metrics: A Rigorous Approach, PWS; 2nd edition, 1998.
2. Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3 Edition, Auerbach Publications, 2008.
3. Srinivas Sesikan and Ramesh Gopalswamy, "Principles of Software Testing", Pearson Education.
4. Stephan H. Kan, "Metrics and Models in Software Quality Engineering", Second Edition, Pearson Education.

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Identify the basics of measurement theory and its application to software (PO-1, 3, 4)
2. Measure the internal and external attributes of software (PO-3, 4)
3. Gain the knowledge of the basic definitions/concepts of Quality engineering and software testing. (PO-1)
4. Apply the concepts of validation and its techniques like boundary value analysis, equivalence class partitioning and decision table testing. (PO- 5)
5. Analyze the verification techniques like Reviews, Walkthroughs, checklists and Inspections in the development of software. Prepare the reports to track and monitor the defects. (PO-1, 4, 5)

SOFTWARE PROJECT MANAGEMENT

Course Code: MSWE22 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator: Mr Rajaram M Gowda

Course Content:

Unit I

Introduction, Contract & Technical project Management, Activities, Plans, Methods, Methodologies, objectives, business case, Success, failure, Management control, Traditional vs Modern project management, Project portfolio management, Project evaluation, Cost-benefit evaluation Techniques, Risk Evaluation, Resource allocation, Strategic management, Benefits, Step Wise Project Planning.

Unit II

Build/Buy? Methodologies, software processes, process models, prototyping, Incremental delivery, Atern, RAD, Agile methods, XP, Scrum, Selection of process model. Basis for software effort estimation, models, Expert judgment, Estimation by analogy, Albrecht FPA, FP Mark II, COSMIC FFP, COCOMO II, Cost estimation, Staffing pattern, Schedule compression, Capers Jones rules, When activity planning? Project schedules & activities, Sequencing & scheduling activities, Network Model, Time, Forward & Backward Pass, Critical path, Activity float, Shorten project duration, Critical activities, Activity on Arrow networks.

Unit III

Categories of risk, deal with risk, Risk identification, assessment, planning, Management. Evaluation of risks to the schedule, PERT, Monte Carlo, Critical chain. Nature of Resources, resource requirements, Scheduling resources, creating critical paths, Counting the cost, Publish resource schedule, cost schedules, scheduling sequence.

Unit IV

Monitoring and control Framework, Collect data, Review, Project termination, progress, cost monitoring, Earned Value Analysis, Prioritizing Monitoring, Get project back to target, Change control, Software Configuration Management, Managing contracts, Stages, terms of contract, contract management, Acceptance.

Unit V

Managing people, Understanding behavior, Organizational behavior, Selecting the right person, Best methods, Motivation, The Oldham-Hackman model, Stress, Ethical concerns. Becoming a team, Decisions, Organizational and Team structures, Coordination, Dispersed and Virtual teams, Communication genres, Communication plans, Leadership. Place and importance of quality, ISO 9126, Product and Process metrics, Product vs Process quality, Quality management systems, CMM, Enhance quality, Testing, Reliability, Quality plans.

References:

1. Bob Hughes, Mike Cotterell, Software Project Management, 4th Edition, Tata McGraw Hill Publications, 2006
2. Kathy Schwalbe, Information Technology Project Management, 5th Ed, Thompson, 2006
3. Watts S. Humphrey, Managing the Software Process, Addison-Wesley, 1989

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Classify and evaluate projects with illustrations. (PO-1,2,3,4)
2. Select a project approach, estimate effort and plan activities by analysis and evaluation.(PO-1,3,4)
3. Analyze, evaluate and manage risks and allocate resources(PO-1,3,4)
4. Analyze, evaluate, monitor and control projects and manage contracts. (PO-1,3,4)
5. Manage people, team work and manage software quality with illustrations .(PO-2,3,4,5)

CLOUD COMPUTING

Course Code: MSWE23 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator: Dr Siddesh G M

Course Content:

Unit I

Introduction: Network centric computing and network centric content, Peer-to-peer systems, Cloud Computing: an old idea whose time has come, Cloud Computing delivery models & Services, Ethical issues, Cloud vulnerabilities, Challenges, **Cloud Infrastructure:** Amazon, Google, Azure & online services, open source private clouds. Storage diversity and vendor lock-in, intercloud, Energy use & ecological impact of data centers, service level and compliance level agreement, Responsibility sharing, user experience, Software licensing.

