

**ACADEMIC REGULATIONS
COURSE STRUCTURE
AND
DETAILED SYLLABI
for
MASTER OF TECHNOLOGY
in
SOFTWARE ENGINEERING**

(For the batches admitted from 2016-2017)

CHOICE BASED CREDIT SYSTEM



**SREE VIDYANIKETHAN ENGINEERING COLLEGE
(AUTONOMOUS)**

**(Affiliated to JNTU Anantapur, Approved by AICTE
Programs Accredited by NBA; NAAC with 'A' grade)
Sree Sainath Nagar, A. Rangampet, Near Tirupati - 517 102. A.P.**

VISION

To be one of the Nation's premier Engineering Colleges by achieving the highest order of excellence in Teaching and Research.

MISSION

- To foster intellectual curiosity, pursuit and dissemination of knowledge.
- To explore students' potential through academic freedom and integrity.
- To promote technical mastery and nurture skilled professionals to face competition in ever increasing complex world.

QUALITY POLICY

Sree Vidyanikethan Engineering College strives to establish a system of Quality Assurance to continuously address, monitor and evaluate the quality of education offered to students, thus promoting effective teaching processes for the benefit of students and making the College a Centre of Excellence for Engineering and Technological studies.

DEPARTMENT OF INFORMATION TECHNOLOGY

VISION

To become a nationally recognized quality education center in the domain of Computer Science and Information Technology through teaching, training, learning, research and consultancy.

MISSION

- The Department offers undergraduate program in Information Technology and Post graduate program in Software Engineering to produce high quality information technologists and software engineers by disseminating knowledge through contemporary curriculum, competent faculty and adopting effective teaching-learning methodologies.
- Igniting passion among students for research and innovation by exposing them to real time systems and problems
- Developing technical and life skills in diverse community of students with modern training methods to solve problems in Software Industry.
- Inculcating values to practice engineering in adherence to code of ethics in multicultural and multi discipline teams

PROGRAM EDUCATIONAL OBJECTIVES

After few years of graduation, the graduates of M. Tech. (SE) Program would have:

1. Enrolled or completed research studies in the core or allied areas of Computer Science, Software Engineering and related areas.
2. Successful entrepreneurial or technical career in the core or allied areas of Computer Science and Software Engineering.
3. Continued to learn and to adapt to the world of constantly evolving technologies in the core or allied areas of Software Engineering.

PROGRAM OUTCOMES

On successful completion of the Program, the graduates of M. Tech. (SE) Program will be able to:

1. Demonstrate in-depth knowledge of specific discipline or professional area, including wider and global perspective, with an ability to discriminate, evaluate, analyze and synthesize existing and new knowledge, and integration of the same for enhancement of knowledge.
2. Analyze complex engineering problems critically, apply independent judgment for synthesizing information to make intellectual and/or creative advances for conducting research in a wider theoretical, practical and policy context.
3. Think laterally and originally, conceptualize and solve engineering problems, evaluate a wide range of potential solutions for those problems and arrive at feasible, optimal solutions after considering public health and safety, cultural, societal and environmental factors in the core areas of expertise.
4. Extract information pertinent to unfamiliar problems through literature survey and experiments, apply appropriate research methodologies, techniques and tools, design, conduct experiments, analyze and interpret data, demonstrate higher order skill and view things in a broader perspective, contribute individually/in group(s) to the development of scientific/technological knowledge in one or more domains of engineering.
5. Create, select, learn and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling, to complex engineering activities with an understanding of the limitations.
6. Possess knowledge and understanding of group dynamics, recognize opportunities and contribute positively to collaborative-multidisciplinary scientific research, demonstrate a capacity for self-management and teamwork, decision-making based on open-mindedness, objectivity and rational analysis in order to achieve common goals and further the learning of themselves as well as others.
7. Demonstrate knowledge and understanding of engineering and management principles and apply the same to one's own work, as a member and leader in a team, manage projects efficiently in respective disciplines and multidisciplinary environments after consideration of economical and financial factors.
8. Communicate with the engineering community, and with society at large, regarding complex engineering activities confidently and effectively, such as, being able to comprehend and write effective reports and design documentation by adhering to appropriate standards, make effective presentations, and give and receive clear instructions.

9. Recognize the need for, and have the preparation and ability to engage in life-long learning independently, with a high level of enthusiasm and commitment to improve knowledge and competence continuously.
10. Acquire professional and intellectual integrity, professional code of conduct, ethics of research and scholarship, consideration of the impact of research outcomes on professional practices and an understanding of responsibility to contribute to the community for sustainable development of society.
11. Observe and examine critically the outcomes of one's actions and make corrective measures subsequently, and learn from mistakes without depending on external feedback.

PROGRAM SPECIFIC OUTCOMES

On successful completion of the program, the graduates of M.Tech (SE) program will be able to:

1. Apply the knowledge of computer science and software engineering to provide solutions to real world problems.
2. Analyze, design, develop, deploy and maintain software systems and solutions.
3. Conduct investigations to identify research gaps and address complex engineering problems in software engineering domain.
4. Apply appropriate techniques, use modern programming languages, tools and packages for quality software development by thorough understanding of needs under given constraints

The Challenge of Change

“Mastery of change is in fact the challenge of moving human attention from an old state to a new state. Leaders can shift attention at the right time and to the right place. The real crisis of our times is the crisis of attention. Those who lead are the ones who can hold your attention and move it in a purposeful way. Transformation is nothing but a shift in attention from one form to another. The form of a beautiful butterfly breaks free from a crawling caterpillar. If you pay enough attention, you would be able to see how the butterfly hides within the caterpillar. The leader points out a butterfly when the follower sees only a caterpillar ”

- Debashis Chatterjee

ACADEMIC REGULATIONS

CHOICE BASED CREDIT SYSTEM

M. Tech. Regular Two Year Degree Program **(for the batches admitted from the academic year 2016–17)**

For pursuing Two year degree program of study in Master of Technology (M.Tech) offered by Sree Vidyanikethan Engineering College under Autonomous status and herein after referred to as SVEC (Autonomous):

- 1. Applicability :** All the rules specified herein, approved by the Academic Council, shall be in force and applicable to students admitted from the academic year 2016-2017 onwards. Any reference to "College" in these rules and regulations stands for SVEC (Autonomous).
- 2. Extent:** All the rules and regulations, specified hereinafter shall be read as a whole for the purpose of interpretation and as and when a doubt arises, the interpretation of the Chairman, Academic Council is final. It shall be ratified by Academic Council in the forth coming meeting. As per the requirements of statutory bodies, Principal, SVEC (Autonomous) shall be the Chairman, Academic Council.

3. Admission :

3.1. Admission into the Two Year M. Tech. Degree Program of study in Engineering:

3.1.1. Eligibility:

- A candidate seeking admission into the two year M. Tech Degree Program should have
 - (i) Passed B.Tech / B.E or equivalent Program recognized by JNTUA, Ananthapuramu, for admission as per the guidelines of Andhra Pradesh State Council of Higher Education (APSCHE).
 - (ii) A minimum percentage of marks in the qualifying degree as prescribed by the AICTE / UGC or Government at the time of admission.
 - (iii) Rank / score secured in the PGECET / GATE examination conducted by APSCHE/ MHRD for allotment of a seat by the convener PGECET, for admission.

3.1.2. Admission Procedure:

Admissions are made into the two year M.Tech. Degree Program as per the stipulations of APSCHE, Government of Andhra Pradesh:

- (a) By the Convener, PGECET (for Category–A Seats)
- (b) By the Management (for Category-B Seats).

4. Programs of study offered leading to the award of M.Tech. Degree and Eligibility:

Following are the two year postgraduate degree Programs of study offered in various branches at in SVEC (Autonomous) leading to the award of M.Tech. degree and eligibility to get admission into the Programs:

Name of the specialization	Offered by the Department	Name of the Degree / Branch eligible for Admission
Electrical Power Systems	EEE	B.E./B.Tech./AMIE in Electrical & Electronics Engg. (or) equivalent / Electrical Engineering
Digital Electronics and Communication Systems	ECE	B.E./B.Tech.in ECE/AMIE in ECE /AMIE Electronics and Telecommunication Engineering/AMIETE (Electronics & Telematics Engg.) / (Electronics & Computer Engg.) or equivalent. M.Sc.(Electronics) is not eligible.
Communication Systems		B.E./B.Tech.in ECE/AMIE in ECE /AMIE Electronics and Telecommunication Engineering/AMIETE (Electronics & Telematics Engg.) / (Electronics & Computer Engg.) or equivalent. M.Sc.(Electronics) is not eligible.
VLSI		B.E./B.Tech./AMIE in ECE/EEE/CSE/Electronics & Computers Engg./ETE/IT/CSIT/ Electronics & Control Engg. / Instrumentation Engg./ Instrumentation Technology/ EIE/ Electronics Engg. / Bio-Medical Engg. / AMIETE and Electronics and Telematics Engg. (OR) equivalent. M.Sc(Electronics) is not eligible.
Computer Science	CSE	B.E./B.Tech./AMIE in any branch of Engg. /Tech. (or) equivalent Master's Degree in Physics, Statistics, Mathematics or Applied Mathematics, Applied Statistics, Applied Physics, Geophysics, M.Sc (Comp. Sci), M.Sc (Information Systems), (Computer Applications and Electronics) and MCA. M.Sc (Electronics) or Equivalent.
Computer Networks and Information Security		B.E./B.Tech. /AMIE in CSE / ECE / CSIT / IT / ETM / EEE / EIE & CSSE equivalent. M.Sc. (Electronics) is not eligible.
Software Engineering	IT	B.E./B.Tech /AMIE in CSE / CSIT / Electronics & Computers Engg./ IT & Computer Science and Systems Engineering. Equivalent (or) MCA.

5. Duration of the Program:

5.1 Minimum Duration: The program will extend over a period of two years leading to the Degree of Master of Technology (M.Tech) of the JNTUA, Ananthapuramu. The two academic years will be divided into four semesters with two semesters per year. In first year, each semester shall normally consist of 22 weeks (≥ 90 working days) having - 'Continuous Internal Evaluation (CIE)' and 'Semester End Examination (SEE)'. In second year, each semester shall consists of 18 weeks and the entire year is for project work. Choice Based Credit System (CBCS) and Credit Based Semester System (CBSS) as suggested by UGC, and Curriculum/ Course Structure as suggested by AICTE are followed.

5.2 Maximum Duration: The student shall complete all the passing requirements of the M.Tech degree program within a maximum duration of 4 years including Gap year, this duration reckoned from the commencement of the semester to which the student was first admitted to the program.

