SCHEME & SYLLABUS FOR MASTER OF TECHNOLOGY (M.TECH.) IN SOFTWARE ENGINEERING (SE) PROGRAM AT U.I.E.T.

As per AICTE Model curriculum

(Applicable w.e.f. session 2018-2019 in Phased Manner)

Scheme for the course of Master of Technology (M.Tech.) in Software Engineering (Credit Based)
(Applicable from session 2018-2019)
Semester-I

S. No.	Course No.	Subject	Teaching Schedule			Hours/Week	Examination Schedule & Percentage Distribution				Duration of Exam (Hrs.)	Credit
			L	T	P		Major Test	Minor Test	Practical	Total		
1	MTSE-101	Essentials of Software Engineering	3	0	0	3	60	40	--	100	3	3
2	MTSE-103	Modeling and Simulation	3	0	0	3	60	40	--	100	3	3
3	*	Program Elective-I	3	0	0	3	60	40	--	100	3	3
4	**	Program Elective-II	3	0	0	3	60	40	--	100	3	3
5	MTSE-117	Software Engineering Lab	0	0	4	4	--	40	60	100	2	2
6	MTSE-119	Agile Software Engineering Lab	0	0	4	4	--	40	60	100	2	2
7	MTRM-111	Research Methodology and IPR	2	0	0	2	60	40	--	100	--	2
8	***	Audit course-I	2	0	0	2	--	100	--	100	3	
Total						**24**	**300**	**280**	**120**	**700**	**-**	**18**

*Programme Elective-I		**Programme Elective-II	
Course No.	**Subject**	**Course No.**	**Subject**
MTSE-105	Software Project Management	MTSE-111	Software Reliability
MTSE-107	Agile Software Process	MTSE-113	Software Agents
MTSE-109	Software Process Maturity Model	MTSE-115	Human Interface System Design

***Audit Course-I	
Course No.	**Subject**
MTAD-101	English for Research Paper Writing
MTAD-103	Disaster Management
MTAD-105	Sanskrit for Technical Knowledge
MTAD-107	Value Education

Note: 1.The course of program elective will be offered at 1/3rd or 6 numbers of students (whichever is smaller) strength of the class.
2. ***Along with the credit course, a student may normally be permitted to take audit course, however for auditing a course; prior consent of the course coordinator of the course is required. These courses shall not be mentioned for any award/calculation of SGPA/CGPA in the DMC. A certificate of successful completion of the audit course will be issued by the Director/Head of institution.

Scheme for the course of Master of Technology (M.Tech.) in Software Engineering
Semester-II

S. No.	Course No.	Subject	Teaching Schedule			Hours/Week	Examination Schedule & Percentage Distribution				Duration of Exam (Hrs.)	Credit
			L	T	P		Major Test	Minor Test	Practical	Total		
1	MTSE-102	Software Risk Management	3	0	0	3	60	40	--	100	3	3
2	MTSE-104	Social Networks	3	0	0	3	60	40	--	100	3	3
3	*	Program Elective-III	3	0	0	3	60	40	--	100	3	3
4	**	Program Elective-IV	3	0	0	3	60	40	--	100	3	3
5	MTSE-118	Software Quality Models & Testing Lab	0		4	4	--	40	60	100	3	2
6	MTSE-120	Social Networks Lab	0		4	4	--	40	60	100	3	2
7	#MTSE-122	Mini Project	0	0	4	4	--	100	--	100	--	2
8	***	Audit course-II	2	0	0	2	--	100	--	100	3	
Total						**26**	**240**	**340**	**120**	**700**	**-**	**18**

*Programme Elective -III		**Programme Elective-IV	
Course No.	**Subject**	**Course No.**	**Subject**
MTSE-106	Cloud Computing	MTSE-112	Object Oriented Programming
MTSE-108	Software Testing & Quality Assurance	MTSE-114	Pattern Oriented Software Architecture
MTSE-110	Data Warehousing and Data mining	MTSE-116	Software Measurement and Metrics

List of Audit Course-II (AC-II) for Second Semester	
Course No.	**Subject**
MTAD-102	Constitution of India
MTAD-104	Pedagogy Studies
MTAD-106	Stress Management by Yoga
MTAD-108	Personality Development through Life Enlightenment Skills.

Note 1: After the second semester exams, the students are encouraged to go to Industrial Training/Internship for at least 6-8 weeks during the summer break with a specific objective for Dissertation Part–I (MTSE-207). The industrial Training/Internship would be evaluated as the part of the Dissertation–I (with the marks distribution as 40 marks for Industrial Training/Internship and 60 marks for Dissertation Part–I).

Note 2: The course of program elective will be offered at 1/3rd or 6 numbers of students (whichever is smaller) strength of the class.

***Note 3:** Along with the credit course, a student may normally be permitted to take audit course, however for auditing a course; prior consent of the course coordinator of the course is required. These courses shall not be mentioned for any award/calculation of SGPA/CGPA in the DMC. A certificate of successful completion of the audit course will be issued by the Director/Head of institution.

#Note 4: Mini project: During this course the student will be able to understand the contemporary/emerging technologies for various processes and systems. During the semester, the students are required to search/gather the material/information on a specific topic, comprehend it and present/discuss the same in the class. He/she will be acquainted to share knowledge effectively in oral (seminar) and written form (formulate documents) in the form of report. The student will be evaluated on the basis of viva/ seminar (40 marks) and report (60 marks).

Semester: III

S. No.	Course No.	Subject	Teaching Schedule			Hours /Week	Examination Schedule & Percentage Distribution				Duration of Exam (Hrs.)	Credit
			L	T	P		Major Test	Minor Test	Practical	Total		
	*	Program Elective -V	3	0	0	3	60	40		100	3	3
1	**	Open Elective	3	0	0	3	60	40		100	3	3
2	MTSE-207	Dissertation Part-I	0	0	20	20	--		100	100	3	10
Total							**120**	**80**	**100**	**300**		**16**

Programme Electives -V	
Course No.	**Subject**
MTSE-201	Software Quality Management
MTSE-203	Language Technologies
MTSE-205	Personal Software Process

****Open Elective**		
1.	MTOE-201	Business Analytics
2.	MTOE-203	Industrial Safety
3.	MTOE-205	Operations Research
4.	MTOE-207	Cost Management of Engineering Projects
5.	MTOE-209	Composite Materials
6.	MTOE-211	Waste to Energy

Semester: IV

S. No.	Course No.	Subject	Teaching Schedule			Hours/Week	Examination Schedule & Percentage Distribution				Duration of Exam (Hrs.)	Credit
			L	T	P		Major test	Minor test	Practical	Total		
1	MTSE-202	Dissertation Part-II	0	0	32	16	0	100	200	300	3	16
Total						**16**		**100**	**200**	**300**		**16**

Total Credits – 68

Note 1: At the end of the second semester each student is required to do his/her Dissertation work in the identified area in consent of the Guide/Supervisor. Synopsis for the Dissertation Part-I is to be submitted within three weeks of the beginning of the Third Semester.

Note 2: Each admitted student is required to submit the report of his/her Dissertation Part-I as per the schedule mentioned in Academic calendar for the corresponding academic session otherwise the Dissertation Part-II cannot be continued at any level.

Note 3: Each admitted student is required to submit his/her final Dissertation Part-II as per the schedule mentioned in Academic calendar for the corresponding academic session only after the publication of two papers in a journal/International/National conference of repute like IEEE, Springer, Elsevier, ACM etc.

Note 4: The course of program/open elective will be offered at 1/3rd or 6 numbers of students (whichever is smaller) strength of the class.

MTSE-101	Essentials of Software Engineering							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	The main purpose of this course is to impart knowledge on the basic principles of software development life cycle.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To understand the software life cycle models							
CO2	To understand the importance of the software development process							
CO3	To understand the importance of modeling and modeling languages							
CO4	To design and develop correct and robust software products							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

Unit-1

Principles and motivation: History, Definitions, why engineered approach to software development, Software Development Process Models from the point of view of technical development and project management: Waterfall, Rapid Prototyping, Incremental Development, Spiral Model, Emphasis on computer assisted environment.

Software development methods: Formal, semi-formal and informal methods, Requirements elicitation, Requirement specification, Data, functions and event based modeling, Some of the popular methodologies such as Yourdon's SAD, SSADM etc., CASE tools classification, features, strengths and weaknesses, CASE: CASE standards.

Unit-2

Software Project Management: Principles of Software Project Management, Organizational and team structure, Project planning, Project Initiation and Project Termination, Technical, Quality and Management plans, Project Control, Project Estimation methods, Function points and COCOMO.

Unit-3

Software Quality Management: Quality Control, Quality Assurance and Quality Standards with emphasis on ISO 9000, Functions of Software QA organization dose in Project, Interaction with developers, Quality plans, Quality assurance towards quality improvement, Role of independent Verification and Validation, Total Quality Management, SEI maturity model, Software metrics.

Unit-4

Configuration Management: Need for Configuration Management, Configuration Management functions and activities, Configuration Management Techniques, Examples and Case studies.

Software Engineering Standards: Government Standards, IEEE (and other professional bodies) standards, Corporate Standards.

Reference books:

1. Eisner Howard, Computer Aided System Engineering, Prentice Hall, New Jersy.
2. Richard Fairly, Software Engineering Concept, Mc-Graw Hill, New York.
3. Pankaj Jalote, An Integrated Approach to Software Engineering, Narosa Pub. House, New Delhi.
4. Roger Pressmen, Software Engineering: A Practitioner's Approach McGraw Hill, New York.
5. Carlo Ghezzi, Mehdi Jazayeri, Dino Manlrioli, Fundamentals of Software Engineering Prentice Hall New Jersy.
6. Dong Bell, Ian Morrey, and Pugh, Software Engineering: A programming Approach Prentice Hall, New Jersy.
7. Kenneth Shere, Software Engineering and Management, Prentice-Hall, New Jersy.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-103	Modelling and Simulation							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	This course will look at professional techniques for understanding, assessing and applying the software simulation models in software development systems.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To appreciate and understand scientific concepts of Software and Hardware design.							
CO2	To apply different simulation Models in Software Development							
CO3	To emphasize the Application of Simulation Models							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT-1

Systems: Models types, principles used in modelling, system studies, interacting subsystems andexample, simulation definition, examples, steps in computer simulation, advantages and disadvantagesof simulation, simulation study, classification of simulation languages.

System Simulation:

Techniques of simulation, monte carlo method, comparision of simulation and analytical methods, numerical computation techniques for continuous and discrete models, distributed leg models, cobweb models.

UNIT-II

Continuous system simulation:

Continuous system models, differential equation, analog computer analog methods, digital analog simulators, CSSLS, CSMPIII language.

System Dynamics: Historical background, exponential, Growth and decay models, modified exponential growth models, logistic curves and generalization of growth models, system dynamics diagrams, dynamo language.

UNIT-III

Probability concepts in simulation:

Stochastic variables, discrete and continuous probability function, continuous uniform distributed and computer generation of random numbers, uniform random number generator, non uniform continuously distributed random numbers, rejection method.

Discrete system simulation: Discrete events, representation of time, generation of arrival patterns, simulation of telephone system, delayed calls, simulation programming tasks, gathering statistics, discrete simulation languages.

UNIT-IV

Object Oriented approach in simulation, simulation in C++, Introduction to GPSS, general description, action times, choice of paths, simulation of a manufacturing shop, facilities andstorage, program control statements, priorities and parameters, numerical attributes, functions, simulation of a supermarket transfer models, GPSS model applied to any application, simulationprogramming techniques like entry types.