Unit II

Cloud Computing: Applications & Paradigms, Challenges, existing and new application opportunities, Architectural styles of cloud applications, Workflows coordination of multiple activities, Coordination based on a state machine model - the Zoo Keeper, The Map Reduce programming model, Apache Hadoop, A case study: the GrepTheWeb application, Clouds for science and engineering, High performance computing on a cloud, Social computing, digital content, and cloud computing.

Unit III

Cloud Resource Virtualization: Layering and virtualization, Virtual machine monitors, Virtual machines Performance and security isolation, Full virtualization and paravirtualization, Hardware support for virtualization Case study: *Xen* -a VMM based on paravirtualization, Optimization of network virtualization in *Xen* 2.0, A performance comparison of virtual machines, Virtual machine security, The darker side of virtualization.

Unit IV

Cloud Resource Management and Scheduling: Policies and mechanisms for resource management, Stability of a two-level resource allocation architecture, Resource bundling; combinatorial auctions for cloud Scheduling algorithms for computing clouds, fair queuing, Start time fair queuing, borrowed virtual time.

Unit V

Storage systems: Evolution, Storage models, file systems, databases, DFS, General parallel File system, GFS, Hadoop, Locks & Chubby, Bigtable, Mega store. **Cloud security:** Risks, Security, privacy and privacy impacts assessments, Trust, VM Security, Security risks in shared images.

Text Book:

1. Dan Marinescu, Cloud Computing: Theory and Practice, 1st edition, MK Publishers, 2013.

References:

1. Kai Hwang, Jack Dongarra, Geoffrey Fox, Distributed and Cloud Computing, From Parallel Processing to the Internet of Things, MK Publishers.
2. Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, Cloud Computing: A Practical Approach, McGraw Fill, 2010.

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Understand the need of Cloud Computing along with various Models, Challenges & Infrastructure in Cloud domain.(PO-2,3,4)
2. Explore different Applications & Paradigms of Cloud Computing. (PO-2,3,4)
3. Analyze different virtualization techniques adopted in cloud domain along with case studies. (PO-2, 4)
4. Understand different mechanisms of Resource Management and Scheduling in Cloud Computing. (PO-2,3,4)
5. Analyze various data storage techniques and security aspects of Cloud Computing (PO-2,3,4)

ENTERPRISE APPLICATION DEVELOPMENT-II LAB

Course Code: MSWEL2 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator: Dr. Krishnaraj P M

Lab Exercises:

Students have to work in groups of three to develop an application using DevOps tools and demonstrate the continuous integration, version management and change management in AWS / GCS ecosystem.

CIE Evaluation:
Regular lab sessions based on adherence to project plan - 20 Marks
Final Demonstration - 15 Marks
Project Report - 5 Marks
Implementation of changes suggested during examination - 10 Marks
Total - 50 Marks

SEE Evaluation:
Validity of Project Plan, Software Architecture and QA Plans - 20 Marks
Final Demonstration - 30 Marks
Project Report - 10 Marks
Viva - Voce - 10 Marks
Implementation of changes suggested during examination - 30 Marks
Total - 100 Marks

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Implement applications using DevOps framework. (PO-1, 3, 4)
2. Develop and Integrate applications using modern IDEs. (PO-1, 3, 4)
3. Learn build and control management systems used in application development.(PO-1, 3, 4)
4. Present the outcomes of the project in written and oral forms. (PO-2)
5. Evaluate the tools used in modern application development. (PO-1, 3, 4)