I SEMESTER (22 weeks)	INSTRUCTION PERIOD:	I Spell : 7 Weeks	16 Weeks
		II Spell : 9 Weeks	
	Internal Examinations :	I Mid : 1 week	2 Weeks
		II Mid : 1 week	
	Preparation & Practical Examinations		2 Week

	External Examinations	2 Weeks
	Semester Break	2 Weeks
II SEMESTER (22 weeks)	INSTRUCTION PERIOD: I Spell : 7 Weeks II Spell : 9 Weeks	16 weeks
	Internal Examinations : I Mid : 1 week II Mid : 1 week	2 Weeks
	Preparation & Practical Examinations	2 Week
	External Examinations	2 Weeks
	Summer Vacation	4 Weeks
III SEMESTER	Project Work Phase – I	19 Weeks
IV SEMESTER	Project Work Phase – II	19 Weeks
	Project Work Viva-Voce examinations	2 Weeks

6. Course Structure: Each Program of study shall consist of:

- Professional core courses:

The list of professional core courses are chosen as per the suggestions of the experts, to impart knowledge and skills needed in the concerned specialization of study.

- Professional elective courses:

Professional elective courses shall be offered to the students to diversify their spectrum of knowledge and skills. The elective courses can be chosen based on the interest of the student to broaden his individual knowledge and skills.

- Audit Courses: Audit courses shall be offered to the students to diversify their knowledge.

Contact periods: Depending on the complexity and volume of the course the number of contact periods per week shall be assigned.

7. Credit System: All Courses are to be registered by a student in a Semester to earn Credits. Credits are assigned based on the following norms given in Table 1.

Table 1

Course	Periods/Week	Credits
Theory	01	01
Practical	04	02
Seminar	--	02
Project Work Phase-I	--	--
Project Work Phase-II	--	28

- As a norm, for the theory courses, **one credit** for one contact period per week is assigned.
- As a norm, for practical courses **two credits** will be assigned for four contact periods per week.
- For courses like Project/Seminar, where formal contact periods are not specified, credits are assigned based on the complexity of the work to be carried out.

iv. There are no credits for audit courses.

Other student activities like NCC, NSS, Sports, Study Tour, Guest Lecture etc. will not carry Credits.

The two year curriculum of any M. Tech Degree Program of study shall have total of **86** credits (28 credits in I Semester, 30 credits in II Semester and 28 credits in IV Semester).

8. Choice Based Credit System (CBCS):

Choice Based Credit System (CBCS) is introduced based on UGC guidelines in order to promote:

- Student centered learning
 - Cafeteria approach
 - Students to learn courses of their choice
 - Learning at their own pace
 - Interdisciplinary learning
- A student is introduced to "Choice Based Credit System (CBCS)"
 - The total credits for the Programme is 86.
 - A student has choice of registering for credits from the theory courses offered in the program ensuring the total credits in a semester are between 24 and 34.
 - In I Semester, the student has the option of registering for one additional theory course from the latter semester or dropping one existing theory course from the current semester within the course structure of the program. In II Semester also, the student has the option of registering for one additional theory course from the previous semester if dropped earlier within the course structure of the program. However the maximum number of credits the student can register in a particular semester cannot exceed 33 credits.
 - Grade points, based on percentage of marks awarded for each course will form the basis for calculation of SGPA (Semester Grade Point Average) and CGPA (Cumulative Grade Point Average).
 - All the registered credits will be considered for the calculation of final CGPA.

9. Course Enrollment and Registration

- 9.1 Each student, on admission shall be assigned to a Faculty Advisor (Mentor) who shall advice and counsel the student about the details of the academic programme and the choice of courses considering the student's academic background and career objectives.
- 9.2 The enrollment of courses in I Semester will commence on the day of admission. If the student wishes, the student may drop or add courses (vide clause 8) within **three** days before commencement of I semester class work and complete the registration process. The student shall enroll for the courses with the help of the student's Faculty Advisor (Mentor). The enrollment of courses in II Semester will commence 10 days prior to the last instructional day of the I semester and complete the registration process for all the remaining theory courses as per program course structure, duly authorized by the Chairman, Board of studies of concern department.
- 9.3 If any student fails to register the courses in a semester, he shall undergo the courses as per the program structure.
- 9.4 After registering for a course, a student shall attend the classes, satisfy the attendance requirements, earn Continuous Assessment marks and appear for the Semester-end Examinations.
- 9.5 No elective course shall be offered by a Department unless a minimum of 8 students register for the course.

10. Massive Open Online Course (MOOC)

A Massive Open Online Course (MOOC) is an online course aimed at unlimited participation and open access via the web. MOOC is a model for delivering learning content online to any person who takes a course, with no limit on attendance.

- A student shall undergo a "Massive Open Online Course (MOOC)" for award of the degree besides other requirements.
- A student is offered this Online Course at the beginning of his II Semester of study and the course has to be completed by the end of III Semester. If the student fails to complete the course by the end of III Semester, it shall be treated as a backlog and needs to be completed before completion of the program for the award of the degree.
- The student shall confirm registration by enrolling the course within 10 days prior to the last instructional day of the I semester like other courses.
- The courses will be approved by the Chairman, Academic Council, SVEC based on the recommendations of the Chairman, Board of Studies of concerned program considering current needs.
- A student has a choice of registering for only one MOOC with the recommendation of Chairman, Board of studies of concerned program and duly approved by the Chairman, Academic Council, SVEC.
- The student shall undergo MOOC without disturbing the normal schedule of regular class work.
- One faculty member assigned by the Head of the Department shall be responsible for the periodic monitoring of the course implementation.
- No formal lectures need be delivered by the faculty member assigned to the students.
- If any student wants to change the MOOC course already registered, he will be given choice to register a new MOOC course in M. Tech. II / III Semester only, with the recommendation of Chairman, Board of studies of concerned program and duly approved by the Chairman, Academic Council, SVEC.
- Finally, the performance of the student in the course will be evaluated as stipulated by the course provider. A certificate will be issued on successful completion of the course by the course provider.
- The performance in the MOOC will not be considered for the calculation of SGPA and CGPA of the student.
- The MOOC course will be listed in the grade sheets of the student.

11. Break of Study from a Programme (Gap Year)

11.1 A student is permitted to go on break of study for a maximum period of one year.

11.2 The student shall apply for break of study in advance, in any case, not later than the last date of the first assessment period in a semester. The application downloaded from website and duly filled by the student shall be submitted to the Head of the Department. In the case of start-up for incubation of idea only, the application for break of study shall be forwarded by the Head of the Department to the Principal, SVEC. A sub-committee appointed by the principal shall give recommendations for approval.

11.3 The students permitted to rejoin the programme after break of study shall be governed by the Curriculum and Regulations in force at the time of rejoining. The

students rejoining in new regulations shall apply to the Principal, SVEC in the prescribed format through Head of the Department at the beginning of the readmitted semester itself for prescribing additional/equivalent courses, if any, from any semester of the regulations in-force, so as to bridge the curriculum in-force and the old curriculum.

11.4 The total period for completion of the programme reckoned from the commencement of the I Semester to which the student was admitted shall not exceed the maximum period specified in clause 5.2 irrespective of the period of break of study in order that the student may be eligible for the award of the degree (vide clause 19).

11.5 If a student has not reported to the department after approved period of break of study without any intimation, the student is treated as detained in that semester. Such students are eligible for readmission for the semester when offered next.

12. Examination System: All components in any Program of study shall be evaluated through internal evaluation and / or an external evaluation conducted as semester-end examination.

12.1. Distribution of Marks:

Sl. No.	Course	Marks	Examination and Evaluation	Scheme of examination
1.	Theory	60	Semester-end examination of 3 hours duration (External evaluation)	The examination question paper in theory courses shall be for a maximum of 60 marks. The question paper shall be of descriptive type with 5 questions, taken one from each unit of syllabus, having internal choice and all 5 questions shall be answered. All questions carry equal marks.
		40	Mid-term Examination of 2 hours duration (Internal evaluation).	The question paper shall be of descriptive type with 5 essay type questions out of which 4 are to be answered and evaluated for 40 marks. Two mid-term examinations each for 40 marks are to be conducted. For a total of 40 marks, 75% of better one of the two and 25% of the other one are added and finalized. Mid-I: After first spell of instruction (I to II Units). Mid-II: After second spell of instruction (III to V Units).
2	Laboratory	50	Semester-end Lab Examination for 3 hours duration (External evaluation)	50 marks are allotted for laboratory examination during semester-end.
		50	30	Day-to-Day evaluation for

Sl. No.	Course	Marks	Examination and Evaluation		Scheme of examination
				Performance in laboratory experiments and Record. (Internal evaluation).	Practical test, each for 50 marks are to be evaluated. For a total of 50 marks 75% of better one of the two and 25% of the other one are added and finalized. Laboratory examination-I: Shall be conducted just before I mid-term examinations. Laboratory examination-II: Shall be conducted just before II mid-term examinations.
			20	Practical test (Internal evaluation).	
3	Audit Course	-	-		Audit course will be conducted as given in 12.2.1
4	Seminar	100	Semester-end Examination		100 marks are allotted for Seminar during semester-end evaluation by the Departmental Committee (DC) as given in 12.2.2.
5	Project Work	400	200	External evaluation	Semester-end Project Viva-Voce Examination by Committee as detailed in 12.2.3.
			200	Internal evaluation	Continuous evaluation by the DC as detailed in 12.2.3.

12.2 Audit Course/ Seminar/Project Work Evaluation:

12.2.1. Audit Course: For audit course, attendance shall be maintained like in case of any regular course. Students may be encouraged to submit assignments and give presentations on the course topics. There won't be any examinations for audit courses. However, the courses shall be listed in the grade sheet of the student.

12.2.2. Seminar: For the seminar, the student shall collect information through literature survey on a specialized topic and prepare a technical report, showing his understanding over the topic, and submit to the Department a week before presentation. The report and the presentation shall be evaluated at the end of the semester during the period of preparation and practicals by the Departmental Committee (DC) consisting of two senior faculty members and concerned supervisor of the department. The DC is constituted by the Principal on the recommendations of the Head of the Department. The department shall have individual DCs for each M. Tech. Program with senior faculty members and the supervisor specialized in the program.

12.2.3. Project Work:

12.2.3.1. Student shall register for the Project work with the approval of DC in the III Semester and continue the work in the IV Semester too. The DC shall monitor the progress of the project work. In III Semester, Phase-I of the Project Work has to be completed. A Student has to identify the topic of work, collect relevant Literature, preliminary data, implementation tools/methodologies etc., and perform a critical study and analysis of the problem identified. He shall submit status report in two different phases in addition to oral presentation before the DC for evaluation and award of internal marks at the end of Phase -I. A candidate shall continue the Project Work in IV Semester (Phase - II) and submit a Project report at the end of Phase-II after approval of the DC. During Phase-II, the student shall submit status report in

two different phases, in addition to oral presentation before the DC. The DC shall evaluate the project based on the progress, presentations and quality of work. A candidate shall be allowed to submit the dissertation only after passing all the courses from 1st to 3rd semesters and on recommendations of the DC. The Viva-Voce examination shall be conducted as per the IV Semester examinations schedule.