Reference books

1. G.Gordan “System Simulation” , 2ndEd, 2002 PHI.
2. T.A. Payer “Introduction to Simulation”, McGraw Hill.
3. W.A. Spriet “Computer Oriented Modeling and Simulation”.
4. Narsingh Deo “System Simulation with Digital Computers”, PHI.
5. V. Rajaraman “Analog Simulation”, PHI
6. Law & Kelton “Simulation Modelling and Analysis” 3 rd Ed., 2000, McGraw Hill.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-105	Software Project Management							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	The course gives an insight of the most commonly used software architecture and design patterns and their applications							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To understand Software Project Models and Software Management Concepts.							
CO2	To understand the various methods of Cost Estimation.							
CO3	To Study about Software Quality Management.							
CO4	To understand Project Evaluation.							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I - PROJECT CONCEPTS AND ITS MANAGEMENT

Project life cycle models-ISO 9001 model-Capability Maturity Model-Project Planning-Project tracking-Project closure. Evolution of Software Economics – Software Management Process Framework: Phases, Artifacts, Workflows, Checkpoints – Software Management Disciplines: Planning / Project Organization and Responsibilities / Automation / Project Control – Modern Project Profiles.

UNIT II - COST ESTIMATION

Problems in Software Estimation – Algorithmic Cost Estimation Process, Function Points, SLIM (Software Life cycle Management), COCOMO II (Constructive Cost Model) – Estimating Web Application Development – Concepts of Finance, Activity Based Costing and Economic Value Added (EVA) – Balanced Score Card.

UNIT III - OFTWARE QUALITY MANAGEMENT

Software Quality Factors – Software Quality Components – Software Quality Plan – Software Quality Metrics – Software Quality Costs – Software Quality Assurance Standard – Certification – Assessment.

UNIT IV - PROJECT EVALUATION AND EMERGING TRENDS

Strategic Assessment–Technical Assessment–Cost Benefit Analysis–Cash Flow Forecasting–Cost Benefit Evaluation Technique–Risk Evaluation–Software Effort Estimation. Emerging Trends: Import of the internet on project Management – people Focused Process Models.

REFERENCES

1. Ramesh Gopalaswamy , *"Managing and global Software Projects"*, Tata McGraw Hill Tenth Reprint, 2011.
2. Roger S.Pressman, *"Software Engineering- A Practitioner's Approach",* 7th Edition ,McGraw Hill, 2010.
3. Daniel Galin, "*Software Quality Assurance: from Theory to Implementation*", Addison-Wesley, 2003.
4. Bob hughes and Mike Cotterell, *"Software Project Management"* second edition,1999.
5. Royce, W. *"Software Project Management: A Unified Framework",* Addison- Wesley, 1998.
6. Demarco, T. and Lister, T. *"Peopleware: Productive Projects and Teams,* 2nd Ed.", Dorset House,1999.
7. Fenton, N.E., and Pfleeger, S.L.. *"Software Metrics: A Rigorous and Practical Approach,* Revised" Brooks Cole, 1998.
8. Kaplan, R.S., Norton, D.P. *"The Balanced Scorecard: Translating Strategy into Action",* Harvard Business School Press, 1996.
9. Boehm, B. W. *"Software Risk Management: Principles and Practices"* in IEEE Software, January 1991, pp32-41.
10. Grant, J.L. *"Foundations of Economic Value Added"*, John Wiley & Sons,
1997.
11. Cooper, R., *"The Rise of Activity-Based Costing- PartOne: What is an Activity-Based Cost System"* Journal of Cost Management, Vol.2, No.2(Summer 1988), pp.45 – 54.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-107	Agile Software Process							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
	0	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	This course imparts knowledge to students in the basic concepts of Agile Software Process, methodology and its development.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To understand the basic concepts of Agile Software Process.							
CO2	To gain knowledge in the area of various Agile Methodologies.							
CO3	To develop Agile Software Process.							
CO4	To know the principles of Agile Testing.							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-INTRODUCTION

Software is new product development – Iterative development – Risk-Driven and Client-Driven iterative planning – Time boxed iterative development – During the iteration, No changes from external stakeholders – Evolutionary and adaptive development - Evolutionary requirements analysis – Early "Top Ten" high-level requirements and skilful analysis – Evolutionary and adaptive planning – Incremental delivery – Evolutionary delivery – The most common mistake – Specific iterative and Evolutionary methods.

UNIT II-AGILE AND ITS SIGNIFICANCE

Agile development – Classification of methods – The agile manifesto and principles – Agile project management – Embrace communication and feedback – Simple practices and project tools – Empirical Vs defined and prescriptive process – Principle-based versus Rule-Based – Sustainable discipline: The human touch – Team as a complex adaptive system – Agile hype – Specific agile methods. The facts of change on software projects – Key motivations for iterative development – Meeting the requirements challenge iteratively – Problems with the waterfall. Research evidence – Early historical project evidence – Standards-Body evidence – Expert and thought leader evidence – A Business case for iterative development – The historical accident of waterfall validity.

UNIT III-AGILE METHODOLOGY

Method overview – Lifecycle – Work products, Roles and Practices values – Common mistakes and misunderstandings – Sample projects – Process mixtures – Adoption strategies – Fact versus fantasy – Strengths versus "Other" history.

UNIT IV-AGILE PRACTICING AND TESTING

Project management – Environment – Requirements – Test – The agile alliances – The manifesto – Supporting the values – Agile testing – Nine principles and six concrete practices for testing on agile teams.

REFERENCES

1. Elisabeth Hendrickson, "*Agile Testing*" Quality Tree Software Inc 2008.
2. Craig Larman "*Agile and Iterative Development – A Manager's Guide*"
Pearson Education – 2004.
3. Alistair "*Agile Software Development series*" Cockburn - 2001.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-109	Software Process Maturity Model							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	To know about the software process and Software Process Maturity Models							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To study about various Software process maturity models							
CO2	To study about how to assess software process							
CO3	To know about the key process areas of the software process							
CO4	To study about software improvement sequences							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I - INTRODUCTION

Software Process - Software Maturity Framework – Software process Improvement – Process Maturity levels – Principles of Software process Change – Software Process Assessment

UNIT II - CMM

CMM Introduction – CMM Maturity Levels - Initial process- Repeatable Process – Defined Process – Managed Process – Optimizing Process.

UNIT III - TMM

Introduction to TMM – Structure of the TMM – Components of TMMi – Generic Goals and Generic Practices – Process areas for Generic practices – TMMi Maturity Levels – Initial – Managed – Defined – Management and Measurement – Optimization.

UNIT IV - AGILE MATURITY MODEL

Agile Software Development – Process Improvement framework for Agile Software Development – Intial Level – Explored Level – Defined level – Improved Level – Sustained Level - Software Process Improvement for Agile Software Development Practices.

REFERENCES

1. Watts S. Humphrey "*Managing the Software Process*", Pearson Education, 2008
2. Marry Beth Chrissis, Mike Konnard and Sandy Shrum, "*CMMI : guidelines for Process Integration and Product Improvement*", Addison Wesley, 3rd Edition,2011.
3. Mark. C. Paulk, "*CMM: Guidelines for Improving the Software Process*" Addison-Wesley, 2011.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-111	Software Reliability							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	This course will look at professional techniques for understanding, assessing and applying the software reliability models in software development systems.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To appreciate and understand scientific concepts of Software and Hardware Reliability.							
CO2	To apply Software Reliability Growth Models in Software Development							
CO3	To emphasize the Application of Software Reliability Models							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-SOFTWARE RELIABILITY MODELS

Introduction - Historical Perspective and Implementation, classification, limitations and issues, Exponential Failure Models – Jelinski-moranda model, Poisson, Musa, Exponential models, Weibull Model, Musa-okumoto Model, Bayseian Model – Littlewood verral Model, Phase Based Model

UNIT II-PREDICTION ANALYSIS

Model Disagreement and Inaccuracy – Short & Long Term Prediction, Model Accuracy, Analyzing Predictive Accuracy – Outcomes, PLR, U & Y Plot, Errors and Inaccuracy, Recalibration – Detecting Bias, Techniques, Power of Recalibration, Limitations in Present Techniques, Improvements.

UNIT III-THE OPERATIONAL PROFILE

Concepts and Development Procedures – Customer Type, User Type, System Mode, Functional and Operational Profile, Test Selection - Selecting Operations, Regression Test, Special Issues – Indirect Input Variables, Updating, Distributed system.

UNIT IV-TESTING FOR RELIABILITY MEASUREMENT

Software Testing – Types, White and Black Box, Operational Profiles – Difficulties, Estimating Reliability, Time/Structure based software reliability – Assumptions, Testing methods, Limits, Starvation , Coverage, Filtering, Microscopic Model of Software Risk.

REFERENCES

1. Patric D. T.O connor, "*Practical Reliability Engineering*", 4th Edition, John Wesley & sons, 2003.
2. John D. Musa, "*Software Reliability Engineering",* Tata McGraw Hill, 1999.
3. Michael Lyu, "*Handbook of Software Reliability Engineering"*, IEEE Computer Society Press, ISBN: 0-07-039400- 8, 1996.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-113	Software Agents							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	This course provides a thorough understanding of agent related system development							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To understand Agent development							
CO2	Gain Knowledge in Multi agent and Intelligent agents							
CO3	To Understand Agents and security							
CO4	Gain Knowledge in Agent Applications							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-INTRODUCTION

The agent landscape – The smart agent framework: Introduction – Initial concepts – Entities-Objects – Agents – Autonomy – Tropistic agent – Specification structure of SMART. – Agent relationships – An operational analysis of Agent relationships.

UNIT II-SOCIOLOGICAL AGENTS

Sociological Agents - Autonomous Interaction - Contract Net as a global directed system – Computational Architecture for BDI agents – Evaluating social dependence networks – Normative agents.

UNIT III-INTELLIGENT AUTONOMOUS AGENTS AND COMMUNICATION

Intelligent Agents –Deductive Reasoning Agents – Practical reasoning agents - Reactive agents – Hybrid Agents – Understanding Each other – Communicating – Methodologies

UNIT IV-APPLICATIONS OF AGENTS

Multi Agent system: Theory approaches and NASA applications – Agent based control for multi-UAV information collection- Agent based decision support system for Glider pilots – Multi agent system in E- Health Territorial Emergencies – Software Agents for computer network security- Multi-Agent Systems, Ontologies and Negotiation for Dynamic Service Composition in Multi- Organizational Environmental Management.