INTERNET OF THINGS

Course Code: MSWEC1 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator Dr. Vijaya Kumar B P

Course Contents:

Unit I

Introduction to Internet of Things Definition & Characteristics of IoT, Physical Design of IoT, Things in IoT, IoT Protocols, Logical Design of IoT, IoT Functional Blocks, IoT Communication Models, IoT Communication APIs, IoT Enabling Technologies, Wireless Sensor Networks, Cloud Computing Big Data Analytics, Communication Protocols, Embedded Systems, IoT Levels and Deployment Templates, IoT Level-1, IoT Level-2, IoT Level-3, IoT Level-4, IoT Level-5, IoT Level-6.

Unit II

IoT and M2M : Introduction,M2M, Difference between IoT and M2M, SDN and NFV for IoT, Software Defined Networking, Network Function Virtualization, IoT System Management with NETCONF-YANG, Need for IoT Systems Management, Simple Network Management Protocol (SNMP), Limitations of SNMP, Network Operator Requirements, NETCONF, YANG IoT Systems Management with NETCONF-YANG, NETOPEER .

Unit III

IoT Platforms Design Methodology: IoT Design Methodology, Purpose & Requirements Specification, Process Specification, Domain Model Specification, Information Model Specification, Service Specifications, IoT Level Specification, Functional View Specification, Operational View Specification, Device & Component Integration, Application Development, **IoT Systems** - Logical Design using Python, Functions Modules, Packages, File Handling Operations Classes, Python Packages of Interest for IoT, JSON, XML, HTTPLib & URLLib, SMTPLib.

Unit IV

IoT Physical Devices & Endpoints, Embedded Boards, Interfaces, Serial SPI, I2C, Programming with Python, Controlling LED with embedded processor (like Raspberry Pi, aurdino, etc.,) Interfacing an LED and Switch, Interfacing a Light Sensor (LDR), Other IoT Devices, pcDuino, Beagle Bone Black, Cubie board. IoT Physical Servers & Cloud Offerings, Cloud storage Models & communication

API's, WAMP - AutoBahn for IoT, Xively Cloud for IoT, Python Web Application Framework – Django, Django Architecture, Starting Development with Django, Designing a RESTful Web API,Web Services for IoT, IoT Messaging Platform.

Unit V

Case Studies Illustrating IoT Design, Home Automation Smart Lighting, Home Intrusion Detection, Smart Parking, Weather Monitoring System 9.4.2 Weather Reporting, Air Pollution Monitoring, Forest Fire Detection, Agriculture, Smart Irrigation, Productivity Applications.
Data Analytics for IoT, Apache Hadoop, Map Reduce Programming Model, Hadoop Map Reduce Job Execution, Map Reduce Job Execution Workflow, Hadoop Cluster Setup, Using Hadoop Map Reduce for Batch Data Analysis, Hadoop YARN, Apache Oozie, Setting up Oozie, Oozie Workflows for IoT Data Analysis, Apache Spark, Apache Storm, Setting up a Storm Cluster, Using Apache Storm for Real-time Data Analysis, REST-based approach, Web Socket-based approach

Text Books:

1. Arshdeep Bagha ,Vijay Madisetti , Internet of Things (A Hands-on-Approach) by University press Aug 2015.
2. Dirk Slama, Frank Puhlmann, Jim Morrish, Rishi M Bhatnagar, Enterprise IoT: Strategies and Best Practices for Connected Products and Services. O'Reilly Media, Sep 2015.
3. Hillar, Internet of Things with Python, Packet book.