12.2.3.2 Three copies of the dissertation certified in the prescribed form by the concerned Supervisor and HOD shall be submitted to the Department. One copy is to be submitted to the Chief Controller of Examinations, SVEC (Autonomous) and one copy to be sent to the examiner. The examiner shall be nominated by the Chief Controller of the Examinations from the panel of three examiners submitted by the Department for a maximum of 5 students at a time for adjudication.

12.2.3.3 If the report of the examiner is favorable, Viva-Voce examination shall be conducted by a board consisting of the concerned Supervisor, Head of the Department and the examiner who adjudicated the dissertation. The board shall jointly evaluate the candidates project work. If the report of the examiner is not favorable, the candidate should revise and resubmit the project report followed by Viva-Voce examination.

12.2.3.4 The candidates who fail in Viva-Voce examination shall have to re-appear the Viva-Voce examination after three months. Extension of time within the total permissible limit for completing the project is to be obtained from the Chairman, Academic Council, SVEC (Autonomous).

12.2.3.5 If a candidate desires to change the topic of the project already chosen, during Phase-II, he has to re-register for Project work with the approval of the DC and repeat Phases-I & II. Marks already earned in Phase-I stand cancelled.

12.2.3.6 If a candidate unable to complete the project work after Phase-II and desires to change the topic of the project already chosen, he has to re-register for Project work with the approval of the DC and repeat Phases-I & II. Marks already earned in Phase-I & II stand cancelled.

12.3. Eligibility to appear for the semester-end examination:

12.3.1 A student shall be eligible to appear for semester-end examinations if he acquires a minimum of 75% of attendance in aggregate of all the courses in a semester.

12.3.2 Condonation of shortage of attendance in aggregate up to 10% (65% and above and below 75%) in each semester may be granted by the College Academic Committee.

- 12.3.3** Shortage of attendance below 65% in aggregate shall in no case be condoned.
- 12.3.4** Students whose shortage of attendance is not condoned in any semester shall not be eligible to take their semester-end examination and their registration shall stand cancelled.
- 12.3.5** A student shall not be promoted to the next semester unless he satisfies the attendance requirements of the semester, as applicable. The student may seek readmission for the semester when offered next. He will not be allowed to register for the courses of the semester while he is in detention.
- 12.3.6** A stipulated fee shall be payable to the college towards condonation of shortage of attendance.
- 12.4. Evaluation:** Following procedure governs the evaluation.
- 12.4.1.** Marks for components evaluated internally by the faculty should be submitted to the Controller of Examinations one week before the commencement of the semester-end examinations. The marks for the internal evaluation components shall be added to the external evaluation marks secured in the semester-end examinations, to arrive at total marks for any course in that semester.
- 12.4.2.** Performance in all the courses is tabulated course-wise and shall be scrutinized by the Results Committee and moderation is applied if needed, and course-wise marks are finalized. Total marks obtained in each course are converted into letter grades.
- 12.4.3.** Student-wise tabulation shall be done and individual grade sheet shall be generated and issued.
- 12.5. Personal verification / Revaluation / Recounting:**
Students shall be permitted for personal verification/request for recounting/revaluation of the semester-end examination answer scripts within a stipulated period after payment of prescribed fee. After recounting or revaluation, records shall be updated with changes if any and the student shall be issued a revised grade sheet. If there are no changes, the student shall be intimated the same through a notice.
- 12.6. Supplementary Examination:**
In addition to the regular semester-end examinations conducted, the College may also schedule and conduct supplementary examinations for all the courses of other semesters when feasible for the benefit of students. Such of the candidates writing supplementary examinations may have to write more than one examination per day.
- 13. Re-Registration for Improvement of Internal Marks:**
Following are the conditions to avail the benefit of improvement of internal evaluation marks.
- 13.1** The candidate should have completed the course work and obtained examinations results for I and II semesters.
- 13.2** Out of the courses the candidate has failed in the examinations due to internal evaluation marks secured being less than 50%, the candidate shall be given one chance for a maximum of 3 theory courses for improvement of internal evaluation marks.

- 13.3** He should have passed all the remaining courses for which the internal evaluation marks secured more than or equal to 50%.
- 13.4** The candidate has to register for the chosen courses and fulfill the academic requirements.
- 13.5** For each course, the candidate has to pay a fee equivalent to one third of the semester tuition fee and the amount is to be remitted in the form of D.D./Challan in favour of the Principal, Sree Vidyanikethan Engineering College payable at Tirupati along with the requisition through the concerned Head of the Department.
- 13.6** In the event of availing the Improvement of Internal evaluation marks, the internal evaluation marks as well as the semester-end examinations marks secured in the previous attempt(s) for the re-registered courses stand cancelled.
- 14. Academic Requirements for completion of M.Tech Program of study:**
- The following academic requirements have to be satisfied in addition to the attendance requirements for completion of M.Tech Program of study.
- 14.1 A student shall be deemed to have satisfied the minimum academic requirements for each theory, laboratory and project work, if he secures not less than 40% of marks in the semester-end examination and a minimum of 50% of marks in the sum total of the internal evaluation and semester-end examination taken together. For the seminar, he should secure not less than 50% of marks in the semester-end examination.
- 14.2 A student shall register for all the 86 credits and earn all the 86 credits. Marks obtained in the 86 credits shall be considered for the calculation of the DIVISION based on CGPA.
- 14.3 A student who fails to earn 86 credits as indicated in the curriculum within **four** academic years from the year of his admission shall forfeit his seat in M.Tech. Program and his admission stands cancelled.
- 15. Transitory Regulations:**
- Students who got detained for want of attendance (**or**) who have not fulfilled academic requirements (**or**) who have failed after having undergone the Program in earlier regulations (**or**) who have discontinued and wish to continue the Program are eligible for admission into the unfinished semester from the date of commencement of class work with the same (**or**) equivalent courses as and when courses are offered and they will be in the academic regulations into which they are presently readmitted.
- A regular student has to satisfy all the eligibility requirements within the maximum stipulated period of **four years** for the award of M.Tech Degree.
- 16. Grades, Grade Point Average and Cumulative Grade Point Average:**
- 16.1. Grade System:** After all the components and sub-components of any course (including laboratory courses) are evaluated, the final total marks obtained shall be converted to letter grades on a "**10 point scale**" as described below.

Grades conversion and Grade points allotted

% of Marks obtained	Grade	Description of Grade	Grade Points (GP)
≥ 95	O	Outstanding	10
≥ 85 to < 95	S	Superior	9
≥ 75 to < 85	A	Excellent	8
≥ 65 to < 75	B	Very Good	7
≥ 55 to < 65	C	Good	6
≥ 50 to < 55	D	Pass	5
< 50	F	Fail	0
Not Appeared	N	Absent	0

Pass Marks: A student shall be declared to have passed theory course, laboratory course and project work if he secures minimum of 40% marks in Semester-end examination, and a minimum of 50% marks in the sum total of internal evaluation and Semester-end examination taken together. For the seminar, he shall be declared to have passed if he secures minimum of 50% of marks in the semester-end examinations. Otherwise he shall be awarded fail grade - **F** in such a course irrespective of internal marks. **F** is considered as a fail grade indicating that the student has to pass the semester-end examination in that course in future and obtain a grade other than **F** and **N** for passing the course.

16.2. Semester Grade Point Average (SGPA):

Semester Grade Point Average (SGPA) shall be calculated as given below on a "10 point scale" as an index of the student's performance at the end of each semester:

$$SGPA = \frac{\sum(C \times GP)}{\sum C}$$

where **C** denotes the credits assigned to the courses undertaken in that semester and **GP** denotes the grade points earned by the student in the respective courses.

Note: SGPA is calculated only for the candidates who passed all the courses in that semester.

16.3. Cumulative Grade Point Average (CGPA):

The CGPA for any student is awarded only when he completes the Program i.e., when the student passes in all the courses prescribed in the Program. The CGPA is computed on a 10 point scale as given below:

$$CGPA = \frac{\sum(C \times GP)}{\sum C}$$

where **C** denotes the credits assigned to courses undertaken up to the end of the Program and **GP** denotes the grade points earned by the student in the respective courses.

17. Grade Sheet: A grade sheet (Marks Memorandum) shall be issued to each student indicating his performance in all courses registered in that semester indicating the SGPA.

18. Transcripts: After successful completion of the entire Program of study, a transcript containing performance in all academic years shall be issued as a final record. Duplicate transcripts will also be issued, if required, after payment of requisite fee. Partial transcript will also be issued upto any point of study to a student on request.

19. Award of Degree: The Degree shall be conferred and awarded by Jawaharlal Nehru Technological University Anantapur, Ananthapuramu on the recommendations of the Chairman, Academic Council, SVEC (Autonomous).

19.1. Eligibility: A student shall be eligible for the award of M.Tech Degree if he fulfills all the following conditions:

- Registered and successfully completed all the components prescribed in the Program of study to which he is admitted.
- Successfully acquired the minimum required credits as specified in the curriculum corresponding to the Program of study within the stipulated time.
- Obtained CGPA greater than or equal to 5.0 (Minimum requirement for declaring as passed).
- Has no dues to the College, Hostel, Library etc. and to any other amenities provided by the College.
- No disciplinary action is pending against him.

19.2. Award of Division: Declaration of division is based on CGPA.

Awarding of Division

CGPA	Division
≥ 7.0	First Class with Distinction
≥ 6.0 and < 7.0	First Class
≥ 5.0 and < 6.0	Second Class

20. Additional academic regulations:

20.1 A student may appear for any number of supplementary examinations within the stipulated time to fulfill regulatory requirements for award of the degree.

20.2 In case of malpractice/improper conduct during the examinations, guidelines shall be followed as shown in the **Annexure-I**.

20.3 When a student is absent for any examination (Mid-term or Semester-end) he shall be awarded **zero** marks in that component (course) and grading will be done accordingly.

20.4 When a component is cancelled as a penalty, he shall be awarded zero marks in that component.

21. Withholding of Results:

If the candidate has not paid dues to the College/University (or) if any case of indiscipline is pending against him, the result of the candidate shall be withheld and he will not be allowed/promoted to the next higher semester

22. Amendments to regulations:

The Academic Council of SVEC (Autonomous) reserves the right to revise, amend, or change the Regulations, Scheme of Examinations, and / or Syllabi or any other policy relevant to the needs of the society or industrial requirements etc., with the recommendations of the concerned Board(s) of Studies.