REFERENCES

1. Mohammad Essaaidi, Maria Ganzha, and Marcin Paprzycki, *"Software Agents, Agent Systems and Their Applications*", IOS Press, 2012.
2. Mark d Inverno and Michael Luck, *"Understanding Agent Systems",* Springer,2010.
3. Michael Wooldridge, *"An Introduction to Multi Agent Systems*", John Wiley & Sons Ltd., 2009.
4. Lin Padgham, Michael Winikoff, *"Developing Intelligent Agent Systems: A Practical Guide"*, John Wiley & Sons Ltd., 2004.
5. Bradshaw , *"Software Agents*" , MIT Press, 1997.
6. Richard Murch, Tony Johnson, *"Intelligent Software Agents"*, Prentice Hall, 2000.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-115	Human Interface System Design							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	This course on user Interface Design provides a basic understanding of interface design and principles.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	Students learn about the design process management							
CO2	To understand about Interaction devices and windows strategies							
CO3	To understand about how to Manage Virtual Environments							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-INTRODUCTION

Goals of System Engineering – Goals of User Interface Design – Motivations of Human factors in Design – High Level Theories –Object-Action Interface Design - Three Principles – Guidelines for Data Display and Data Entry

UNIT II-MANAGING DESIGN PROCESS

Introduction- Organizational Design to Support Usability – The Three Pillars of Design- Development Methodologies- Ethnographic Observation – Participating Design- Scenario Development- Social Impact Statement for Early Design – Legal Issues- Reviews – Usability Testing and laboratories- Surveys- Acceptance tests – Evaluation during Active use- Specification Methods- Interface – Building Tools- Evaluation and Critiquing tools

UNIT III-MANIPULATION AND VIRTUAL ENVIRONMENTS

Introduction-Examples of Direct Manipulation Systems –Explanation of Direct Manipulation- Visual Thinking and Icons – Direct manipulation Programming – Home Automation- Remote Direct manipulation- Virtual Environments- Task

UNIT IV-WINDOWS STRATEGIES AND INFORMATION SEARCH

Introduction- Individual Widow Design- Multiple Window Design- Coordination by Tightly – Coupled Widow- Image Browsing- Personal Role Management and Elastic Windows – Goals of Cooperation – Asynchronous Interaction – Synchronous Distributed – Face to Face- Applying Computer Supported Cooperative Work to Education – Database query and phrase search in Textual documents – Multimedia Documents Searches – Information Visualization – Advance Filtering Hypertext and Hypermedia – World Wide Web- Genres and Goals and Designers – Users and their tasks – Object Action Interface Model for Web site Design

REFERENCE

1. Alan Dix et al, " *Human - Computer Interaction ",* Pearson , 2010.
2. Ben Shneiderman , "*Designing the User Interface*", 4th Edition, Pearson, 2010.
3. Dr. Jonathan Lazar, Dr. Jinjuan Heidi Feng, Dr. Harry Hochheiser, "*Research Methods in Human Computer Interaction*" –John Wiley 2010.
4. Wilbert O. Galiz , "*The Essential guide to User Interface Design*", Wiley Dreamtech, 2009.
5. Jef Raskin , "*The Human Interface* ", Addison – Wesley – 2008.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTRM-111	Research Methodology and IPR						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Practical**	**Total**
2	**0**	**0**	**2**	**60**	**40**	**-**	**100**
Program Objective (PO)	To enable students to Research Methodology and IPR for further research work and investment in R & D which leads to creation of new and better products, and in turn brings about, economic growth and social benefits.						
Course Outcomes (CO)							
CO1	Understand research problem formulation.						
CO2	Analyze research related information						
CO3	Understand that today's world is controlled by Computer, Information Technology, but tomorrow world will be ruled by ideas, concept, and creativity.						
CO4	Understanding that when IPR would take such important place in growth of individuals & nation, it is needless to emphasise the need of information about Intellectual Property Right to be promoted among students in general & engineering in particular.						

Unit 1:

Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem.

Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations

Unit 2:

Effective literature studies approaches, analysis, Plagiarism, Research ethics, Effective technical writing, how to write report, Paper.

Developing a Research Proposal, Format of research proposal, a presentation and assessment by a review committee.

Unit 3:

Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of Patenting and Development: technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property. Procedure for grants of patents, Patenting under PCT.

Unit 4:

Patent Rights: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases. Geographical Indications.

New Developments in IPR: Administration of Patent System. New developments in IPR; IPR of Biological Systems, Computer Software etc. Traditional knowledge Case Studies, IPR and IITs.

References:

1. Stuart Melville and Wayne Goddard, "Research methodology: an introduction for science & engineering students'"
2. Wayne Goddard and Stuart Melville, "Research Methodology: An Introduction"
3. Ranjit Kumar, 2 ndEdition , "Research Methodology: A Step by Step Guide for beginners"
4. Halbert, "Resisting Intellectual Property", Taylor & Francis Ltd ,2007.
5. Mayall , "Industrial Design", McGraw Hill, 1992.
6. Niebel , "Product Design", McGraw Hill, 1974.
7. Asimov , "Introduction to Design", Prentice Hall, 1962.
8. Robert P. Merges, Peter S. Menell, Mark A. Lemley, " Intellectual Property in New Technological Age", 2016.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-117	Software Engineering Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Practical**	**Minor Test**	**Total**	**Time**
0	**0**	**4**	**2**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	**This Software Laboratory focuses on the software engineering methodologies for project development and to gain knowledge about open source tools for Computer Aided Software Engineering.**						
Course Outcomes (CO)							
CO1	To develop test cases for any problem						
CO2	Use open source case tools to develop software.						
CO3	Analyze and design software requirements in efficient manner.						

List of Practical

SOFTWARE REQUIRED:

Open source Tools: StarUML / UMLGraph / Topcased/ Argo UML

Prepare the following documents for each experiment and develop the software using software engineering methodology.

1. **Problem Analysis and Project Planning -**Thorough study of the problem –Identify Project scope, Objectives and Infrastructure.
2. **Software Requirement Analysis -** Describe the individual Phases/modules of the project and Identify deliverables.
3. **Data Modeling -** Use work products – data dictionary, use case diagrams and activity diagrams, build and test class diagrams, sequence diagrams and add interface to class diagrams.
4. **Software Development and Debugging** – implement the design by coding
5. **Software Testing** - Prepare test plan, perform validation testing, coverage analysis, memory leaks, develop test case hierarchy, Site check and site monitor.

Case Studies:

Academic domain

1. Course Registration System
2. Student marks analysing system

Railway domain

3. Online ticket reservation system
4. Platform assignment system for the trains in a railway station

Medicine domain

5. Expert system to prescribe the medicines for the given symptoms
6. Remote computer monitoring

Finance domain

7. ATM system
8. Stock maintenance

Human Resource management

9. Quiz System
10. E-mail Client system.

MTSE-119	Agile Software Engineering Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Practical**	**Minor Test**	**Total**	**Time**
0	**0**	**4**	**2**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	**This Software Laboratory focuses on to analyze, design and provide optimal solution for Computer Science & Engineering and multidisciplinary problems.**						
Course Outcomes (CO)							
CO1	To Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.						
CO2	To Design solutions for complex engineering problems						
CO3	To Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools						
CO4	To demonstrate the knowledge of and need for sustainable development. .						

List of practical

1. Understand the background and driving forces for taking an Agile Approach to Software Development. Study the Important Characteristics that make agile approach best suited for Software Development.

2. Understand the business value of adopting agile approach.

3. Study the Agile Process Examples
   a) SCRUM
   b) FDD
   c) Lean software development
   d) XP

3. Understand agile development practices using SCRUM

4. Drive Development with Unit Test using Test Driven Development.

5. Apply Design principle and Refactoring to achieve agility

6. To study automated build tool.

7. To study version control tool.

8. To study Continuous Integration tool.

9. Perform Testing activities within an agile project.

MTSE-102	Software Risk Management							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	The goal of this course is to engage students in active discovery of risk management principles and the process of designing and implementing a risk management program.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To understands fundamentals of Risk Management Process.							
CO2	To learn Risk Management Infrastructure process.							
CO3	To learn applications of Risk Management.							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT-1

Introduction to Software Risk Management: P212 Success Formula: Major Factors in Risk Management Capability, People, Process, Infrastructure, Implementation, Risk Management Roadmap.

UNIT-2

Risk Management Process: Identity Risk, Analyze Risk, Plan Risk, Resolve Risk.

UNIT-3

Risk Management Infrastructure: Develop policy, Define standard process, Train Risk Technology, Verify Compliance, Improve Practice.

UNIT-4

Risk Management Implementation: Establish Initiative, Develop Plan, Tailor Standard Process, Assess Risk, Control Risk. People in Crisis and Control Problem, Mitigation, Prevention, Anticipation, Opportuninty.

Reference Books:

1. Elaine M. Hall, Managing Risk: Methods for Software Systems Development, The SEI Series in Software Engineering, Addison –Welsey, Masschachusetts.
2. Down. Alex, Michael Coleman. And Peter Absolon. Risk Management For Software Projects, McGraw-Hill, New York.
3. Charette. Robert N, Application Strategies for Risk Analysis, McGraw Hill, New York.
4. Grey. Stephen, Practical Risk Assessment for Project Management. Chichester, John Wiley & Sons. New York.
5. Glendon. A and Alan Waring, Managing Risk. International Thomson Business & COMPUTER Press, New York.
6. Jones.Capres. Assessment and Control of Software, Prentice Hll Press, New Jersey.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-104	Social Networks					
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**
3	**0**	**0**	**3**	**60**	**40**	**100**
Program Objective (PO)	This emerging and innovative field will provide the insight into latest communication techniques used in the online social networks for identifying and representing the hidden relationships, tracking the flow of information and to recognize data patterns in social networks by using graph, matrix, relationships, clustering, and equivalence between users.					
Course Outcomes (CO)						
CO1	To understand the essentials of social networks by learning different types of entities and relationships as nodes, edges within the graph and represent these information as relational data to determine the relative importance of a vertex to find the design levels					
CO2	To explore the detailed explanation of data generalization and mining from Twitter, Facebook and LinkedIn in well informed and efficient manner.					
CO3	To describe the semantic web using mining associations, correlations, classification, betweenness, centrality, equivalence relation, centralization, clustering coefficient and structural cohesion to generate visualizations and perform empirical investigations of network data.					
CO4	To interpret and synthesize the results with respect to collated datasets by using structural equivalence, automorphic equivalence and regular equivalence for interpreting quality factors and mining of complex type of data to execute better recommendation.					

Unit: I: Social Networks and Related Concepts

Introduction to Social Networks: Introduction, uses, examples and types of social networks, Social and economic networks, Opportunities and challenges in social networks, Social structure in social networks, Properties of social networks, algorithmic and economic aspects of social networks

Social Network Data: Nodes, Edges, Relationship, Graphs, Samples and Boundaries, Formal methods, Adjacency Matrix for undirected and directed networked graphs and using matrices to represent social relations, Random graphs, Properties of random graphs, Percolations, Branching processes, Growing spanning tree in random graphs.

Level in Social Networks: Ego networks, partial networks, complete or global networks, social networks methods including binary or valued, directed or undirected.

Unit: II Mining the Social Web

Mining Twitter: Fundamental Twitter Terminology, creating a Twitter API Connection, Exploring Trending Topics, searching for Tweets, extracting Tweets entities, analyzing Tweets and Tweet entities with frequency analysis, computing the lexical diversity of Tweets, Examining patterns in Retweets, Visualizing frequency data with histograms.

Mining Facebook: Understanding the social graph API, Understanding the open graph protocol, Analyzing social graph connections

Mining LinkedIn: Making LinkedIn API requests, Downloading LinkedIn connections as a CSV file, Clustering, normalizing data for analysis, measuring similarity, and clustering algorithms.

Unit: III Mining Web pages and Semantic Web

Mining Web pages: Scraping, Parsing and Crawling the Web, Discovering semantics by decoding syntax, Entity-Centric analysis: A paradigm shift, Quality of analytics for processing human language data.

Mining the Semantically Marked-Up Web: Microformats: Easy-to-implement Metadata, Semantics markup to semantic Web: A brief interlude, The semantic Web: An evolutionary revolution.

Social Network Analysis: Introduction, History, Metrics in social network analysis (Betweenness, Centrality, Equivalence relation, Centralization, Clustering coefficient and Structural cohesion).

Unit IV: Equivalence in Social Networks

Structural equivalence, Automorphic equivalence and Regular equivalence

Text Books:

1. Matthew A. Russell, “Mining the Social Web”, O'Reilly and SPD, Second edition New Delhi, 2013.
2. Hanneman, R. A., & Riddle, M., “Introduction to social network methods, Riverside, California: University of California, Riverside. Available at: http://faculty.ucr.edu/~hanneman/nettext/.
3. “Social network analysis: Theory and applications”. A free, Wiki Book available at: http://train.ed.psu.edu/WFED-543/SocNet_TheoryApp.pdf.