Course Outcomes (COs):

At the end of the course, students will be able to:

1. Explain the design principles and issues involved in IoT and their Standards. (PO-3, 4, 5)
2. Identify the networking functions and the management of IoT systems. (PO-3,4)
3. Distinguish and apply the design techniques with cloud and embedded solution for IoT. (PO-1, 3,4)
4. Illustrate the implementation and interfacing methodologies for IoT frameworks. (PO-3, 4)
5. Design and illustrate the application specific IoT system and perform data analytics, along with the project reports. (PO-1, 2, 3, 4)

DEEP LEARNING

Course Code: MSWEC2 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator Dr. Sumana M

Course Content:

Unit I

Introduction: Human brain, neuron models, neural nets as directed graphs, feedback, neural architectures, knowledge representation, Learning Process, Learning Tasks.

Unit II

Multilayer Perceptrons: Introduction , Some Preliminaries , Batch Learning and On-Line Learning ,The Back-Propagation Algorithm , XOR Problem ,Heuristics for Making the Back-Propagation Algorithm Perform Better , Back Propagation and Differentiation , The Hessian and Its Role in On-Line Learning, Cross-Validation , Virtues and Limitations of Back-Propagation Learning.

Unit III

Convolutional Neural Networks:- The Convolution Operation ,Motivation ,Pooling ,Convolution and Pooling as an Infinitely Strong Prior , Variants of the Basic Convolution Function , Structured Outputs , Data Types , Efficient Convolution Algorithms , Random or Unsupervised Features, The Neuroscientific Basis for Convolutional Networks.

Unit IV

Sequence Modeling: Recurrent and Recursive Nets , Unfolding Computational Graphs , Recurrent Neural Networks Bidirectional RNNs ,Encoder-Decoder Sequence-to-Sequence Architectures ,Deep Recurrent Networks , Recursive Neural Networks , The Challenge of Long-Term Dependencies, The Long Short-Term Memory and Other Gated RNNs.

Unit V

Autoencoders: Under complete Autoencoders, Regularized Autoencoders, Denoising Autoencoders, Learning Manifolds with Autoencoders, Contractive Autoencoders, Predictive Sparse Decomposition Applications of Autoencoders.

Applications : Large-Scale Deep Learning , Computer Vision ,Speech Recognition , Natural Language Processing , Other Applications , Deep Belief Networks, Learning Vectorial Representations of Words.

References:

1. Simon Haykin, Neural networks: A comprehensive foundation, Second Edition, Prentice Hall, New Delhi, 1999, ISBN-81-203-2373-4.
2. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press, 2016.
3. Deng & Yu, Deep Learning: Methods and Applications, Now Publishers, 2013.
4. Josh Patterson & Adam Gibson, Deep Learning – A Practitioners Approach, O'Reilly, 1st Edition 2017.

Course Outcomes (COs):

At the end of the course, students will be able to :

1. Explain knowledge representation and learning in neural networks (PO-3, 4)
2. Design the back-propagation algorithm, its virtues and limitations, and understand its role as an optimum method for computing partial derivations. (PO-2,3,4)
3. Demonstrate construction of convolutional neural networks for images. (PO-1,2,3,4)
4. Design and develop recurrent neural networks for processing sequential data. (PO-1, 3,4)
5. Illustrate autoencoder neural networks to perform unsupervised learning by applying backpropagation. (PO-1,2,3,4)

ADVANCES IN OPERATING SYSTEMS

Course Code: MSWEC3 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator Dr. Sumana M

Course Content:

Unit I

Operating System Overview, Process description & Control: Operating System Objectives and Functions, The Evolution of Operating Systems, Major Achievements, Developments Leading to Modern Operating Systems, Microsoft Windows Overview, Traditional UNIX Systems, Modern UNIX Systems, what is a Process? Process States, Process Description, Process Control, Execution of the Operating System, Security Issues.

Unit II

Threads, SMP, and Microkernel, Virtual Memory: Processes and Threads, Symmetric Multiprocessing (SMP), Micro Kernels, Windows Vista Thread and SMP Hours Management, Linux Process and Thread Management. Hardware and Control Structures, Operating System Software, UNIX Memory Management, Windows Vista Memory Management, Summary.