23. General:

The words such as "he", "him", "his" and "himself" shall be understood to include all students irrespective of gender connotation.

Note: *Failure to read and understand the regulations is not an excuse.*

**GUIDE LINES FOR DISCIPLINARY ACTION FOR MALPRACTICES /
IMPROPER CONDUCT IN EXAMINATIONS**

Rule No.	Nature of Malpractices/ Improper conduct	Punishment
	<i>If the candidate:</i>	
1. (a)	Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, Cell phones, pager, palm computers or any other form of material concerned with or related to the course of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the candidate which can be used as an aid in the course of the examination)	Expulsion from the examination hall and cancellation of the performance in that course only.
(b)	Gives assistance or guidance or receives it from any other candidate orally or by any other body language methods or communicates through cell phones with any candidate or persons in or outside the exam hall in respect of any matter.	Expulsion from the examination hall and cancellation of the performance in that course only of all the candidates involved. In case of an outsider, he will be handed over to the police and a case is registered against him.
2.	Has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the course of the examination (theory or practical) in which the candidate is appearing.	Expulsion from the examination hall and cancellation of the performance in that course and all other courses the candidate has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the courses of that Semester. The Hall Ticket of the candidate is to be cancelled.
3.	Impersonates any other candidate in connection with the examination.	The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred for four consecutive semesters from class work and all Semester-end examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat. The performance of the original candidate who has been impersonated, shall be cancelled in all the courses of the examination (including labs and project work) already appeared and shall not be allowed to appear for examinations of the remaining courses of that semester. The candidate is also debarred for four consecutive semesters from class work and all Semester-end examinations, if his involvement is established. Otherwise, The candidate is debarred for two consecutive semesters from class work and all Semester-end examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat. If the imposter is an outsider, he will be handed over to the police and a case is registered against him.
4.	Smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination.	Expulsion from the examination hall and cancellation of performance in that course and all the other courses the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the courses of that semester. The candidate is also debarred for two consecutive semesters from class work and all Semester-end examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5.	Uses objectionable, abusive or offensive language in the answer paper or in letters to the	Cancellation of the performance in that course only.

	examiners or writes to the examiner requesting him to award pass marks.	
6.	Refuses to obey the orders of the Chief Controller of Examinations/Controller of Examinations/any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the Controller of Examinations or any person on duty in or outside the examination hall of any injury to his person or to any of his relations whether by words, either spoken or written or by signs or by visible representation, assaults the Controller of Examinations, or any person on duty in or outside the examination hall or any of his relations, or indulges in any other act of misconduct or mischief which result in damage to or destruction of property in the examination hall or any part of the College campus or engages in any other act which in the opinion of the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.	In case of students of the college, they shall be expelled from examination halls and cancellation of their performance in that course and all other courses the candidate(s) has (have) already appeared and shall not be permitted to appear for the remaining examinations of the courses of that semester. If the candidate physically assaults the invigilator/Controller of the Examinations, then the candidate is also debarred and forfeits his/her seat. In case of outsiders, they will be handed over to the police and a police case is registered against them.
7.	Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that course and all the other courses the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the courses of that semester. The candidate is also debarred for two consecutive semesters from class work and all Semester-end examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
8.	Possess any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that course and all other courses the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the courses of that semester. The candidate is also debarred and forfeits the seat.

Note: Whenever the performance of a student is cancelled in any course(s) due to Malpractice, he has to register for Semester-end Examinations in that course(s) consequently and has to fulfill all the norms required for the award of Degree.

SREE VIDYANIKETHAN ENGINEERING COLLEGE

Sree Sainath Nagar, Tirupati – 517 102.

SVEC16 M. Tech. SE Course Structure

I-Semester

Course Code	Course Title	L	T	P	Periods per week	Credits	Scheme of Examination Max. Marks		
							Int.	Ext.	Total
16MT12501	Cloud Computing	4	-	-	4	4	40	60	100
16MT12502	Data Structures and Algorithms	4	-	-	4	4	40	60	100
16MT12503	Software Development Methodologies	4	-	-	4	4	40	60	100
16MT12504	Software Measurement and Metrics	4	-	-	4	4	40	60	100
16MT12505	Software Requirements and Estimation	4	-	-	4	4	40	60	100
	Professional Elective-1	4	-	-	4	4	40	60	100
16MT12506	Distributed Databases								
16MT12507	Machine Learning								
16MT12508	Software Reliability and Reuse								
16MT12509	User Interface Design								
16MT12531	Advanced Software Engineering Lab -1	-	-	4	4	2	50	50	100
16MT12532	Cloud Computing Lab	-	-	4	4	2	50	50	100
	Total:	24	-	8	32	28	340	460	800
16MT13808	Research Methodology (Audit Course)	-	2	-	2	-	-	-	-

II-Semester

Course Code	Course Title	L	T	P	Periods per week	Credits	Scheme of Examination Max. Marks		
							Int.	Ext.	Total
16MT22501	Big Data Technologies	4	-	-	4	4	40	60	100
16MT22502	Service Oriented Architecture	4	-	-	4	4	40	60	100
16MT22503	Software Architecture & Design Patterns	4	-	-	4	4	40	60	100
16MT22504	Software Testing Techniques	4	-	-	4	4	40	60	100
16MT22505	Web Technologies	4	-	-	4	4	40	60	100
	Professional Elective-2	4	-	-	4	4	40	60	100
16MT10502	Advanced Database Management Systems								
16MT22506	Software Process and Project Management								
16MT22507	Software Reverse Engineering								
16MT22508	Software Security								
16MT22531	Advanced Software Engineering Lab-2	-	-	4	4	2	50	50	100
16MT22532	Big Data Technologies Lab	-	-	4	4	2	50	50	100
16MT22533	Seminar	-	-	-	-	2	--	100	100
	Total:	24	-	8	32	30	340	560	900
16MT23810	Intellectual Property Rights (Audit Course)	--	2	-	2	-		-	-

III-Semester

Course Code	Course Title	L	T	P	Periods per week	Credits	Scheme of Examination Max. Marks		
							Int.	Ext.	Total
16MT3MOOC	MOOC	-	-	-	-	-	-	-	-
16MT32531	Project Work – Phase I	-	-	-	-	-	100	--	100
	Total:	-	-	-	-	-	100	--	100

*Fulltime Project Work

IV-Semester

Course Code	Course Title	L	T	P	Periods per week	Credits	Scheme of Examination Max. Marks		
							Int.	Ext.	Total
16MT42531	Project Work – Phase II	-	-	-	-	28	100	200	300
	Total:	-	-	-	-	28	100	200	300
	Grand Total:					86	880	1220	2100

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. I Semester
(16MT12501) CLOUD COMPUTING

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Operating Systems" and "Computer Networks"

COURSE DESCRIPTION: Virtualization, Case studies – XEN, VMware, Microsoft Hyper-V; Cloud architecture; Services and Applications; Cloud Programming; Industry practices and Case studies –Amazon Web Services, Google App Engine, and Microsoft Azure.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Demonstrate knowledge on Virtualization models, Cloud Architecture, Services and Programming concepts.
2. Analyze the problems in existing cloud architectures.
3. Apply concurrent programming, throughput computing and Data intensive computing in Cloud programming.
4. Develop research insights into emerging technologies and energy management.
5. Apply virtualization techniques to optimize resource sharing.

DETAILED SYLLABUS

UNIT-I: INTRODUCTION TO VIRTUALIZATION (Periods: 09)

Characteristics of Virtualized Environments, Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Pros and Cons of Virtualization, Technology Examples – XEN, VMware, Microsoft Hyper-V.

UNIT-II: CLOUD ARCHITECTURE (Periods: 11)

Introduction to Cloud: Defining Cloud Computing, Cloud Types - The NIST model, The Cloud Cube Model, Deployment models, Service models, Examining the Characteristics of Cloud Computing, Paradigm shift, Benefits of cloud computing, Disadvantages of cloud computing, Assessing the Role of Open Standards.

Cloud Architecture: Exploring the Cloud Computing Stack, Composability, Infrastructure, Platforms, Virtual Appliances, Communication Protocols, and Applications.

UNIT-III: DEFINING CLOUD SERVICES (Periods: 10)

Defining Infrastructure as a Service (IaaS) – IaaS workloads, Pods, aggregation, and silos, **Defining Platform as a Service (PaaS)**, **Defining Software as a Service (SaaS)** – SaaS characteristics, Open SaaS and SOA, Salesforce.com and CRM SaaS, **Defining Identity as a Service (IDaaS)** – what is an identity? Networked identity service classes, Identity system codes of conduct, IDaaS interoperability, **Defining Compliance as a Service (CaaS)**.

UNIT-IV: CLOUD PROGRAMMING CONCEPTS**(Periods: 12)**

Concurrent Programming – Introduction to Parallelism for Single Machine Computation, Programming Applications with Threads, **High Throughput Computing** – Task Programming, Task based Application Models, **Data Intensive Computing** – What is Data Intensive Computing and Technologies for Data Intensive Computing.

UNIT-V: INDUSTRIAL PLATFORMS AND TRENDING DEVELOPMENTS (Periods: 13)

Case Studies on Cloud Platforms – Amazon Web Services, Google App Engine, and Microsoft Azure, Case Studies on Cloud Applications – Scientific Applications, Business and Consumer Applications.

Enhancements in Cloud – Energy Efficiency in Clouds, Market based Management of Clouds, Federated Clouds / InterCloud, Third Party Cloud Services.

[Total Periods: 55]**TEXT BOOKS:**

1. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, "*Mastering Cloud Computing: Foundations and Applications Programming*," 1st Edition, McGraw Hill, 2013.
2. Barrie Sosinsky, "*Cloud Computing Bible*," 1st Edition, Wiley India Pvt Ltd, 2011.

REFERENCE BOOKS:

1. Anthony T. Velte, Toby J. Velte Robert Elsenpeter, "*Cloud Computing: A Practical Approach*," 1st Edition, Tata Mc Graw Hill, 2010.
2. George Reese, "*Cloud Application Architectures*," 1st Edition, O'Reilly Publishers, 2010.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. I Semester
(16MT12502) DATA STRUCTURES AND ALGORITHMS
(Common to M.Tech. (SE and CS))

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A course on "Computer Programming".

COURSE DESCRIPTION:

Introduction to Data Structures and Algorithms; Searching and Sorting; Trees and Graphs; Divide and Conquer; Greedy method; Dynamic Programming, Back Tracking; Branch and Bound.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on:
 - Linear data structures including Stack, Queue and Linked Lists and Non-linear data structures like Trees and Graphs.
 - Divide and Conquer Method, Greedy Method, Dynamic Programming, Backtracking and Branch & Bound algorithms.
2. Analyze the efficiency of algorithms using space and time complexities.
3. Apply algorithm design techniques in providing solutions to real world problems.
4. Apply Dynamic programming techniques to provide software solutions.