Reference Books:

1. Lon Safko, “The Social Media Bible: Tactics, Tools, and Strategies for Business Success”, Wiley 3rd Ed., 2012.
2. Peter K Ryan, “Social Networking”, Rosen Publishing Group, 2011.
3. John Scott, Peter J. Carrington, “Social Network Analysis”, SAGE Publishing Ltd., 2011.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-106	Cloud Computing							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	To provide a comprehensive introduction to cloud computing and about cloud Services							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To understand Cloud Computing basics and its models.							
CO2	To learn the fundamentals of Data Centers.							
CO3	To understand the Architecture of Data Centers and Design Principles							
CO4	To understand the Security aspects and security framework.							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-INTRODUCTION

Cloud Computing Introduction, From, Collaboration to cloud, Working of cloud computing, pros and cons, benefits, developing cloud computing services, Cloud service development, discovering cloud services.

UNIT II-CLOUD COMPUTING FOR EVERYONE

Centralizing email communications, cloud computing for community, collaborating on schedules, collaborating on group projects and events, cloud computing for corporation, mapping schedules managing projects, presenting on road.

UNIT III-USING CLOUD SERVICES

Collaborating on calendars, Schedules and task management, exploring on line scheduling and planning, collaborating on event management, collaborating on contact management, collaborating on project management, collaborating on word processing, spreadsheets, and databases.

UNIT IV-OUTSIDE THE CLOUD

Evaluating web mail services, Evaluating instant messaging, Evaluating web conference tools, creating groups on social networks, Evaluating on line groupware, collaborating via blogs and wikis

Storing and Sharing: Understanding cloud storage, evaluating on line file storage, exploring on line book marking services, exploring on line photo editing applications, exploring photo sharing communities, controlling it with web based desktops.

REFERENCES

1. Michael Miller, "*Cloud Computing*", Pearson Education, New Delhi, 2009.
2. Toby Velte, Anthony Velte, Robert Elsenpeter, "*Cloud Computing: A Practical Approach"*, McGraw Hill, 2009.
3. Mauricio Arregoces,Maurizio Portolani, "*Data Center Fundamentals*", Cisco Press,2004.
4. Scott Lowe, Jason W, Mc. Carty and Mathew K. Johnson, "*VMware, Vsphere 4 Administration, Instant Reference"*, Published by Sybex, 2009.
5. George Reese, "*Cloud Application Architectures Building Applications and Infrastructure in the Cloud*", O'Reilly Media, 2009.
6. Grantt Sauls "*Introduction to Data Centers*", Certified Data Centers Specialist, Tutorial.
7. Brendan O'Brien, Alberto Rodriguez, Stephen Sutherland and Mark Wheatley, "*Server Virtualization Software"*, Tutorial, 2009.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-108	Software Testing & Quality Assurance							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	The purpose of this course is to presents the knowledge about Testing background such introduction of Bug , cause of Bug, how it effect on cost of project, role of STLC cycle realities of software testing. This subject also gives the knowledge software testing fundamentals, under the study of types of testing this subject enlighten the Configuration testing, Compatibility testing, Foreign language testing, Usability testing, Testing the documentation, Testing for software security, Web site testing and more. At the end this subject focuses on the test planning and quality assurance.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To discuss software testing background.							
CO2	To introduce software testing techniques.							
CO3	To explain different types of testing to understand realistic problem.							
CO4	To create awareness about the process part as per as software testing is concern.							

UNIT I-INTRODUCTION TO SOFTWARE TESTING

Introduction – s/w testing background - What is a bug? Why do bugs occur? The cost of bugs. Goals of a software tester. Characteristics of s/w tester. Software development process- product component, software project staff, software development lifecycle model. The realities of s/w testing – testing axioms, s/w testing terms and definitions, Software Testing Life Cycle(STLC).

UNIT II- S/W TESTING FUNDAMENTALS

S/w testing fundamentals- Examining the specifications - Black box and white box testing, Static and dynamic testing, Static black box testing, Performing a high level review of the specification, low level specification test techniques. Testing the s/w with blinders on – Dynamic black box testing, Test to pass and test to fail, Equivalence partitioning, data testing, State testing, Other black box test techniques. Examining the code – Static white box testing, Formal review, Coding standards and guidelines, Generic code review checklist. Testing the software with X-ray glasses – Dynamic white box testing, Dynamic white box testing, verses debugging testing the pieces

UNIT III TYPES OF TESTING

Configuration testing, Compatibility testing, Foreign language testing, Usability testing, Testing the documentation, Testing for software security.Web site testing, Automated testing and test tools- Benefits of automation and tools, various test tools, Software test automation, Random testing. Bug bashes and beta testing – Having other people test your s/w, Test sharing, Beta testing, Outsourcing your testing.

Performance Testing – Introduction, Benefits of performance testing. Types of performance testing Tools for performance Testing, Process for performance testing, challenges.

UNIT IV-TEST PLANNING AND QUALITY ASSURANCE

Planning the test – Goal of test planning, Various test planning topics, Writing and tracking test cases- Goal of test case planning, Test case planning overview, Test case organization and tracking, Reporting what you find - Getting the bug fixed, Isolating and replacing bugs, Bug's lifecycle, Bug tracking system, Measuring the success, Software quality assurance- Quality is free, Testing and quality assurance in the work place, Test management and organizational structures, capability maturity model (CMM), ISO 9000 Test Metrics and Measurement – Test Defect Metrics.

TEXT BOOKS:

1. Ron Patton, "Software Testing" SAMS Publishing
2. Marnei L. Hutcheson – "Software Testing Fundamentals: Methods and Metrics" WILEY Pub.

REFERENCE BOOKS:

1. Pressman "Software Engineering" McGraw-Hill publications.
2. Strinivasan Desikan and Gopal swami Ramesh,Software Testing – Principles and Practices, Pearsons.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-110	Data Warehousing and Data Mining							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	This course enables to understand the concepts of Data Warehousing and Data Mining.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To learn the fundamentals of designing a large-scale data warehouse using relational technologies							
CO2	To understand the Data Warehouse and OLAP Technology in Data Mining							
CO3	To study the Mining Association Rules in Large Databases, Classification							
CO4	To know Cluster Analysis and its Application Trends in Data Mining.							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-DATA WAREHOUSING AND BUSINESS ANALYSIS

Data Warehousing and Business Analysis: - Data warehousing Components – Building a Data warehouse – Mapping the Data Warehouse to a Multiprocessor Architecture – DBMS Schemas for Decision Support – Data Extraction, Cleanup, and Transformation Tools –Metadata – reporting – Query tools and Applications – Online Analytical Processing (OLAP) – OLAP and Multidimensional Data Analysis.

UNIT II-DATA MINING

Data Mining: - Data Mining Functionalities – Data Preprocessing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation. Association Rule Mining: - Efficient and Scalable Frequent Item set Mining Methods – Mining Various Kinds of Association Rules – Association Mining to Correlation Analysis – Constraint-Based Association Mining.

UNIT III-CLASSIFICATION AND PREDICTION

Classification and Prediction: - Issues Regarding Classification and Prediction – Classification by Decision Tree Introduction – Bayesian Classification – Rule Based Classification – Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Ensemble Methods – Model Section.

UNIT IV-APPLICATIONS OF DATA MINING

Mining Object, Spatial, Multimedia, Text and Web Data: Multidimensional Analysis and Descriptive Mining of Complex Data Objects – Spatial Data Mining – Multimedia Data Mining – Text Mining – Mining the World Wide Web.

REFERENCES

1. Jiawei Han and Micheline Kamber "*Data Mining Concepts and Techniques"* Second Edition, Elsevier, Reprinted 2008.
2. Sam Anahory & Dennis Murray, "*Data Warehousing in the real world*", Pearson Education Ltd, 2011.
3. Alex Berson and Stephen J. Smith "*Data Warehousing, Data Mining & OLAP*", Tata McGraw – Hill Edition, Tenth Reprint 2007.
4. K.P. Soman, Shyam Diwakar and V. Ajay "*Insight into Data mining Theory and Practice"*, Easter Economy Edition, Prentice Hall of India, 2006.
5. Gupta G. K. "*Introduction to Data Mining with Case Studies*", Easter Economy Edition, Prentice Hall of India, 2006.
6. Pang-Ning Tan, Michael Steinbach and Vipin Kumar *"Introduction to Data Mining"*, Pearson Education, 2007.
7. Jiawei Han & Micheline Kamber *"Data Mining Concepts and Techniques"* , Morgan Kaufmann Publishers, Elsevier,2nd Edition, 2006.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-112	Object Oriented Programming							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	The course provide insight knowledge about programming language (C++ and JAVA)							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To learn the fundamentals of Object Oriented Programming							
CO2	To understand the concepts of Classes & Objects in C++ and Java							
CO3	To understand the concept of static and dynamic polymorphism in C++and Java.							
CO4	To understand the concept of streams in C++ and Java.							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-INTRODUCTION TO OOP

Overview of C++ - classes - structures - union - friend function - friend class -inline function - constructors – static members - scope resolution operator - passing objects to functions - function returning objects -Arrays - pointers - this pointer - references - dynamic memory allocation

UNIT II-OVERLOADING & INHERITANCE

Function overloading - default arguments - overloading constructors - pointers to functions Operator overloading - member operator function - friend operator function - type conversion - inheritance - types of inheritance - virtual base class - polymorphism - virtual function.

UNIT III-TEMPLATES & EXCEPTION

Class templates and generic classes - function templates and generic functions -- exception handling - derived class exception - exception handling functions - Streams - formatted I/O with its class functions and manipulators - creating own manipulators - file I/O - conversion functions- standard template library.

UNIT IV-INTRODUCTION FOR JAVA

JAVA Basics: Importance and features of java- Modifiers- Access Controls-Data types- Expressions-Declarations-Statements- classes and objects and Control Structures-Program Structures-String handling-Packages-Interfaces-Working with java.util Package- Garbage Collection-Object Class - Exception Handling, I/O and JDBC: Exception Handling: Fundamentals exception types- uncaught exceptions throw- throw final- built in exception- creating your own exceptions.

REFERENCES

1. Balagurusamy E, "*Object Oriented Programming with C++",* 4/E, TMG, 2011.
2. Hubbard, *"Programming with C++",* 3/e, Schaum Outline Series, TMH, 2010.
3. Thomas Wu- "*An Introduction to Object Oriented Programming with Java – Special"* Indian Edition 5th 2010.
4. Balagurusamy E, "*Programming with Java: A Primer"*, 4th Edition, Tata Mcgraw Hill, 2009.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-114	Pattern Oriented Software Architecture							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	The course gives an insight of the most commonly used software architecture and design patterns and their applications							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	The students get basic knowledge of patterns and description of patterns							
CO2	To understand basic architectural patterns.							
CO3	To get an insight on the design patterns and mining.							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-INTRODUCTION TO SOFTWARE ARCHITECTURE

Introduction – Software architecture – An engineering discipline for software - Architectural Styles – Pipes and filters – Layered Systems - Black board – Repositories - Process control - Distributed system – Interactive system – Adaptive system

UNIT II-DESIGN PATTERNS & PATTERN SYSTEM

Introduction to patterns – Pattern category – Relationship between patterns –Pattern Description – Patterns software architecture -Structural decomposition Organization of work – Access control – Management and Communication –Idioms, Pattern system – Pattern Classification – Pattern Selection –implementation – Evolution – Patterns in Software architecture – Non –functionalproperties – Techniques of Software architecture.

UNIT III-COMMUNITY, MINING, CONCURRENT & NETWORKED

Roots – Community – Pattern Mining - Organizing and Indexing – Methods andtools – Algorithm – Data Structures and Patterns – Formalizing Patterns,Concurrent and Networked Objects, Service Access and Configuration Patterns

UNIT IV-EVENT HANDLING & SYNCHRONIZATION PATTERNS

Event Handling Patterns – Reactor, Proactor, Asyn Completion Tokens, Acceptor- Connector, Synchronization Patterns – Locking – Scoped, Strategized, Thread - safe Interface, Double-Checked Locking Optimization.