Unit III

Multiprocessor and Real-Time Scheduling: Multiprocessor Scheduling, Real-Time Scheduling, Linux Scheduling, UNIX PreclsSl) Scheduling, Windows Vista Hours Scheduling, Process Migration, Distributed Global States, Distributed Mutual Exclusion, Distributed Deadlock

Unit IV

Embedded Operating Systems: Embedded Systems, Characteristics of Embedded Operating Systems, eCOS, TinyOS, Computer Security Concepts, Threats, Attacks, and Assets, Intruders, Malicious Software Overview, Viruses, Worms, and Bots, Rootkits.

Unit V

Kernel Organization: Using Kernel Services, Daemons, Starting the Kernel, Control in the Machine, Modules and Device Management, MODULE Organization, MODULE Installation and Removal, Process and Resource

Management, Running Process Manager, Creating a new Task, IPC and Synchronization, The Scheduler , Memory Manager , The Virtual Address Space, The Page Fault Handler , File Management. The windows NT/2000/XP kernel: Introduction, The NT kernel, Objects , Threads, Multiplication Synchronization, Traps, Interrupts and Exceptions, The NT executive , Object Manager, Process and Thread Manager , Virtual Memory Manager, I/o Manager.

Text Books:

1. William Stallings, Operating Systems: Internals and Design Principles, 6th Edition, Prentice Hall, 2013.
2. Gary Nutt, Operating Systems, 3rd Edition, Pearson, 2014.

References:

1. Silberschatz, Galvin, Gagne: Operating System Concepts, 8th Edition, Wiley, 2008
2. Andrew S. Tanenbaum, Albert S. Woodhull: Operating Systems, Design and Implementation, 3rd Edition, Prentice Hall, 2006.
3. Pradeep K Sinha: Distribute Operating Systems, Concept and Design, PHI, 2007.

Course Outcomes (COs):

At the end of the course, students will be able to:

1. Demonstrate the Mutual exclusion, Deadlock detection and agreement protocols of Distributed operating system. (PO-1,3)
2. Learn the various resource management techniques for distributed systems. (PO-1,3)
3. Identify the different features of real time and mobile operating system. (PO-1,3)
4. Modify existing open source kernels in terms of functionality or features used. (PO-1,3)

COGNITIVE COMPUTING

Course Code: MSWED1 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator Dr. Vijaya Kumar B P

Course Content:

Unit I

Introduction: Foundation of Cognitive Computing, On platforms- machine learning, reasoning, natural language processing, speech recognition and vision (object recognition), human–computer interaction, dialog and narrative generation, among other technologies. Features: Adaptive, Interactive, Iterative and stateful, Context aware.

Unit II

Design Principles for Cognitive Systems, Sentiment analysis, Natural language processing, text analysis, computational linguistics, and biometrics to systematically identify, extract, quantify, and study affective states and subjective information. Sentiment analysis to voice of the customer materials such as reviews and survey responses, online and social media, and healthcare materials for applications.

Unit III

Cognitive analytics, Relationship Between Big Data and Cognitive Computing Representing Knowledge in Taxonomies and Ontologies, Applying Advanced Analytics to Cognitive Computing, Using Machine Learning and Deep Learning Neural Networks to Model Cognition.

Unit IV

Role of Cloud and Distributed Computing in Cognitive Computing. The Business Implications of Cognitive Computing, Cognitive analytics- Cognitive computing-technology platforms, Word processing documents, emails, videos, images, audio files, presentations, webpages, social media and many other data formats often need to be manually tagged with metadata before they can be fed to a computer for analysis and insight generation. Principal benefit of utilizing cognitive analytics over traditional big data analytics.