DETAILED SYLLABUS:

UNIT-I: INTRODUCTION TO DATA STRUCTURES AND ALGORITHMS (Periods: 10)

Data Structures: Review of Data Structures - Stack, Queue, Circular Queue, Linked Lists, Applications, **Algorithm Analysis:** Efficiency of algorithms, Apriori Analysis, Asymptotic Notations, Polynomial vs Exponential Algorithms, Average, Best and Worst Case Complexities, Analyzing Recursive Algorithms.

UNIT-II: SEARCHING, SORTING AND TREES & GRAPHS (Periods: 09)

Searching and Sorting: Linear Search, Fibonacci Search, Counting Sort, Bucket Sort, Radix Sort, **Trees and Graphs:** Introduction to trees, representation of trees, binary trees, binary tree traversal techniques, Introduction to graphs, representation of graphs, graph traversal techniques.

UNIT-III: BINARY SEARCH TREES, AVL TREES, B- TREES AND HASH TABLES

(Periods: 10)

Binary Search Trees: Definition, Operations, Applications, **AVL Trees:** Definition, Operations, Applications, **Heaps:** Definition, Heap Implementation, Applications, **Hash Tables:** Definition, Hash Functions, Applications.

UNIT-IV: DIVIDE AND CONQUER & GREEDY METHODS (Periods: 10)

Divide and Conquer: General Method, Binary Search, Finding Maximum and Minimum, Quick Sort, Merge sort, Strassen's Matrix Multiplication.

Greedy Method: General Method, Job sequencing with deadlines, Minimum Cost Spanning Tree and Single Source Shortest Path.

UNIT-V: DYNAMIC PROGRAMMING, BACK TRACKING & BRANCH AND BOUND

(Periods: 11)

Dynamic Programming: General Method, All Pairs Shortest Path, 0/1 Knapsack problem, Traveling Salesperson Problem, **Back Tracking:** General Method, 8 – Queen’s Problem, Graph Coloring, **Branch and Bound:** General Method, LC Search, LIFO and FIFO branch and bound solutions of 0/1 Knapsack Problem.

[Total Periods: 50]

TEXT BOOKS:

1. G. A. V. Pai, "*Data Structures and Algorithms: Concepts, Techniques and Applications*," 1st Edition, Tata McGraw Hill, 2008.
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "*Fundamentals of Computer Algorithms*," 2nd Edition, Universities Press (India) Pvt. Ltd, 2008.

REFERENCE BOOKS:

1. Richard Gileberg, Behrouz A. Forouzan, "*Data Structures: A Pseudocode Approach with C*," Second Edition, 2007.
2. Mark Allen Weiss, "*Data Structures and Algorithm Analysis in C++*," 3rd Edition, Pearson Education, 2007.
3. Sartaj Sahni, "*Data structures, Algorithms and Applications in C++*," 2nd Edition, Universities press (India) Pvt. Ltd., 2005.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT12503) SOFTWARE DEVELOPMENT METHODOLOGIES

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A course on "Software Engineering".

COURSE DESCRIPTION: Introduction-Software Process; Software Requirements and Analysis; Software Design; Software Implementation- Implementation Issues, Modern Programming Language Features; Software Testing and Maintenance.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Demonstrate in depth knowledge on:
 - Software Paradigms, Agile Development, Software Reuse, and Testing
2. Perform requirements analysis and build requirements model.
3. Apply advanced software engineering methods in software development life cycle.

DETAILED SYLLABUS

UNIT-I: INTRODUCTION (Periods: 12)

Software Engineering: Software Process-Generic process model, Prescriptive process model, specialized, unified process. Agile development-Agile Process, Extreme Programming, Adaptive Software Development(ASD), Scrum, Dynamic Systems Development Method(DSDM), Crystal, Feature Driven Development(FDD), Lean Software Development(LSD), Agile Modeling(AM), Agile Unified Process (AUP). Software Engineering Knowledge-core Principles, Principles that guide each framework Activity.

UNIT-II: SOFTWARE REQUIREMENTS AND ANALYSIS (Periods: 10)

Establishing the Groundwork, Eliciting Requirements, Developing use cases, Building the requirements model, Negotiating, Validating Requirements. Requirements Analysis, Requirements Modeling Strategies.

UNIT-III: SOFTWARE DESIGN (Periods: 11)

Design Process, Design concepts - Abstraction, Architecture, Patterns, Separation of Concerns, Modularity, Information Hiding, Functional Independence, Refinement, Aspects, Refactoring, Object Oriented Design Concepts, Design Classes. Design Model- Data, Architectural, Interface, Component, Deployment Level Design Elements. Design Techniques- Stepwise Refinement, Level of Abstraction, Structure Design, Integrated Top-Down Development, Jackson Structured Programming, Summary of Design Techniques.

UNIT-IV: SOFTWARE IMPLEMENTATION (Periods: 10)

Implementation Issues: Structured coding Techniques, Coding Styles, Standards and Guidelines, Documentation Guidelines.

Modern Programming Language Features: Type checking, User defined data types, Data Abstraction, Exception Handling, Concurrency Mechanism.

UNIT-V: SOFTWARE TESTING AND MAINTENANCE (Periods: 12)

Testing: Strategic Approach to software Testing, Strategic Issues, Testing Strategies for Conventional Software, Object oriented software and Web Apps, Validating Testing, System Testing, Art of Debugging.

Maintenance: Software Maintenance, Enhancing Maintainability during Development, Managerial Aspects of Software Maintenance, Configuration Management, Software Supportability, Reengineering, Business Process Reengineering, Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering, Economics of Reengineering.

[Total Periods: 55]

TEXT BOOKS:

1. Roger S. Pressman, "*Software Engineering – A Practitioner’s Approach*," 7th Edition, Tata McGraw-Hill, 2009.
2. Richard Fairley, "*Software Engineering Concepts*," Tata McGraw Hill, 2008.

REFERENCE BOOKS:

1. Ian Sommerville, "*Software Engineering*," 7th Edition, Pearson Education Asia, 2007.
2. Shari Lwarence Pfleeger, Joanne M. Atlee, "*Software Engineering Theory and Practice*," 3rd Edition, Pearson Education, 2006.

M.Tech. (SE) I Semester (16MT12504) SOFTWARE MEASUREMENT AND METRICS

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PRE-REQUISITES: A course on "Software Engineering".

COURSE DESCRIPTION: Software measurement theory; Models of software engineering measurement; Software products metrics, software process metrics; Measuring & management and Software quality metrics.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge in:
 - Data collection and measures.
 - Product attributes
 - Software quality and Object oriented metrics.
2. Gain skills in analyzing what to measure and complexity assessment in software development.
3. Initiate research to improve Software Estimation and Quality in software development.
4. Apply OO metric tools for software measurement.
5. Apply project cost calculation procedures in software development.

DETAILED SYLLABUS

UNIT-I: FUNDAMENTALS OF MEASUREMENT (Periods: 10)

Measurement and Basics of Measurement - Measurement in Everyday Life, Measurement in Software Engineering, Scope of Software Metrics, Representational Theory of Measurement, Measurement and Models, Measurement Scales and Scale Types.

UNIT-II: ANALYSIS OF MEASUREMENT (Periods: 11)

Goal-Based Frame Work for Software Measurement - Classifying Software Measures, Determining what to measure, Applying Frame Work, Software Measurement Validation

Software Metrics Data Collection - Good Data, Definition of Data, Collecting, Storing and Extracting Data.

UNIT-III: PRODUCT ATTRIBUTES (Periods: 12)

Measuring Internal Product Attributes – Size - Aspects of software size, Length, Reuse, Functionality, Complexity.

Structure - Types of structural measures, Control-flow structure, Modularity and information flow attributes.

UNIT-IV: MEASUREMENT AND MANAGEMENT (Periods: 11)

Measuring External Product Attributes - Modeling Software Quality, Measuring Aspects of Quality.

Object-Oriented Metrics - Object-Oriented Concepts and Constructs, Design and Complexity metrics, Productivity Metrics, Quality and Quality Management Metrics.

UNIT-V: QUALITY METRICS**(Periods: 11)**

Software Quality Metrics Overview - Product Quality Metrics, In-Process Quality Metrics, Metrics for Software Maintenance, Examples of Metrics Programs-Motorola, HP, IBM, Collecting Software Engineering Data, Applying the Seven Basic Quality Tools in Software Development.

[Total Periods: 55]**TEXT BOOKS:**

1. Fenton, Pfleeger, "*Software Metrics*," 2nd Edition, Thomson, 2005.
2. Stephen H. Kan, "*Metrics and Models in Software Quality Engineering*," 2nd Edition, Addison Wesley, 2011.

REFERENCE BOOKS:

1. Linda M. Laird and Carol Brennan, "*Software Measurement and Estimation - A Practical Approach*," IEEE Computer Science Press and Wiley Inter Science, 2006.
2. C Ravindranath Pandian: "*Software Metrics: A guide to Planning Analysis and Implementation*," Auerbach Publications, 2005.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT12505) SOFTWARE REQUIREMENTS AND ESTIMATION

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Software Engineering," and "Object Oriented Analysis & Design".

COURSE DESCRIPTION:

Introduction- Software requirements and risk management, Software Requirements Engineering; Requirements management, Software Requirements Modeling; Software Estimation, Size Estimation; Effort, Schedule and Cost Estimation; Requirements Management Tools, Software Estimation Tools.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on:
 - Requirements Engineering and Management
 - Estimation of software - size, effort, schedule and cost.
2. Analyze the problems in estimation & factors influencing estimation and build traceability matrix, links in requirement chain.
3. Solve size and cost estimation for software development using COCOMO II, Putnam Estimation and Algorithmic models.
4. Apply requirement management and estimation tools for software development.
5. Gain the understanding of the requirements engineering and management principles for effective software implementation.

DETAILED SYLLABUS

UNIT-I: SOFTWARE REQUIREMENTS (Periods: 15)

Essential Software requirement, Good practices for requirements engineering, Improving requirements processes, Software requirements and risk management.

Software Requirements Engineering: Requirements elicitation, requirements analysis documentation, review, elicitation techniques, analysis models, Software quality attributes, risk reduction through prototyping, setting requirements priorities, verifying requirements quality.

UNIT-II: SOFTWARE REQUIREMENTS MANAGEMENT (Periods: 11)

Requirements management, Principles and Practices, Requirements Attributes, Change Management Process, Requirements Traceability Matrix, Links in requirements chain.