REFERENCES

1. Frank Buschmann, Kelvin Henney & Douglas Schimdt, "*Pattern-Oriented Software Architecture - A System of Patterns",* Volume 1, Wiley,2007.
2. Frank Buschmann, Kelvin Henney & Douglas Schimdt, "*Pattern-Oriented Software Architecture – Pattern for Concurrent and Networked Objects"*,Volume 2 ,Wiley,2000.
3. Mary Shaw , David Garlan , "*Software architecture perspectives on a Emerging Dicipline*",EEE,PH1,1996.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-116	Software Measurement and Metrics							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	The purpose of this course is to provide the knowledge about Software Metrics, Essentials of software metrics and practical knowledge to assess software.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To provide a solid background knowledge about software Metrics.							
CO2	To educate various metrics and models to assess software.							
CO3	To provide hands on experience to use and implement metrics.							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-THE HISTORY AND EVOLUTION OF SOFTWARE METRICS

Evolution of the software industry and evolution of software measurements – The cost of counting function point metrics – The paradox of reversed productivity for high-Level languages- The Varieties of functional metrics – Variations in application size and productivity rates – Future Technical Developments in Functional Metrics- Software measures and metrics not based on function points.

UNIT II-MEASURING SOFTWARE QUALITY

Quality control and international competition – Defining quality for measurement and estimation – Five steps to software quality control- Measuring software defect removal- Measuring Defect removal efficiency – Measuring the costs of defect removal – Evaluating defect prevention methods – Measuring customer reported defects- Measuring invalid defects, Duplicate defects and special cases-Reliability Models - The Rayleigh Model- Reliability Growth Models.

UNIT III-PROCESS METRICS

In-Process Metrics for Software Testing - Test Progress S Curve - Testing Defect Arrivals Over Time - Product Size Over Time - CPU Utilization - Effort/Outcome Model. Complexity Metrics and Models - Lines of Code - Halstead's Software Science - Cyclomatic Complexity. - Syntactic Constructs - Structure Metrics.
Metrics for Object-Oriented Projects - Concepts and Constructs - Design and Complexity Metrics - Lorenz Metrics and Rules of Thumb - CK OO Metrics Suite - Productivity Metrics.

UNIT IV-MECHANICS OF MEASUREMENT

Software Assessments – Software Baselines – Software Benchmarks- What a Baseline analysis covers – Developing or Acquiring a baseline data collection Instrument – Administering the data collection questionnaire – Analysis and aggregation of the Baseline data. Measuring and Analyzing Customer Satisfaction - Surveys - Data Collection - Sampling Methods - Analyzing Satisfaction Data. Conducting In-Process Quality Assessments - Preparation - Evaluation - Quantitative Data - Qualitative Data - Evaluation Criteria - Overall Assessment.

REFERENCES

1. Caper Jones, *"Applied Software Measurement: Global Analysis of Productivity and Quality"* , Third Edition, McGraw Hill Companies, 2008.
2. Stephen H. Kan, *"Metrics and Models in Software Quality Engineering*", Addison Wesley, 2011.
3. Mark Lorenz, Jeff Kidd, *"Object-Oriented Software Metrics*", Prentice Hall, 2000.
4. Naresh Chauhan, *"Software Testing Principles and Practices*", Oxford University Press, 2010.
5. Ravindranath Pandian C., *"Software Metrics A Guide to planning, Analysis, and Application*", Auerbach, First Indian Reprint, 2011.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-118	Software Quality Models & Testing Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Practical**	**Minor Test**	**Total**	**Time**
0	**0**	**4**	**2**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	**This Software Laboratory focuses on test case generation on testing different kinds of software and to provide the in-depth coverage of software quality models and software testing strategies.**						
Course Outcomes (CO)							
CO1	To develop test cases for any problem						
CO2	To pursue testing on any level of software design by using different testing strategies						
CO3	Create a test plan document of real time applications.						
CO4	To apply testing tools for designing the test case to test the real time application.						

Case Study 1: Write the test cases for the largest of three number based on:

- Boundary value analysis test
- Robustness based testing
- Equivalence class partitioning test
- Decision table based test

Case Study 2: Cause Effect Graph Testing for a Triangle Program-Perform cause effect graph testing to find a set of test cases for the following program specification: Write a program that takes three positive integers as input and determine if they represent three sides of a triangle, and if they do, indicate what type of triangle it is. To be more specific, it should read three integers and set a flag as follows:
If they represent a scalene triangle, set it to 1.
If they represent an isosceles triangle, set it to 2.
If they represent an equilateral triangle, set it to 3.
If they do not represent a triangle, set it to 4.

Case Study 3: Boundary Value Analysis for a Software Unit-The following is a specification for a software unit. The unit computes the average of 25 floating point numbers that lie on or between bounding values which are positive values from 1.0 (lowest allowed boundary value) to 5000.0 (highest allowed boundary value). The bounding values and the numbers to average are inputs to the unit. The upper bound must be greater than the lower bound. If an invalid set of values is input for the boundaries an error message appears and the user is reported. If the boundary values are valid the unit computes the sum and the average of the numbers on and within the bounds. The average and sum are output by the unit, as well as the total number of inputs that lie within the boundaries. Derive a set of equivalence classes for the averaging unit using the specification, and complement the classes using boundary value analysis. Be sure to identify valid and invalid classes. Design a set of test cases for the unit using your equivalence classes and boundary values. For each test case, specify the equivalence classes covered, input values, expected outputs, and test case identifier. Show in tabular form that you have covered all the classes and boundaries. Implement this module in the programming language of your choice. Run the module with your test cases and record the actual outputs. Save an uncorrected version of the program for future use.

Case Study 4: Write the test cases for any known application (e.g. banking application) using
I) Basis path testing
II) Component testing
III) Data flow analysis test

Case Study 5: Create a test plan document for any application (e.g. Library Management System)

Case Study 6: Model Based Testing-Design and develop a scientific calculator program using various GUI components and events. Build the test model for the same. Determine the inputs that can be given to the model.
Calculate expected output for the model. Run the test cases. Compare the actual output with the expected output. Any model-based technique can be used for building the test model.

Case Study 7: Study and implementation of

- Mutation test
- Slice based test

Case Study 8: Introduction to any two open source testing tool:

- Study of any testing tool (e.g. Win runner)
- Study of any web testing tool (e.g. Selenium)
- Study of any bug tracking tool (e.g. Bugzilla, bugbit)
- Study of any test management tool (e.g. Test Director)
- Study of any open source-testing tool (e.g. Test Link)

Case Study 9: Web Application Testing for Student Grade System-With educational organizations under increasing pressure to improve their performance to secure funding for future provision of programmes, it is vital that they have accurate, up-to-date information. For this reason, they have MIS systems to record and track student enrolment and results on completion of a learning programme. In this way they can monitor achievement statistics. All student assignment work is marked and recorded by individual module tutors using a spreadsheet, or similar, of their own design. In the computing department these results are input into a master spreadsheet to track a student's overall progress throughout their programme of study. This is then made available to students through the web portal used in college. Perform web application testing for this scenario.

MTSE-120	Social Networks Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Practical**	**Minor Test**	**Total**	**Time**
0	**0**	**4**	**2**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	This Software Laboratory focuses on accessing the dataset from social networks and then applying machine learning techniques, data cleaning and visualization of data in real time environments using Python programming and NLTK						
Course Outcomes (CO)							
CO1	To access the data from social networks						
CO2	To deign machine learning modules for efficient system						
CO3	Create the algorithms for accessing Social Media and data cleaning						
CO4	To apply testing tools for visualization of data in real time application.						

List of practical

1. Write a python program to remove an item from tuple and merge three dictionaries.
2. Write a python program to construct pyramids of stars (*) and numbers using nested for loop.
3. Write a python function to check whether a number is perfect or not and use filter function to print vowels from a given list.
4. Write a python program to estimate coefficients of an equation using linear regression model.
5. Write a python program to predict gender of a person if heigth, weight and shoe size are given using any four supervised learning algorithms.
6. Write a python program to find noun, verb and adjective in a given sentence.
7. Write a python program to calculate frequency of each word in a file after removing stopwords from it.
8. Write a program to for analyzing the behaviour (i.e. check whether a tweet is of positive, negative, or compound nature) of tweets and plot the results.
9. Write a program to sort the list of numbers using shell sort.
10. Write a python program to predict gender of a person from his/her name.
11. Write a python program to make a prediction about a movie from its review.
12. Write a program to plot the image in PNG format using matplotlib for average, max, and min of the data taken from a CSV file.
13. Write a program for classifying the text using NLTK.
14. Write a python program to guess behavior of a person.
15. Write a python program to print trending and common trends tweets in world,us and india.
16. Write a python program to use hashtag as basis of search query to fetch some tweets for further analysis.
17. Write a python program extract twitter entities such as hashtags, screen names.
18. Write a python program to clean any given dataset.
19 Write a python program to visualize a data using histogram, boxplot and scatter plot matrix.
20. Write a program for sentiment analysis of tweets (i.e. polarity and subjectivity).

MTSE-201	Software Quality Management							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	This course covers the principles of software development emphasizing processes and activities of quality assurance.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	The student must relate to quality assurance plan							
CO2	The students must apply quality assurance tools & techniques in their project							
CO3	To learn about standards and certifications							
CO4	To describe procedures and work instructions in software organizations							

UNIT I-INTRODUCTION

The Software Quality Challenge - Software Quality Factors - Components of the Software Quality Assurance System. Pre-Project Software Quality Components - Contract Review - Development and Quality Plans

UNIT II-SOFTWARE QUALITY ASSURANCE COMPONENTS IN THE PROJECT LIFE CYCLE

Integrating Quality Activities in the Project Life Cycle – Reviews - Software Testing – Strategies - Software Testing – Implementation - Assuring the Quality of Software Maintenance - Assuring The Quality of External Participants' Parts – Case Tools and their Affect on Software Quality.

UNIT III-SOFTWARE QUALITY INFRASTRUCTURE COMPONENTS

Procedures and Work Instructions - Supporting Quality Devices - Staff Training, Instructing and Certification - Preventive and Corrective Actions – Configuration Management - Documentation and Quality Records Controls

UNIT IV-SOFTWARE QUALITY MANAGEMENT COMPONENTS

Project Progress Control- Components, Internal & External Participants, Progress control regimes, Computerized tools, Software Quality Metrics – Objective, Classification, Process & Product Metrics, Implementation & Limitation of Software Metrics - Software Quality Costs – Objective, Classification Model of cost, Extended Model and Applications.

REFERENCES

1. Daniel Galin, “Software Quality Assurance: From Theory to Implementation”, Pearson Addison-Wesley, 2012.
2. Roger S. Pressman, “Software Engineering-A Practitioner’s Approach”, McGraw Hill pub.2010.
3. Allen Gilles “Software quality: Theory and management”, International Thomson, Computer press 1997.
4. Stephen H.Kan, “Metrics and models in software quality Engineering”, Addison –Wesley 2003.
5. Humphrey Watts, “Managing the Software Process” Addison Wesley, 1986.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTSE-203	Language Technologies							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	This course enables to understand the importance and the benefits of software configuration and change management.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To learn the basic concepts of natural language processing							
CO2	To study the different techniques involved with information retrieval							
CO3	To learn about text mining							
CO4	To study the different scenarios and future directions							

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

UNIT I-INTRODUCTION

Natural Language Processing – Linguistic Background- Spoken language input and output Technologies – Written language Input - Mathematical Methods - Statistical Modeling and Classification Finite State methods

UNIT II-INFORMATION RETRIEVAL

Information Retrieval architecture - Indexing- Storage – Compression Techniques – Retrieval Approaches – Evaluation - Search engines- commercial search engine features- comparison- performance measures – Document Processing – NLP based Information Retrieval – Information Extraction.