Unit V

Process of building a Cognitive Application, Building a Cognitive Healthcare Application, Smarter Cities: Cognitive Computing in Government, Emerging Cognitive Computing Areas, Future Applications for Cognitive Computing.

References:

1. Judith S. Hurwitz, Marcia Kaufman, Adrian Bowles, Cognitive Computing and Big Data Analytics, Wiley 2015.
2. Mark Watson, Introduction to Cognitive Computing, 2018, Copyright 2016-2018 Mark Watson.
3. Vijay V Raghavan, Venkat N. Gudivada, Venu Govindaraju, C.R. Rao, Cognitive Computing: Theory and Applications, Volume 35 (Handbook of Statistics) 1st Edition, 2018.

Course Outcomes (COs):

At the end of the course, students will be able to-

1. Classify and interpret the features of cognitive system and explain the necessary platforms used for cognitive computing. (PO-2, 3, 4, 5)
2. Design the cognitive systems and illustrate the methodologies used for some of the applications using cognitive computing (PO-2, 3, 4)
3. Analyze and quantify the concepts used to model the cognition and ontologies for knowledge representation and implementation using ANN. (PO-1, 3, 4, 5)
4. Apply and compare the cognitive computing with Cloud and distributed Computing platforms and their implications. (PO-3, 4)
5. Build the cognitive applications in some of social relevant areas. (PO-1, 2, 3, 4)

SYSTEM PERFORMANCE AND ANALYSIS

Course Code: MSWED2 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator Dr. Myna A N

Course Content:

Unit I

AN OVERVIEW OF PERFORMANCE EVALUATION: Introduction, Common Mistakes in Performance Evaluation, A Systematic Approach to Performance Evaluation, Selecting an Evaluation Technique, Selecting Performance Metrics, Commonly used Performance Metrics, Utility Classification of Performance Metrics, Setting Performance Requirements.

Unit II

WORKLOAD SELECTION AND CHARACTERIZATION: Types of Workloads, Addition instructions, Instruction Mixes, Kernels; Synthetic programs, Application Benchmarks, Popular benchmarks, Workload Selection: Services exercised, level of detail, Representativeness, Timeliness, Other considerations in Workload selection, Workload Characterization & Techniques: Terminology, Averaging, Specifying dispersion, Single Parameter Histograms, Multi parameter histograms, Principal Component Analysis, Markov Models, Clustering.

Unit III

MEASUREMENT TECHNIQUES AND TOOLS: Monitors: Terminology and classification, software and hardware monitors, Software versus hardware monitors, firmware and hybrid monitors, Distributed System Monitors, Program-Execution Monitors and Accounting Logs: Program Execution Monitors, Techniques for Improving Program Performance, Accounting Logs, Analysis and Interpretation of Accounting log data, Using accenting logs to answer commonly asked questions.

Unit IV

CAPACITY PLANNING, BENCHMARKING AND EXPERIMENTAL DESIGN: Steps in capacity planning and management, Problems in capacity planning, Common mistakes in benchmarking, Remote-Terminal Emulation, Components of an RTE, Limitations of RTEs, Experimental design and analysis: Terminology, Common mistakes in experiments, Types of Experimental designs, 2^k factorial designs, concepts, computation of effects, Sign table method for computing effects, allocation of Variance, General 2^k factorial designs.

Unit V

SIMULATION: Introduction to Simulation, Analysis of Simulation Results: Model Verification Techniques: Top-down Modular design, Antibugging, Structured Walk-through, Deterministic Models, Run simplified cases, Trace, Online graphic displays, Continuity Tests, Degeneracy Tests, Consistency Tests, Seed Independence, Model Validation Techniques: Expert Intuition, Real System Measurements, Theoretical Results, Transient Removal: Long Runs, Proper Initialization, Truncation, Initial Date Deletion, Moving Average of Independent Replications, Batch Means, Terminating Simulation, Stopping Criteria, Variance Reduction.