Software Requirements Modeling: Use case modeling, Analysis models, Data flow diagrams, State transition diagrams, Class diagrams, and Object analysis.

UNIT-III: SOFTWARE ESTIMATION (Periods: 10)

Components of Software Estimations, Estimation methods, Problems associated with estimation, Key project factors that influence estimation.

Size Estimation-Two views of sizing, Function Point Analysis, Mark II FPA, Full Function Points, LOC Estimation, and Conversion between size measures.

UNIT-IV: EFFORT, SCHEDULE AND COST ESTIMATION (Periods: 10)

Introduction to Productivity, Estimation Factors, Approaches to Effort and Schedule Estimation, COCOMO II, Putnam Estimation Model, Algorithmic models, Cost Estimation.

UNIT-V: REQUIREMENTS MANAGEMENT TOOLS (Periods: 09)

Benefits of using a requirements management tool, commercial requirements management tool, Rational Requisite Pro, Caliber – RM, Implementing requirements management automation.

SOFTWARE ESTIMATION TOOLS

Desirable features in software estimation tools, IFPUG, USC's COCOMO II, SLIM (Software Life Cycle Management) Tools.

[Total Periods: 55]

TEXT BOOKS:

1. Karl E. Weigers, "*Software Requirements*," 2nd Edition, Microsoft Press, 2013.
2. Rajesh Naik and Swapna Kishore, "*Software Requirements and Estimation*," Tata McGraw Hill, 2001.

REFERENCE BOOKS:

1. Dean Leffingwell and Don Widrig, "*Managing Software Requirements*," Pearson Education, 2003.
2. Suzanne Robertson and James Robertson, "*Mastering the Requirements Process*," 2nd Edition, Pearson Education, 2006.
3. Capers Jones, "*Estimating Software Costs*," 2nd Edition, Tata McGraw-Hill, 2007.
4. M.A. Parthasarathy, "*Practical Software Estimation*," 1st Edition, Pearson Education, 2007.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT12506) DISTRIBUTED DATABASES
(PROFESSIONAL ELECTIVE-I)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Database Management Systems" and "Computer Networks"

COURSE DESCRIPTION: Distributed Databases Overview & Distributed Database Design; Translation of Global Queries to Fragment Queries, Optimization of Access Strategies; Management of Distributed Transactions, Distributed Database administration; Concurrency, Reliability; Case studies.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on query processing and optimization.
2. Gain analytical skills to implement distributed and parallel databases.
3. Initiate research in advanced Query Optimization, authorization and protection in database.
4. Apply transaction management techniques in distributed environment.

DETAILED SYLLABUS:

UNIT-I: DISTRIBUTED DATABASES OVERVIEW & DISTRIBUTED DATABASE DESIGN
(Periods: 10)

Distributed databases:

An overview, Level of Distribution Transparency: Reference Architecture for Distributed databases, Types of Data Fragmentation.

Distributed Database Design

A framework for Distributed Database Design, Design of Database Fragmentation, Allocation of fragments.

UNIT-II: TRANSLATION OF GLOBAL QUERIES TO FRAGMENT QUERIES AND OPTIMIZATION OF ACCESS STRATEGIES
(Periods: 12)

Translation of Global Queries to Fragment Queries

Global Queries, fragment Queries, Equivalence Transformations for Queries, transforming Global Queries into Fragment Queries, Distributed Grouping and Aggregate Function Evaluation, Parameter Queries.

Optimization of Access Strategies

Frame Work for Query Optimization, Join Queries, General Queries.

UNIT-III: MANAGEMENT OF DISTRIBUTED TRANSACTIONS & DISTRIBUTED DATABASE ADMINISTRATION (Periods: 09)

Management of Distributed Transactions

Framework for Transaction Management, Supporting Atomicity of Distributed Transactions, Concurrency Control for Distributed Transactions, Architectural aspects of Distributed Transactions.

Distributed Database administration

Catalog management in distributed Databases, authorization and protection.

UNIT-IV: CONCURRENCY & RELIABILITY (Periods: 12)

Concurrency

Foundations of Distributed Concurrency Control, Distributed Deadlocks, Concurrency Control Based on Timestamp.

Reliability

Basic concepts, Nonblocking Commitment Protocols, Reliability and Concurrency Control, Determining a Consistent View of the Network, Detection and Resolution of Inconsistency, Check points and Cold Restart.

UNIT-V: CASE STUDIES (Periods: 07)

Case studies: Tandem's ECOMPASS Distributed Database System, DDM: A distributed database manager based on Adaplex, Distributed -INGRESS, POREL MULTIBASE.

[Total Periods: 50]

TEXT BOOK:

1. Stefano Ceri. Giuseppe Pelagatti, "*Distributed Databases: Principles and Systems*", 1985, MCG

REFERENCE BOOK:

1. Ozsu, "*Principles of Distributed Database Systems*", 1st Edition, 2002, PEA.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT12507) MACHINE LEARNING
(PROFESSIONAL ELECTIVE-I)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: --Nil--

COURSE DESCRIPTION: Machine learning fundamentals, applications; Multivariate methods, Bayesian networks, Decision tree learning; Support Vector Machines, Statistical learning methods, Unsupervised learning; Kernel Machines; Combining Multiple Learners and Reinforcement learning.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on supervised, unsupervised, and reinforcement machine learning techniques.
2. Solve real-life problems using Multivariate Methods, Decision Trees, Kernel Machines and Combining Multiple Learners.
3. Initiate research in pattern recognition, classification and clustering techniques.

DETAILED SYLLABUS

UNIT-I: INTRODUCTION

(Periods: 11)

Introduction to Machine Learning, Examples of Machine Learning Applications. Supervised Learning: Learning a Class from Examples, Vapnik-Chervonenkis (VC) Dimension, Probably Approximately Correct (PAC) Learning, Noise, Learning Multiple Classes, Regression, Model Selection and Generalization and Dimensions of a Supervised Machine Learning Algorithm. Bayesian Decision Theory: Classification, Losses and Risks, Discriminant Functions, Utility Theory and Association Rules.

UNIT-II: MULTIVARIATE METHODS

(Periods: 11)

Multivariate Data, Parameter Estimation, Estimation of Missing Values, Multivariate Normal Distribution, Multivariate Classification, Tuning Complexity, Discrete Features, Multivariate Regression. Dimensionality Reduction: Subset Selection, Principal Components Analysis, Factor Analysis, Multidimensional Scaling, Linear Discriminant Analysis, Isomap, Locally Linear Embedding. Clustering: Mixture Densities, k-Means Clustering, Expectation-Maximization Algorithm, Mixtures of Latent Variable Models, Supervised Learning after Clustering, Hierarchical Clustering, Choosing the Number of Clusters.

UNIT-III: DECISION TREES

(Periods: 11)

Univariate Trees, Pruning, Rule Extraction from Trees, Learning Rules from Data, Multivariate Trees. Linear Discrimination: Generalizing the Linear Model, Geometry of the Linear Discriminant, Pairwise Separation, Parametric Discrimination Revisited, Gradient Descent, Logistic Discrimination, and Discrimination by Regression. Multilayer Perceptrons: The Perceptron, Training a Perceptron, Learning Boolean Functions, Multilayer Perceptrons, MLP as a Universal Approximator, Back propagation Algorithm, Training Procedures, Tuning the Network Size, Bayesian View of Learning, Dimensionality Reduction, Learning Time.

UNIT-IV: KERNEL MACHINES

(Periods: 11)

Optimal Separating Hyperplane, The Nonseparable Case: Soft Margin Hyperplane, v-SVM, Kernel Trick, Vectorial Kernels, Defining Kernels, Multiple Kernel Learning, Multiclass Kernel Machines, Kernel Machines for Regression, One-Class Kernel Machines, Kernel Dimensionality Reduction. Bayesian Estimation: Estimating the Parameter of a Distribution, Bayesian Estimation of the Parameters of a Function, Gaussian Processes. Hidden Markov Models: Discrete Markov Processes, Hidden Markov Models, Three Basic Problems of HMMs, Evaluation Problem, Finding the State Sequence, Learning Model Parameters, Continuous Observations, The HMM with Input, Model Selection in HMM.

UNIT-V: COMBINING MULTIPLE LEARNERS

(Periods: 11)

Rationale, Generating Diverse Learners, Model Combination Schemes, Voting, Error-Correcting Output Codes, Bagging, Boosting, Mixture of Experts Revisited, Stacked Generalization, Fine-Tuning an Ensemble, Cascading. Reinforcement Learning: Single State Case: K-Armed Bandit, Elements of Reinforcement Learning, Model-Based Learning, Temporal Difference Learning, Generalization, Partially Observable States.

[Total Periods: 55]

TEXT BOOK:

1. Ethem Alpaydin, "*Introduction to Machine Learning*," 2nd Edition, MIT Press, 2009.

REFERENCE BOOKS:

1. Tom M. Mitchell, "*Machine Learning*," McGraw-Hill, 2013.
2. Christopher M. Bishop, "*Pattern Recognition and Machine Learning*," 2nd Edition, Springer-Verilog, 2006.
3. Kevin P. Murphy, "*Machine Learning: A Probabilistic Perspective*," MIT Press, 2012.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT12508) SOFTWARE RELIABILITY AND REUSE
(PROFESSIONAL ELECTIVE-I)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Software Engineering" and "Probability and Statistics".

COURSE DESCRIPTION: Software reliability engineering process, Software reliability strategies, availability; Software reliability modeling; Software metrics for reliability assessment; Best practice of software reliability engineering, and neural networks for software reliability, software system failures, free software intensive system and reusable components.

COURSE OUTCOMES:

On successful completion of this course, the student will be able to:

1. Gain knowledge on:
 - Software Reliability Modeling.
 - Software metrics for Reliability Assessment.
 - Software Reliability Estimation.
 - Best practices of Software Reliability Engineering.
2. Analyze software system failures and operational profile.
3. Solve Software system reliability issues using optimum reliability models.
4. Initiate research in producing failure free software intensive system.
5. Apply advanced methods to analyze complex legacy software systems and identify reusable components.

DETAILED SYLLABUS

UNIT – I: SOFTWARE RELIABILITY (Periods: 10)

Software Reliability Ideas of Software Reliability, Computation of software reliability, Classes of software reliability Models.

Time Dependent Software Reliability Models: Time between failure reliability Models, Fault Counting Reliability Models.

UNIT – II: TIME INDEPENDENT SOFTWARE RELIABILITY MODELS (Periods: 08)

Fault injection model of Software Reliability, Input Domain Reliability Model, Orthogonal defect classification, Software availability Models.

Software Reliability Modeling: A general procedure for reliability modeling.