UNIT III-TEXT MINING

Categorization – Extraction based Categorization- Clustering- Hierarchical Clustering- Document Classification and routing- finding and organizing answers from Text search – use of categories and clusters for organizing retrieval results – Text Categorization and efficient Summarization using Lexical Chains – Pattern Extraction.

UNIT IV-APPLICATIONS

Machine Translation – Transfer Metaphor - Interlingua and Statistical Approaches - Discourse Processing – Dialog and Conversational Agents – Natural Language Generation – Surface Realization and Discourse Planning.

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

REFERENCES

1. Daniel Jurafsky and James H. martin, “Speech and Language Processing”, Pearson Prentice Hall; 2 edition, 2008.
2. Ron Cole, J.Mariani, et.al “Survey of the State of the Art in Human Language Technology”, Cambridge University Press, 2007.
3. Michael W. Berry “Survey of Text Mining: Culstering, Classification and Retrieval”, Springer Verlag, 2003.
4. Christopher D.Manning and Hinrich Schutze, “Foundations of Statistical Natural Language Processing “, MIT Press, 2000.

MTSE-205	Personal Software Process							
Lecture	**Tutorial**	**Practical**	**Credit**	**Theory**	**Sessional**	**Practical**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**-**	**100**	**3 Hrs.**
Program Objective (PO)	To learn about how a software professional personally manages the software processes in all aspects.							
Course Outcomes (CO)								
After completion of course students will be able to								
CO1	To study how to manage and track the time for software processes.							
CO2	To learn how to schedule the process and manage the commitment.							
CO3	To learn about software Development process							
CO4	To learn how to estimate the product and process quality.							

UNIT I-INTRODUCTION AND TIME MANAGEMENT

Software Engineering – Personal Software Process – Improvement Process – Time Management – Logic of Time Management - Elements of Time Management – Categorizing your Activities – Gather Data on time spent by Activity – Evaluating your Time Distribution – Setting Ground rules – Prioritizing your time – Track Time – Recording your Time Data – Tracking your time – Handling Interruptions – Tracking Completed tasks.

UNIT II-MANAGING COMMITMENTS AND SCHEDULES

Defining Commitment – Responsibly made Commitment – Handling Missed Commitments – Importance of Managing Commitments – Consequences of not Managing Commitments – Way to Manage Commitments – Need for Schedules – Gantt Chart – Making a Project Schedule – Checkpoints – Tracking Project Plans – Tracking Earned Value

UNIT III-SOFTWARE PROCESSES AND QUALITY

Need for Processes – Process Script – Checkpoints and phases – Updated Project Plan Summary Form - Defects – Software Quality – Defects and Quality – Defects Versus Bugs – Defect Types – Understanding Defects – Defect Recording Log – Steps in Finding Defects – Ways to Find and Fix Defects.

UNIT IV-PRODUCT AND PROCESS QUALITY

Product Quality – Testing – The Filter view of Testing - Calculating yield values – Estimating the Ultimate Yield – Prototyping – Process Quality – Process Measures – Defect Removal Paradox – Defect Removal strategy – Appraisal/Failure ratio.

REFERENCES

1. Watts.S.Humphery, "*PSP: A Self-Improvement Process for Software Engineers"*, Addison Wesley, 2005.
2. Watts.S.Humphery, "*Introduction to the Personal Software Process*", Addison Wesley, 1997.
3. http://www.sei.cmu.edu/library/abstracts/reports/00tr022.cfm
4. http://repository.cmu.edu/cgi/viewcontent.cgi
5. http://dl.acm.org/citation.cfm?id=650271

Note for paper setter: Nine questions will be set in all. Question No. 1, which will be objective/ short answer type covering the entire syllabus, will be compulsory. The remaining eight questions will be set section-wise, with two questions from each unit. The candidate will be required to attempt FIVE questions in all with Q.1 (compulsory) and four other questions, selecting one question from each unit.

MTOE-201	Business Analytics						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	The main objective of this course is to give the student a comprehensive understanding of business analytics methods.						
Course Outcomes (CO)							
CO1	*Able to have knowledge of various business analysis techniques.*						
CO2	*Learn the requirement specification and transforming the requirement into different models.*						
CO3	*Learn the requirement representation and managing requirement assests.*						
CO4	*Learn the Recent Trends in Embedded and collaborative business*						

Unit 1

Business Analysis: Overview of Business Analysis, Overview of Requirements, Role of the Business Analyst. Stakeholders: the project team, management, and the front line, Handling, Stakeholder Conflicts.
Life Cycles: Systems Development Life Cycles, Project Life Cycles, Product Life Cycles, Requirement Life Cycles.

Unit 2

Forming Requirements: Overview of Requirements Attributes of Good Requirements, Types of Requirements, Requirement Sources, Gathering Requirements from Stakeholders, Common Requirements Documents.
Transforming Requirements: Stakeholder Needs Analysis, Decomposition Analysis, Additive/Subtractive Analysis, Gap Analysis, Notations (UML & BPMN), Flowcharts, Swim Lane Flowcharts, Entity-Relationship Diagrams, State-Transition Diagrams, Data Flow Diagrams, Use Case Modeling, Business Process Modeling

Unit 3

Finalizing Requirements: Presenting Requirements, Socializing Requirements and Gaining Acceptance, Prioritizing Requirements.
Managing Requirements Assets: Change Control, Requirements Tools

Unit 4

Recent Trends in: Embedded and collaborative business intelligence, Visual data recovery, Data Storytelling and Data Journalism.

References:

1. Business Analysis by James Cadle et al.
2. Project Management: The Managerial Process by Erik Larson and, Clifford Gray

MTOE-203	Industrial Safety						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	*To enable students to aware about the industrial safety.*						
Course Outcomes (CO)							
CO1	*Understand the industrial safety.*						
CO2	*Analyze fundamental of maintenance engineering.*						
CO3	*Understand the wear and corrosion and fault tracing.*						
CO4	*Understanding that when to do periodic inceptions and apply the preventing maintenance.*						

Unit-1

Industrial safety: Accident, causes, types, results and control, mechanical and electrical hazards, types, causes and preventive steps/procedure, describe salient points of factories act 1948 for health and safety, washrooms, drinking water layouts, light, cleanliness, fire, guarding, pressure vessels, etc, Safety color codes. Fire prevention and firefighting, equipment and methods.

Fundamentals of maintenance engineering: Definition and aim of maintenance engineering, Primary and secondary functions and responsibility of maintenance department, Types of maintenance, Types and applications of tools used for maintenance, Maintenance cost & its relation with replacement economy, Service life of equipment.

Unit-2

Wear and Corrosion and their prevention: Wear- types, causes, effects, wear reduction methods, lubricants-types and applications, Lubrication methods, general sketch, working and applications, i. Screw down grease cup, ii. Pressure grease gun, iii. Splash lubrication, iv. Gravity lubrication, v. Wick feed lubrication vi. Side feed lubrication, vii. Ring lubrication, Definition, principle and factors affecting the corrosion. Types of corrosion, corrosion prevention methods.

Unit-3

Fault tracing: Fault tracing-concept and importance, decision treeconcept, need and applications, sequence of fault finding activities, show as decision tree, draw decision tree for problems in machine tools, hydraulic, pneumatic,automotive, thermal and electrical equipment's like, I. Any one machine tool, ii. Pump iii. Air compressor, iv. Internal combustion engine, v. Boiler, vi. Electrical motors, Types of faults in machine tools and their general causes.

Unit-4

Periodic and preventive maintenance: Periodic inspection-concept and need, degreasing, cleaning and repairing schemes, overhauling of mechanical components, overhauling of electrical motor, common troubles and remedies of electric motor, repair complexities and its use, definition, need, steps and advantages of preventive maintenance. Steps/procedure for periodic and preventive maintenance of: I. Machine tools, ii. Pumps, iii. Air compressors, iv. Diesel generating (DG) sets Program and schedule of preventive maintenance of mechanical and electrical equipment, advantages of preventive maintenance. Repair cycle concept and importance

Reference:

1. Maintenance Engineering Handbook, Higgins & Morrow, Da Information Services.
2. Maintenance Engineering, H. P. Garg, S. Chand and Company.
3. Pump-hydraulic Compressors, Audels, Mcgrew Hill Publication.
4. Foundation Engineering Handbook, Winterkorn, Hans, Chapman & Hall London.

MTOE-205	Operations Research						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	To enable students to aware about the dynamic programming to solve problems of discreet and continuous variables and model the real world problem and simulate it.						
Course Outcomes (CO)							
CO1	*Students should able to apply the dynamic programming to solve problems of discreet and continuous variables.*						
CO2	*Students should able to apply the concept of non-linear programming*						
CO3	*Students should able to carry out sensitivity analysis*						
CO4	*Student should able to model the real world problem and simulate it.*						

Unit -1

Optimization Techniques, Model Formulation, models, General L.R Formulation, Simplex Techniques, Sensitivity Analysis, Inventory Control Models

Unit -2

Formulation of a LPP - Graphical solution revised simplex method - duality theory - dual simplex method - sensitivity analysis - parametric programming

Nonlinear programming problem - Kuhn-Tucker conditions min cost flow problem - max flow problem - CPM/PERT

Unit- 3

Scheduling and sequencing - single server and multiple server models - deterministic inventory models - Probabilistic inventory control models - Geometric Programming.

Unit -4

Competitive Models, Single and Multi-channel Problems, Sequencing Models, Dynamic Programming, Flow in Networks, Elementary Graph Theory, Game Theory Simulation

References:

1. H.A. Taha, Operations Research, An Introduction, PHI, 2008
2. H.M. Wagner, Principles of Operations Research, PHI, Delhi, 1982.
3. J.C. Pant, Introduction to Optimisation: Operations Research, Jain Brothers, Delhi, 2008
4. Hitler Libermann Operations Research: McGraw Hill Pub. 2009
5. Pannerselvam, Operations Research: Prentice Hall of India 2010
6. Harvey M Wagner, Principles of Operations Research: Prentice Hall of India 2010

MTOE-207	Cost Management of Engineering Projects						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	*To enable students to make aware about the cost management for the engineering project and apply cost models the real world projects.*						
Course Outcomes (CO)							
CO1	*Students should able to learn the strategic cost management process.*						
CO2	*Students should able to types of project and project team types*						
CO3	*Students should able to carry out Cost Behavior and Profit Planning analysis.*						
CO4	*Student should able to learn the quantitative techniques for cost management.*						

Unit-1

Introduction and Overview of the Strategic Cost Management Process Cost concepts in decision-making; relevant cost, Differential cost, Incremental cost and Opportunity cost.Objectives of a Costing System; Inventory valuation; Creation of a Database for operational control; Provision of data for Decision-Making.

Unit-2

Project: meaning, Different types, why to manage, cost overruns centres, various stages of project execution: conception to commissioning. Project execution as conglomeration of technical and nontechnical activities.Detailed Engineering activities. Pre project execution main clearances and documents Project team: Role of each member. Importance Project site: Data required with significance. Project contracts.Types and contents. Project execution Project cost control. Bar charts and Network diagram. Project commissioning: mechanical and process

Unit-3

Cost Behavior and Profit Planning Marginal Costing; Distinction between Marginal Costing and Absorption Costing; Break-even Analysis, Cost-Volume-Profit Analysis. Various decision-making problems.Standard Costing and Variance Analysis. Pricing strategies: Pareto Analysis. Target costing, Life Cycle Costing. Costing of service sector.Just-in-time approach, Material Requirement Planning, Enterprise Resource Planning, Total Quality Management and Theory of constraints.Activity-Based Cost Management, Bench Marking; Balanced Score Card and Value-Chain Analysis.Budgetary Control; Flexible Budgets; Performance budgets; Zero-based budgets.Measurement of Divisional profitability pricing decisions including transfer pricing.