References:

1. Raj Jain. "The Art of Computer Systems Performance Analysis". John Wiley and sons, New York, USA, 1991.
2. Law A M and Kelton W.D. "Simulation Modeling and Analysis ", McGraw Hill, New York, USA, 1991
3. Paul J Fortier, Howard E Michel: Computer Systems Performance Evaluation and Prediction, Elsevier, 2003

Course Outcomes (COs):

At the end of the course, students will be able to:

1. Identify the concepts of performance evaluation. (PO-1,3,4)
2. Select proper workload and characterize the workload. (PO-1,3,4)
3. Analyze performance statistics, data and display results using monitors. (PO-1,3,4)
4. Design performance experiments. (PO-1,3,4)
5. Perform computer system performance analysis using simulation. (PO-1,3)

BLOCKCHAIN AND CYBER SECURITY

Course Code: MSWED3 **Credit: 4:0:0**
Prerequisite: NIL **Contact Hours: 56L**
Course Coordinator Dr. Sumana M

Course Content:

Unit I

Cyber Threat Landscape and Security Challenges: Current threat landscape, Ransomware, Distributed denial-of-service (DDoS) attacks, Insider threats, Data breaches, Advanced persistence threat (APT), Defender perspectives, Live attack execution, Emerging security challenges. The security ecosystem, The zero-trust approach, The assume breach approach.

Unit II

Introducing Blockchain and Ethereum: What is blockchain? A brief history, Fundamentals of the blockchain ,Who is using blockchain and how? Internet versus blockchain, IP packet versus block ,Web app versus dApp,,How blockchain works, The building blocks of blockchain - Block, Cryptography digital signature and hashing algorithm, Consensus : the core of blockchain, Ethereum: History, What is Ethereum? Smart contract, EVM, Gas, dApp, Private versus public blockchain, Public blockchain , Private blockchain, Business adaptation.

Unit III

Hyperledger: Blockchain for Businesses: Technical requirements, Hyperledger overview, Blockchain-as-a-service (BaaS), Program goal, Architecture and core components , Hyperledger Fabric model, Hyperledeger Fabric core components, Workings of Hyperledger and transaction processing, Bitcoin versus Ethereum versus Hyperledger, Hyperledger Fabric capabilities, Tuna application.

Unit IV

Blockchain on the CIA Security Triad, What is the CIA security triad?, Understanding blockchain on confidentiality, Blockchain on integrity, Understanding blockchain on availability, Deploying PKI-Based Identity with Blockchain, Components, Architecture, Challenges of the existing PKI model, How can blockchain help? Decentralized infrastructure, Deployment method, Testing.

Unit V

Two-Factor Authentication with Blockchain, Blockchain for 2FA, Blockchain-Based DNS Security Platform: DNS, Understanding DNS components, DNS structure and hierarchy: Root name server, Current TLD structure, Registries, registrars, and registrants, DNS records ,DNS topology for large enterprises, Challenges with current DNS, Blockchain-based DNS solution, Deploying Blockchain-Based DDoS Protection: DDoS attacks, Types of DDoS attacks, Facts about Blockchain and Cyber Security.

References:

1. Rajneesh Gupta. "Hands-On Cybersecurity with Blockchain: Implement DDoS protection, PKI-based identity, 2FA, and DNS security using Blockchain" , Packt Publishing, 2018.
2. Imran Bashir, "Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained", 2nd Edition, Packt Publishing, 2018.

Course Outcomes (COs):

At the end of the course, students will be able to:

1. Understand the cyberthreat landscape. (PO- 1,3)
2. Build Blockchains using Ethereum and Hyperledger. (PO-1,4)
3. Program Blockchain solutions and build Blockchain-based apps for 2FA, and DDoS protection. (PO-1,3,4)
4. Develop Blockchain-based PKI solutions and apps for storing DNS entries. (PO-1,3,4)
5. Identify the challenges and the future of cybersecurity and Blockchain. (PO-1,2,4)