UNIT – III: SOFTWARE REUSE (Periods: 10)

Introduction Software Reuse and Software Engineering, Concepts and Terms, Software Reuse products, Software Reuse processes, Software reuse paradigms. State of the Art and the Practice: Software Reuse Management, Software Reuse Techniques, Aspects of Software Reuse, Organizational Aspects, Technical Aspects and Economic Aspects.

UNIT –IV: PROGRAMMING PARADIGMS AND REUSABILITY (Periods: 10)

Usability Attributes, Representation and Modeling Paradigms, Abstraction and Composition in development paradigm.

Object - Oriented Domain Engineering: Abstraction and parameterization techniques, Composition techniques in Object Orientation.

UNIT-V: APPLICATION ENGINEERING (Periods: 10)

Component Storage and Retrieval, Reusable Asset Integration. Software Reuse technologies: Component Based Software Engineering, COTS based development, Software Reuse Metrics, Tools for Reusability.

[Total Periods: 48]

TEXT BOOKS:

1. Michael R. Lyu, "*Handbook of Software Reliability Engineering*," IEEE Computer Society Press, McGraw-Hill Book Company, 2005.
2. Ivar Jacobson, Martin Gress, Patrick Johnson, "*Software Reuse*," Pearson Education, 2004.

REFERENCE BOOKS:

1. John D. Musa, "*Software Reliability Engineering*," 2nd Edition, Tata Mc GrawHill, 2011.
2. Eve-Andre Karisson, "*Software Reuse – A Holistic Approach*," John Wiley and Sons, 1996.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT12509) USER INTERFACE DESIGN
(PROFESSIONAL ELECTIVE-I)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A course on "Software Development Methodologies"

COURSE DESCRIPTION: Characteristics & principles of User Interface Design; Requirement analysis-direct & indirect methods; Design- using Formatting menus & windows; Design-using Text boxes, multimedia and Windows layout.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain the knowledge on concepts of user interfaces and related business functions.
2. Analyze user requirements necessary for UI development.
3. Design interfaces using appropriate menus, windows, interfaces.
4. Solve real world problems by applying theoretical user interface concepts.
5. Usage and customize of advanced tools for various window layouts in project management and development of UI computing systems.

UNIT-I: INTRODUCTION **(Periods: 09)**
Human-Computer Interface – Characteristics of Graphics Interface –Direct Manipulation Graphical System – Web User Interface –Popularity –Characteristic & Principles- User Interface Design Process – Obstacles -Usability

UNIT-II: HUMAN COMPUTER INTERACTION **(Periods: 12)**
Human Characteristics In Design – Human Interaction Speed –Business Functions – Requirement Analysis – Direct –Indirect Methods – Basic Business Functions – Design Standards – System Timings –Human Consideration In Screen Design – Structures of Menus – Functions of Menus–Contents of Menu

UNIT-III: FORMATTING MENUS AND WINDOWS **(Periods: 09)**
Formatting – Phrasing the Menu – Selecting Menu Choice–Navigating Menus– Graphical Menus.
Windows: Characteristics– Components– Presentation Styles– Types– Managements– Organizations– Operations– Web Systems– Device– Based Controls Characteristics– Screen – Based Controls – Operate Control.

UNIT-IV: TEXT BOXES AND MULTIMEDIA **(Periods: 11)**
Text Boxes– Selection Control–Combination Control– Custom Control– Presentation Control Text for Web Pages – Effective Feedback– Guidance & Assistance–Internationalization– Accessibility– Icons– Image– Multimedia – Coloring.

UNIT-V: WINDOWS LAYOUT **(Periods: 09)**
Prototypes – Kinds of Tests – Retest – Information Search – Visualization –Hypermedia – WWW– Software Tools.

[Total Periods: 50]

TEXT BOOKS:

1. Wilbent. O. Galitz, "*The Essential Guide To User Interface Design*", John Wiley& Sons, 2001.
2. Ben Sheiderman, "*Design the User Interface*", Pearson Education, 1998.

REFERENCE BOOK:

1. Alan Cooper, "*The Essential Of User Interface Design*", Wiley – Dream Tech Ltd., 2002.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT12531) ADVANCED SOFTWARE ENGINEERING LAB-1

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	4	2

PREREQUISITES: Courses on “Software Development Methodologies”, “Software Requirements and Estimation”, Data Structures and Algorithms” and “Object Oriented Analysis & Design”

COURSE DESCRIPTION: Software development life cycle activities- requirements specification using open source Requirement documentation tool, modeling using AgroUML tool; Implementation of various linear and non-linear data structures using C++; Refactoring using InsRefactor and SafeRefactor Eclipse Plugins.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Demonstrate hands-on experience on:
 - Requirements Engineering and Management
 - Estimation of software-size, effort, schedule and cost.
2. Identify key entities and relationships in the problem domain and write succinct textual descriptions of problems, modeling and Implement linear and non-linear data structures using C++.
3. Identify and apply designing, estimating tools and methodologies for complex engineering problems.
4. Apply algorithm design approaches and C++ programming skills to solve real world applications.
5. Work individually and in teams collaboratively in implementing mini projects.
6. Demonstrate communication skills both oral and written for preparing and presenting reports.
7. Engage in life-long learning and enthusiasm to upgrade knowledge and skills in latest technologies and tools.

LIST OF EXERCISES:

1. Prepare the Software Requirement Specification (SRS), High Level Design (HLD) and Detailed Design (DD) for the following experiments
(i) Employee Information System (ii) Online Airline Reservation
Note: For the reference of SRS, HLD and DD templates refer department manual and use any open source Requirement documentation tool.
2. Estimate project parameters such as size, effort and time for development for a Library Information system using Basic COCOMO model.
3. Model UML Use case, Sequence, Collaboration and Component diagrams for the following experiments using Argo UML tool (i) Students Marks Analyzing System (ii) Course Registration System.

4. Study and prepare a report on the following tools: (i) Raptor-Flowchart based programming tool (ii) Microsoft Visio 2010 (iii) Jenkins tool.
5. Write C++ program to implement the following data structures using a singly linked list.
 - a) Stack b) Queue
6. Write C++ program to implement the operations of doubly linked list.
7. Write a C++ program to perform the following operations of BST:
 - a) Node Insertion b) Node Deletion c) Key Search
8. Write C++ program to traverse the given binary tree in Pre-order, In-order and Post-order using recursion.
9. Write C++ program for the implementing BFS and DFS graph traversal techniques using queue and stack data structures.
10. Write C++ program for implementing the following search and sorting techniques.
 - a) Binary search b) Fibonacci Search c) Quick Sort d) Shell Sort
11. Write C++ program to construct the Minimum Cost Spanning Tree using Kruskal's algorithm.
12. Write a C++ program to implement 0/1 Knapsack problem.
13. Mini Project on any web based application using Refactoring

Note: Use InsRefactor and SafeRefactor Eclipse Plugins for refactoring

REFERENCE BOOKS:

1. Roger S. Pressman, "Software Engineering, A practitioner's Approach," 6th Edition, Tata McGraw-Hill, Edition, 2010.
2. Sommerville, "Software Engineering," 8th Edition, Pearson Education, 2007.
3. Rajesh Naik and Swapna Kishore, "Software Requirements and Estimation," Tata McGraw Hill, New Delhi, 2001.
4. Sartaj Sahni "Data structures, Algorithms and Applications in C++," 2nd Edition, Universities press (India) Pvt. Ltd, 2005.
5. Adam Drozdek "Data Structures and Algorithms in C++," 4th Edition, Delmar Cengage Learning, 2012.
6. "Estimation of Project Metrics," <http://vlssit.iitkgp.ernet.in/isad/isad/2/>, drafted on July 01, 2016 at 11:30 AM.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. I Semester
(16MT12532) CLOUD COMPUTING LAB

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	4	2

PREREQUISITES: Courses on "Cloud Computing" and "Operating Systems".

COURSE DESCRIPTION: Hands-on experience on creating Virtual machines on Windows and Linux platforms, Development of Service based web applications & their deployment and Mobile app development.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain hands-on experience on Virtualization models and Cloud Environment.
2. Analyze the given experiment and relate to existing architectures.
3. Apply API development skills in web applications for Cloud deployment.
4. Initiate research to develop novel Service based web solutions.
5. Gain knowledge on investigative approach and identify suitable Cloud platforms for SOA based problems.
6. Devise virtual environments based on virtualization techniques.
7. Develop written and oral communications in preparing and presenting reports

LIST OF EXERCISES:

1. Create Virtual machines with given set of configuration on Hyper-V: "Ubuntu 14 LTS OS, with 2 GB RAM and 200 GB HDD". (IaaS)
2. Create Virtual machines with given set of configuration on Hyper-V: "Windows 7 OS with 4 GB RAM and 500 GB HDD". (IaaS)
3. Create Virtual machines with given set of configuration on Ubuntu OS: "Any Unix OS with 2 GB RAM and 200 GB HDD". (IaaS)
4. Create Virtual machines with given set of configuration on Ubuntu OS: "Windows 7 OS with 4 GB RAM and 500 GB HDD". (IaaS)
5. Develop a simple web application for performing Calculator operations and deploy it on cloud platform. (SaaS)
6. Develop a Design document for a web application, to perform operations based on service calls and to be deployed on cloud environment. (Design Doc)
7. Develop a web application for performing Calculator operations by selecting relevant services. Deploy it on cloud platform. (SaaS)
8. Develop a HTTPS web application with social media interfaces (Facebook / Twitter / Instagram / Google+ APIs). (SaaS)
9. Develop a mobile app on Google App Engine for uploading a resume into a website, collaborated with Drop box. The resume should be encrypted. (PaaS)

10. Develop a service call to run on Drop box resumes for picking the resumes of given skill set. (PaaS)
- i. 6+ years of Exp in Java Development.
 - ii. 10 years of experience in Automation Testing.
 - iii. 15+ years of Managerial experience with technical background.
 - iv. 5-7 years of on-site experience in .NET support and programming.

REFERENCE BOOKS:

1. Ivanka Menken and Ivanka Menken, "*Cloud Computing Virtualization Specialist Complete Certification Kit - Study Guide Book*," 1st Edition, Emereo Pty. Ltd., 2009.
2. Barrie Sosinsky, "*Cloud Computing Bible*," 1st Edition, Wiley India Pvt Ltd, 2011.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT13808) RESEARCH METHODOLOGY
(Common to all M. Tech. Programs)
(Audit Course)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
-	-	-	-	2	-	-

PREREQUISITES: --

COURSE DESCRIPTION: Overview of Research, research problem and design, various research designs, data collection methods, statistical methods for research, importance of research reports and its types.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Acquire in-depth knowledge on
 - Research design and conducting research
 - Various data collection methods
 - Statistical methods in research
 - Report writing techniques.
2. Analyze various research design issues for conducting research in core or allied areas.
3. Formulate solutions for engineering problems by conducting research effectively in the core or allied areas.
4. Carryout literature survey and apply research methodologies for the development of scientific/technological knowledge in one or more domains of engineering.
5. Select and Apply appropriate techniques and tools to complex engineering activities in their respective fields.
6. Write effective research reports.
7. Develop attitude for lifelong learning to do research.
8. Develop professional code of conduct and ethics of research.