Unit-4

Quantitative techniques for cost management, Linear Programming, PERT/CPM, Transportation problems, Assignment problems, Simulation, Learning Curve Theory.

References:

1. Cost Accounting A Managerial Emphasis, Prentice Hall of India, New Delhi
2. Charles T. Horngren and George Foster, Advanced Management Accounting
3. Robert S Kaplan Anthony A. Alkinson, Management & Cost Accounting
4. Ashish K. Bhattacharya, Principles & Practices of Cost Accounting A. H. Wheeler publisher
5. N.D. Vohra, Quantitative Techniques in Management, Tata McGraw Hill Book Co. Ltd.

MTOE-209	Composite Materials						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	*To enable students to aware about the composite materials and their properties.*						
Course Outcomes (CO)							
CO1	*Students should able to learn the Classification and characteristics of Composite materials.*						
CO2	*Students should able reinforcements Composite materials.*						
CO3	*Students should able to carry out the preparation of compounds.*						
CO4	*Student should able to do the analysis of the composite materials.*						

UNIT–1:

INTRODUCTION: Definition – Classification and characteristics of Composite materials. Advantages and application of composites.Functional requirements of reinforcement and matrix.Effect of reinforcement (size, shape, distribution, volume fraction) on overall composite performance.

REINFORCEMENTS: Preparation-layup, curing, properties and applications of glass fibers, carbon fibers, Kevlar fibers and Boron fibers. Properties and applications of whiskers, particle reinforcements. Mechanical Behavior of composites: Rule of mixtures, Inverse rule of mixtures. Iso-strain and Iso-stress conditions.

UNIT – 2

Manufacturing of Metal Matrix Composites: Casting – Solid State diffusion technique, Cladding – Hot isostaticpressing.Properties and applications. Manufacturing of Ceramic Matrix Composites: Liquid Metal Infiltration – Liquid phase sintering. Manufacturing of Carbon – Carbon composites: Knitting, Braiding, Weaving. Properties and applications.

UNIT–3

Manufacturing of Polymer Matrix Composites: Preparation of Moulding compounds and prepregs – hand layup method – Autoclave method – Filament winding method – Compression moulding – Reaction injection moulding. Properties and applications.

UNIT – 4

Strength: Laminar Failure Criteria-strength ratio, maximum stress criteria, maximum strain criteria, interacting failure criteria, hygrothermal failure. Laminate first play failure-insight strength; Laminate strength-ply discount truncated maximum strain criterion; strength design using caplet plots; stress concentrations.

TEXT BOOKS:

1. Material Science and Technology – Vol 13 – Composites by R.W.Cahn – VCH, West Germany.
2. Materials Science and Engineering, An introduction. WD Callister, Jr., Adapted by R.
3. Balasubramaniam, John Wiley & Sons, NY, Indian edition, 2007.

References:

1. Hand Book of Composite Materials-ed-Lubin.
2. Composite Materials – K.K.Chawla.
3. Composite Materials Science and Applications – Deborah D.L. Chung.
4. Composite Materials Design and Applications – Danial Gay, Suong V. Hoa, and Stephen W. Tasi.

MTOE-211	Waste to Energy						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**60**	**40**	**100**	**3 Hrs.**
Program Objective (PO)	*To enable students to aware about the generation of energy from the waste.*						
Course Outcomes (CO)							
CO1	*Students should able to learn the Classification of waste as a fuel.*						
CO2	*Students should able to learn the Manufacture of charcoal.*						
CO3	*Students should able to carry out the designing of gasifiers and biomass stoves.*						
CO4	*Student should able to learn the Biogas plant technology.*						

Unit-1

Introduction to Energy from Waste: Classification of waste as fuel – Agro based, Forest residue, Industrial waste - MSW – Conversion devices – Incinerators, gasifiers, digestors

Biomass Pyrolysis: Pyrolysis – Types, slow fast – Manufacture of charcoal – Methods - Yields and application – Manufacture of pyrolytic oils and gases, yields and applications.

Unit-2

Biomass Gasification: Gasifiers – Fixed bed system – Downdraft and updraft gasifiers – Fluidized bed gasifiers – Design, construction and operation – Gasifier burner arrangement for thermal heating – Gasifier engine arrangement and electrical power – Equilibrium and kinetic consideration in gasifier operation.

Unit-3

Biomass Combustion: Biomass stoves – Improved chullahs, types, some exotic designs, Fixed bed combustors, Types, inclined grate combustors, Fluidized bed combustors, Design, construction and operation - Operation of all the above biomass combustors.

Unit-4

Biogas: Properties of biogas (Calorific value and composition) - Biogas plant technology and status - Bio energy system - Design and constructional features - Biomass resources and their classification - Biomass conversion processes - Thermo chemical conversion - Direct combustion - biomass gasification - pyrolysis and liquefaction - biochemical conversion - anaerobic digestion - Types of biogas Plants – Applications - Alcohol production from biomass - Bio diesel production - Urban waste to energy conversion - Biomass energy programme in India.

References:

1. Non Conventional Energy, Desai, Ashok V., Wiley Eastern Ltd., 1990.
2. Biogas Technology - A Practical Hand Book - Khandelwal, K. C. and Mahdi, S. S., Vol. I & II, Tata McGraw Hill Publishing Co. Ltd., 1983.
3. Food, Feed and Fuel from Biomass, Challal, D. S., IBH Publishing Co. Pvt. Ltd., 1991.
4. Biomass Conversion and Technology, C. Y. WereKo-Brobby and E. B. Hagan, John Wiley & Sons, 1996.

MTAD-101	English For Research Paper Writing						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
2	**0**	**0**	**0**	**-**	**100**	**100**	**3 Hrs.**
Program Objective (PO)	*Student will able to understand the basic rules of research paper writing.*						
Course Outcomes (CO)							
CO1	*Understand that how to improve your writing skills and level of readability*						
CO2	*Learn about what to write in each section*						
CO3	*Understand the skills needed when writing a Title*						
CO4	*Ensure the good quality of paper at very first-time submission*						

Unit 1

Planning and Preparation, Word Order, Breaking up long sentences, Structuring Paragraphs and Sentences, Being Concise and Removing Redundancy, Avoiding Ambiguity and Vagueness

Unit 2

Clarifying Who Did What, Highlighting Your Findings, Hedging and Criticizing, Paraphrasing and Plagiarism, Sections of a Paper, Abstracts. Introduction

Unit 3

Review of the Literature, Methods, Results, Discussion, Conclusions, the Final Check. key skills are needed when writing a Title, key skills are needed when writing an Abstract, key skills are needed when writing an Introduction, skills needed when writing a Review of the Literature,

Unit4

Skills are needed when writing the Methods, skills needed when writing the Results, skills are needed when writing the Discussion, skills are needed when writing the Conclusions Useful phrases, how to ensure paper is as good as it could possibly be the first- time submission.

References:

1. Goldbort R (2006) Writing for Science, Yale University Press (available on Google Books)
2. Day R (2006) How to Write and Publish a Scientific Paper, Cambridge University Press
3. Highman N (1998), Handbook of Writing for the Mathematical Sciences, SIAM. Highman'sbook.
4. Adrian Wallwork, English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011

MTAD-103	Disaster Management						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
2	**0**	**0**	**0**	**-**	**100**	**100**	**3 Hrs.**
Program Objective (PO)	*Develop an understanding of disaster risk reduction and management*						
Course Outcomes (CO)							
CO1	*Learn to demonstrate a critical understanding of key concepts in disaster risk reduction and humanitarian response.*						
CO2	*Critically evaluate disaster risk reduction and humanitarian response policy and practice from multiple perspectives.*						
CO3	*Develop an understanding of standards of humanitarian response and practical relevance in specific types of disasters and conflict situations.*						
CO4	*critically understand the strengths and weaknesses of disaster management approaches, planning and programming in different countries, particularly their home country or the countries they work in*						

Unit 1

Disaster: Definition, Factors and Significance; Difference between Hazard and Disaster; Natural and Manmade Disasters: Difference, Nature, Types and Magnitude.

Unit 2

Repercussions of Disasters and Hazards: Economic Damage, Loss of Human and Animal Life, Destruction of Ecosystem. Natural Disasters: Earthquakes, Volcanisms, Cyclones, Tsunamis, Floods, Droughts And Famines, Landslides And Avalanches, Man-made disaster: Nuclear Reactor Meltdown, Industrial Accidents, Oil Slicks And Spills, Outbreaks Of Disease And Epidemics, War And Conflicts.

Unit 3

Study Of Seismic Zones; Areas Prone To Floods And Droughts, Landslides And Avalanches; Areas Prone To Cyclonic And Coastal Hazards With Special Reference To Tsunami; Post-Disaster Diseases And Epidemics Preparedness: Monitoring Of Phenomena Triggering A Disaster Or Hazard; Evaluation Of Risk: Application Of Remote Sensing, Data From Meteorological And Other Agencies, Media Reports: Governmental And Community Preparedness.

Unit 4

Disaster Risk: Concept and Elements, Disaster Risk Reduction, Global and National Disaster Risk Situation. Techniques of Risk Assessment, Global Co-Operation in Risk Assessment and Warning, People's Participation in Risk Assessment.Strategies for Survival.Meaning, Concept and Strategies of Disaster Mitigation, Emerging Trends in Mitigation.Structural Mitigation and Non-Structural Mitigation, Programs Of Disaster Mitigation in India.

References:

1. R. Nishith, Singh AK, "Disaster Management in India: Perspectives, issues and strategies "'New Royal book Company.
2. Sahni, PardeepEt.Al. (Eds.)," Disaster Mitigation Experiences And Reflections", Prentice Hall Of India, New Delhi.
3. Goel S. L., Disaster Administration And Management Text And Case Studies",Deep&Deep Publication Pvt. Ltd., New Delhi.

MTAD-105	Sanskrit for Technical Knowledge						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
2	**0**	**0**	**0**	**-**	**100**	**100**	**3 Hrs.**
Program Objective (PO)	*Students will be able to Understanding basic Sanskrit language and Ancient Sanskrit literature about science & technology can be understood and Being a logical language will help to develop logic in students*						
Course Outcomes (CO)							
CO1	*To get a working knowledge in illustrious Sanskrit, the scientific language in the world*						
CO2	*Learning of Sanskrit to improve brain functioning*						
CO3	*Learning of Sanskrit to develop the logic in mathematics, science & other subjects enhancing the memory power*						
CO4	*The engineering scholars equipped with Sanskrit will be able to explore the huge knowledge from ancient literature*						

Unit –1

Alphabets in Sanskrit, Past/Present/Future Tense, Simple Sentences.

Unit – 2

Order, Introduction of roots,Technical information about Sanskrit Literature

Unit –3

Technical concepts of Engineering: Electrical, Mechanical

Unit –4

Technical concepts of Engineering: Architecture, Mathematics

References

1. "Abhyaspustakam" – Dr.Vishwas, Samskrita-Bharti Publication, New Delhi
2. "Teach Yourself Sanskrit" PrathamaDeeksha-VempatiKutumbshastri, Rashtriya Sanskrit Sansthanam, New Delhi Publication
3. "India's Glorious Scientific Tradition" Suresh Soni, Ocean books (P) Ltd., New Delhi.

MTAD-107	Value Education						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
2	**0**	**0**	**0**	**-**	**100**	**100**	**3 Hrs.**
Program Objective (PO)	*Understand value of education and self- development, Imbibe good values in students and Let the should know about the importance of character*						
Course Outcomes (CO)							
CO1	*Knowledge of self-development*						
CO2	*Learn the importance of Human values*						
CO3	*Developing the overall personality*						
CO4	*Know about the importance of character*						

Unit 1

Values and self-development –Social values and individual attitudes.Work ethics, Indian vision of humanism.Moral and non- moral valuation.Standards and principles.Value judgements.

Unit 2

Importance of cultivation of values.Sense of duty.Devotion, Self-reliance.Confidence, Concentration.Truthfulness, Cleanliness.Honesty, Humanity.Power of faith, National Unity.Patriotism.Love for nature,Discipline

Unit 3

Personality and Behavior Development - Soul and Scientific attitude.Positive Thinking.Integrity and discipline.Punctuality, Love and Kindness. Avoid fault Thinking. Free from anger, Dignity of labour. Universal brotherhood and religious tolerance.True friendship. Happiness Vs suffering, love for truth. Aware of self-destructive habits.Association and Cooperation. Doing best for saving nature

Unit 4

Character and Competence –Holy books vs Blind faith.Self-management and Good health.Science of reincarnation. Equality, Nonviolence,Humility, Role of Women. All religions and same message. Mind your Mind, Self-control. Honesty, Studying effectively

References

1.Chakroborty, S.K. “Values and Ethics for organizations Theory and practice”, Oxford University Press, New Delhi

MTAD-102	Constitution of India						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
2	**0**	**0**	**0**	**-**	**100**	**100**	**3 Hrs.**
Program Objective (PO)	*Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective and to address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.*						
Course Outcomes (CO)							
CO1	*Discuss the growth of the demand for civil rights in India for the bulk of Indians before the arrival of Gandhi in Indian politics.*						
CO2	*Discuss the intellectual origins of the framework of argument that informed the conceptualization of social reforms leading to revolution in India.*						
CO3	*Discuss the circumstances surrounding the foundation of the Congress Socialist Party [CSP] under the leadership of Jawaharlal Nehru and the eventual failure of the proposal of direct elections through adult suffrage in the Indian Constitution.*						
CO4	*Discuss the passage of the Hindu Code Bill of 1956.*						

Unit I

History of Making of the Indian Constitution: History, Drafting Committee, (Composition & Working) Philosophy of the Indian Constitution: Preamble, Salient Features

Unit 2

Contours of Constitutional Rights & Duties: Fundamental Rights , Right to Equality , Right to Freedom , Right against Exploitation , Right to Freedom of Religion, Cultural and Educational Rights , Right to Constitutional Remedies , Directive Principles of State Policy , Fundamental Duties.

Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, Executive , President, Governor , Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications. Powers and Functions

Unit 3

Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative CEO of Municipal Corporation, Panchayati raj: Introduction, PRI: ZilaPanchayat, Elected officials and their roles, CEO ZilaPanchayat: Position and role. Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

Unit 4

Election Commission: Election Commission: Role and Functioning. Chief Election Commissioner and Election Commissioners. State Election Commission: Role and Functioning. Institute and Bodies for the welfare of SC/ST/OBC and women.

References

1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.
3. M. P. Jain, Indian Constitution Law, 7th Edn., Lexis Nexis, 2014.
4. D.D. Basu, Introduction to the Constitution of India, Lexis Nexis, 2015.

MTAD-104	Pedagogy Studies						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
2	**0**	**0**	**0**	**-**	**100**	**100**	**3 Hrs.**
Program Objective (PO)	*Review existing evidence on the review topic to inform programme design and policy making undertaken by the DFID, other agencies and researchers and Identify critical evidence gaps to guide the development*.						
Course Outcomes (CO)							
CO1	*What pedagogical practices are being used by teachers in formal and informal classrooms in developing countries?*						
CO2	*What is the evidence on the effectiveness of these pedagogical practices, in what conditions, and with what population of learners?*						
CO3	*How can teacher education (curriculum and practicum) and the school curriculum and guidance materials best support effective pedagogy?*						
CO4	*What is the importance of identifying research gaps?*						

Unit 1

Introduction and Methodology: Aims and rationale, Policy background, Conceptual framework and terminology , Theories of learning, Curriculum, Teacher education., Conceptual framework, Research questions. Overview of methodology and Searching. Thematic overview: Pedagogical practices are being used by teachers in formal and informal classrooms in developing countries. , Curriculum, Teacher education.

Unit 2

Evidence on the effectiveness of pedagogical practices, Methodology for the in depth stage: quality assessment of included studies. How can teacher education (curriculum and practicum) and the school curriculum and guidance materials best support effective pedagogy? Theory of change.Strength and nature of the body of evidence for effective pedagogical practices.Pedagogic theory and pedagogical approaches.Teachers' attitudes and beliefs and Pedagogic strategies.

Unit 3

Professional development: alignment with classroom practices and follow-up support, Peer support from the head teacher and the community. Curriculum and assessment, Barriers to learning: limited resources and large class sizes,

Unit 4

Research gaps and future directions: Research design, Contexts , Pedagogy, Teacher education Curriculum and assessment, Dissemination and research impact.

References

1. Ackers J, Hardman F (2001) Classroom interaction in Kenyan primary schools, Compare, 31 (2): 245-261.
2. Agrawal M (2004) Curricular reform in schools: The importance of evaluation, Journal of Curriculum Studies, 36 (3): 361-379.
3. Akyeampong K (2003) Teacher training in Ghana - does it count? Multi-site teacher education research project (MUSTER) country report 1. London: DFID.
4. Akyeampong K, Lussier K, Pryor J, Westbrook J (2013) Improving teaching and learning of basic maths and reading in Africa: Does teacher preparation count? International Journal Educational Development, 33 (3): 272–282.
5. Alexander RJ (2001) Culture and pedagogy: International comparisons in primary education. Oxford and Boston: Blackwell.
6. Chavan M (2003) Read India: A mass scale, rapid, 'learning to read' campaign.

MTAD-106	Stress Management by Yoga						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
2	**0**	**0**	**0**	**-**	**100**	**100**	**3 Hrs.**
Program Objective (PO)	To achieve overall health of body and mind and to overcome stress						
Course Outcomes (CO)							
CO1	*Develop healthy mind in a healthy body thus improving social health.*						
CO2	*Improve efficiency*						
CO3	*Learn the Yogasan*						
CO4	*Learn the pranayama*						

Unit – 1

Definitions of Eight parts of yog (Ashtanga).

Unit- 2

Yam and Niyam, Do`s and Don't's in life; Ahinsa, satya, astheya, bramhacharya and aparigraha; Shaucha, santosh, tapa, swadhyay, ishwarpranidhan.

Unit- 3

Asan and Pranayam, Various yog poses and their benefits for mind & body,

Unit- 4

Regularization of breathing techniques and its effects-Types of pranayam.

References

1. 'Yogic Asanas for Group Tarining-Part-I" :Janardan Swami YogabhyasiMandal, Nagpur
2. "Rajayoga or conquering the Internal Nature" by Swami Vivekananda, AdvaitaAshrama (Publication Department), Kolkata

MTAD-108	Personality Development through Life Enlightenment Skills						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
2	**0**	**0**	**0**	**-**	**100**	**100**	**3 Hrs.**
Program Objective (PO)	To learn to achieve the highest goal happily To become a person with stable mind, pleasing personality and determination To awaken wisdom in students						
Course Outcomes (CO)							
CO1	*Students become aware about leadership.*						
CO2	*Students will learn how to perform his/her duties in day to day work.*						
CO3	*Understand the team building and conflict*						
CO4	*Student will learn how to become role model for the society.*						

Unit – 1

Neetisatakam-Holistic development of personality: Verses: 19, 20, 21, 22 (wisdom); Verses: 29, 31, 32 (pride & heroism); Verses: 26, 28, 63, 65 (virtue); Verses: 52, 53, 59 (don's); Verses: 71, 73, 75, 78 (do's).

Unit – 2

Approach to day to day work and duties; ShrimadBhagwadGeeta: Chapter-2: Verses: 41, 47, 48; Chapter-3: Verses: 13, 21, 27, 35; Chapter-6: Verses: 5, 13, 17, 23, 35; Chapter-18: Verses: 45, 46, 48.

Unit - 3

Statements of basic knowledge; ShrimadBhagwadGeeta: Chapter-2: Verses: 56, 62, 68; Chapter-12: Verses: 13, 14, 15, 16, 17, 18.

Unit – 4

Personality of Role model; ShrimadBhagwadGeeta: Chapter-2: Verses: 17; Chapter-3: Verses: 36, 37, 42: Chapter-4: Verses: 18, 38, 39; Chapter-18: Verses: 37, 38, 63.

References:

1. Srimad Bhagavad Gita, Swami SwarupanandaAdvaita Ashram (Publication Department), Kolkata.
2. Bhartrihari's Three Satakam (Niti-sringar-vairagya), P. Gopinath, Rashtriya Sanskrit Sansthanam, New Delhi.

	Dissertation Part-I (MTSE-207) and Dissertation Part-II (MTSE-202)
Course Outcomes (CO)	
CO1	Ability to synthesize knowledge and skills previously gained and applied to an in depth study and execution of new technical problem.
CO2	Capable to select from different methodologies, methods and forms of analysis to produce a suitable research design, and justify their design.
CO3	Ability to present the findings of their technical solution in a written report.
CO4	Presenting the work in International/ National conference or reputed journals.

Syllabus Contents:

The dissertation / project topic should be selected / chosen to ensure the satisfaction of the urgent need to establish a direct link between education, national development and productivity and thus reduce the gap between the world of work and the world of study. The dissertation should have the following:

Relevance to social needs of society
Relevance to value addition to existing facilities in the institute
Relevance to industry need
Problems of national importance
Research and development in various domain

The student should complete the following:

Literature survey Problem Definition
Motivation for study and Objectives
Preliminary design / feasibility / modular approaches
Implementation and Verification
Report and presentation

The dissertation part- II is based on a report prepared by the students on dissertation allotted to them. It may be based on:

Experimental verification / Proof of concept.
The viva-voce examination will be based on the above report and work.

Guidelines for Dissertation Part – I and Dissertation Part - II

As per the AICTE directives, the dissertation is a yearlong activity, to be carried out and evaluated in two parts i.e. Part– I: July to December and Part– II: January to June.

The dissertation may be carried out preferably in-house i.e. department's laboratories and centers OR in industry allotted through department's T & P coordinator.

After multiple interactions with guide and based on comprehensive literature survey, the student shall identify the domain and define dissertation objectives.

The referred literature should preferably include IEEE/IET/IETE/Springer/Science Direct/ACM journals in the areas of Computing Engineering and any other related domain. In case of Industry sponsored projects, the relevant application notes, white papers, product catalogues should be referred and reported.

Student is expected to detail out specifications, methodology, resources required, critical issues involved in design and implementation and phase wise work distribution, and submit the proposal within a month from the date of registration.

Part–I deliverables: A document report comprising of summary of literature survey, detailed objectives, project specifications, paper, proof of concept/functionality, part results, and record of continuous progress.

Part–I evaluation: A committee comprising of guides of respective specialization shall assess the progress/performance of the student based on report, presentation and Q & A. In case of unsatisfactory performance, committee may recommend repeating the Part-I work.

During Part– II, student is expected to exert on design, development and testing of the proposed work as per the schedule. Accomplished results/contributions/innovations should be published in terms of research papers in reputed journals and reviewed focused conferences OR IP/Patents.

Part–II deliverables: A dissertation report as per the specified format, developed system in the form of hardware and/or software, and record of continuous progress.

Part-II evaluation: Guide along with appointed external examiner shall assess the progress/performance of the student based on report, presentation and Q & A. In case of unsatisfactory performance, committee may recommend for extension or repeating the Part-I work.