DETAILED SYLLABUS

UNIT-I: INTRODUCTION TO RESEARCH METHODOLOGY (Periods: 07)

Objectives and Motivation of Research, Types of Research, Research Approaches, Research Process, Criteria of good Research, Defining and Formulating the Research Problem, Problem Selection, Necessity of Defining the Problem, Techniques involved in Defining a Problem.

UNIT-II: RESEARCH PROBLEM DESIGN AND DATA COLLECTION METHODS

(Periods: 09)

Features of Good Design, Research Design Concepts, Different Research Designs, Different Methods of Data Collection, Data preparation: Processing Operations, Types of Analysis.

UNIT-III: STATISTICS IN RESEARCH**(Periods: 09)**

Review of Statistical Techniques - Mean, Median, Mode, Geometric and Harmonic Mean, Standard Deviation, Measure of Asymmetry, ANOVA, Regression analysis.

UNIT-IV: HYPOTHESIS TESTING**(Periods: 09)**

Normal Distribution, Properties of Normal Distribution, Basic Concepts of Testing of Hypothesis, Hypothesis Testing Procedure, Hypothesis Testing: t-Distribution, Chi-Square Test as a Test of Goodness of Fit.

UNIT-V: INTERPRETATION AND REPORT WRITING**(Periods: 06)**

Interpretation – Techniques and Precautions, Report Writing – Significance, Stages, Layout, Types of reports, Precautions in Writing Reports.

[Total Periods: 40]**TEXT BOOK:**

1. C.R. Kothari, "*Research Methodology: Methods and Techniques*," New Age International Publishers, New Delhi, 2nd Revised Edition, 2004.

REFERENCE BOOKS:

1. Ranjit Kumar, "*Research Methodology: A step-by-step guide for beginners*," Sage South Asia, 3rd ed., 2011.
2. R. Panneerselvam, "*Research Methodology*," PHI learning Pvt. Ltd., 2009.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT22501) BIG DATA TECHNOLOGIES

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Database Management Systems, Data Warehousing and Data Mining"

COURSE DESCRIPTION: Fundamentals of Big Data; Data-parallel programming model-Hadoop, Hadoop I/O; MapReduce features, HDFS; Hive, HBase, Zookeeper; Sqoop and Case studies.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on Big Data storage, processing, querying and reporting.
2. Analyze complex analytical problems to provide optimal solutions.
3. Apply Big Data Technologies to solve real-world problems.
4. Initiate research using HDFS and MapReduce programming model for the implementation of parallelism.
5. Apply various Big Data tools: Sqoop, HBase, MapReduce and Mahout for data analytics.

DETAILED SYLLABUS

UNIT-I: INTRODUCTION TO BIG DATA

(Periods: 11)

The Evolution of Big Data, Characteristics of Big Data, Big Data Sources, The Big Data Revolution, Security, Compliance, Auditing and Protection, Advantages and disadvantages, Challenges of Big Data. **Meet Hadoop:** Data Storage and Analysis, Comparison with Other Systems, Hadoop Ecosystem. **MapReduce:** Analyzing the Data with UNIX Tools, Analyzing the Data with Hadoop, Scaling Out, Hadoop Streaming, Hadoop Pipes. Hadoop Distributed File system.

UNIT-II: HADOOP I/O

(Periods: 11)

Data Integrity, Compression, Serialization, File-Based Data Structures. **Developing a MapReduce Application:** The Configuration API, Configuring the Development Environment, Writing a Unit Test, Running Locally on Test Data, Running on a Cluster, Tuning a Job, MapReduce Workflows. **Working with MapReduce:** Anatomy of a MapReduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution. MapReduce Types and Formats: MapReduce Types, Input Formats, Output Formats.

UNIT-III: MAPREDUCE FEATURES

(Periods: 11)

Counters, Sorting, Joins, Side Data Distribution, MapReduce Library Classes. **Setting Up a Hadoop Cluster:** Cluster Specification, Cluster Setup and Installation, SSH Configuration, Hadoop Configuration, YARN Configuration, Security, Benchmarking a Hadoop Cluster, Hadoop in the Cloud. **Administering Hadoop:** HDFS, Monitoring, Maintenance. Pig: Installing and Running Pig, Comparison with Databases.

UNIT-IV: HIVE, HBase, ZOOKEEPER**(Periods: 11)**

Installing Hive, Running Hive, Comparison with Traditional Databases, HiveQL, Tables, Querying Data, User - Defined Functions. HBase: HBasics, Concepts, Installation, Clients, HBase versus RDBMS, Praxis. ZooKeeper: Installing and Running ZooKeeper, the ZooKeeper Service, Building Applications with ZooKeeper, ZooKeeper in Production.

UNIT-V: SQOOP, MAHOUT and CASE STUDIES**(Periods: 11)**

Getting Sqoop, Generated Code, Database Imports: A Deeper Look, Working with Imported Data, Importing Large Objects, Performing an Export, Exports: A Deeper Look. Mahout: The Three C's of Mahout – Mahout Concepts: Classification, Clustering and Collaborative Filtering (CF). Case Studies: Best Practices for Big Data Analytics, Hadoop Usage at Last.fm, Hadoop and Hive at Facebook, Nutch Search Engine, Log Processing at Rackspace, Cascading, and TeraByte Sort on Apache Hadoop, Using Pig and Wukong to Explore Billion-edge Network Graphs.

[Total Periods: 55]**TEXT BOOKS:**

1. Tom White, "*Hadoop: The Definitive Guide*," O'Reilly and Yahoo Press, 3rd Edition, 2012.
2. Frank J. Ohlhorst, "*Big Data Analytics: Turning Big Data into Big Money*," Wiley Publication, December 2012.

REFERENCE BOOKS:

1. Kevin Roebuck, "*Big Data: High-Impact Strategies - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*," Tebbo Publisher, 2011.
2. Alex Holmes, "*Hadoop in Practice*," Manning Publications Publisher, 2012.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT22502) SERVICE ORIENTED ARCHITECTURE

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Software Engineering" and "Web Technologies"

COURSE DESCRIPTION: Introduction to SOA, Web services & Primitive SOA, Contemporary SOA, Principles of SOA, Service Layers, Delivery strategies, Service Modeling, Service and Business process design- Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL), and Web Services- Business Process Execution Language (WS-BPEL), SOA support in .NET and J2EE.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on:
 - Fundamentals of web services
 - Development strategies, Principles, Services, Layers and characteristics of service orientation.
2. Analyze complex business process critically in identifying appropriate service model logic.
3. Solve real time problems related to design the Web Services using XML Schema, WSDL, SOAP and BPEL.
4. Initiate research using XML Schema, WSDL, SOAP, BPEL and Service Oriented Enterprise model.
5. Apply the modern tools and techniques of .NET and J2EE to modeling the web services.

DETAILED SYLLABUS

UNIT-I: SOA AND WEB SERVICES FUNDAMENTALS (Periods: 10)

Introducing SOA: Fundamental SOA, Common Characteristics of Contemporary SOA, Common tangible benefits of SOA, Common pitfalls of adopting SOA.

Web Services and Primitive SOA: The Web Services frame work, Services, Service descriptions, Messaging.

UNIT-II: SOA AND WS-* EXTENSIONS (Periods: 11)

Web Services and Contemporary SOA (Part-I Activity Management and Composition): Message Exchange Patterns, Service Activity, Coordination, Atomic transactions, Business Activities, Orchestration, Choreography.

Web Services and Contemporary SOA (Part-II Advanced Messaging, Metadata and Security): Addressing, Reliable messaging, Correlation, Policies, Metadata exchange, Security, Notification and eventing.

UNIT-III: PRINCIPLES, SERVICE LAYERS AND PLANNING (Periods: 11)

Principles of Service-Oriented: Anatomy of SOA, Common principles of Service Orientation, Service Orientation and Object Orientation.

Service Layers: Service-Oriented and Contemporary SOA, Service Layer Abstraction, Application Service Layer, Business Service Layer, Orchestration Service Layer, Agnostic Services, Service Layer Configuration Scenarios.

SOA Delivery Strategies: SOA delivery lifecycle phases, The Top-down strategy, The bottom-up strategy, The agile strategy.

UNIT-IV: BUILDING SOA (ANALYSIS & DESIGN) (Periods: 12)

Analysis Introduction: Introduction to Service Oriented Analysis, Benefits of a Business Centric SOA, Deriving Business Services.

Service Modeling: Service Modeling, Service Modeling guidelines, Classifying Service model logic, Contrasting Service modeling approaches.

Design Introduction: Introduction to Service-Oriented design, WSDL related XML Schema language basics, WSDL language basics, SOAP language basics, Service interface design tools.

UNIT-V: BUILDING SOA (DESIGN & TECHNOLOGY) (Periods: 11)

Service Design: Service Design overview, Entity-centric business Service Design, Application Service Design, Task-centric business Service Design, Service Design guidelines.

Business Process Design: WS-BPEL language basics, Service Oriented Business process Design.

SOA Platforms: SOA platform basics, SOA Support in J2EE, SOA Support in .NET

[Total Periods: 55]

TEXT BOOK:

1. Thomas Erl, "*Service-Oriented Architecture - Concepts, Technology and Design*," Pearson, 2008.

REFERENCE BOOKS:

1. Shankar Kambhampaty, "*Service Oriented Architecture for Enterprise and Cloud Applications*," Wiley- India, 2012
2. Eric Newcomer and Greg Lomow, "*Understanding SOA with Web Services*," Pearson Education, 2007.
3. M. Rosen et al., "*Applied SOA*," Wiley India Pvt. Ltd, 2009.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT22503) SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Software Development Methodologies," and "Object Oriented Programming".

COURSE DESCRIPTION: Envisioning Architecture - Software Architecture, Pattern System; Creating Architecture – Understanding the Requirements, Designing the Architecture, Documenting Software Architectures, Reconstructing Software Architectures; Analyzing Architectures and moving from one system to many – Evaluating the Architecture; Introduction to Design Patterns and Creational Patterns; Structural and behavioral patterns.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on:
 - Software architecture styles and business life cycle.
 - Various design issues and patterns.
2. Analyze and identify architectural styles and patterns to solve software design problems.
3. Solve Software Architecture design problems using design patterns.
4. Apply appropriate novel software pattern to solve real world problems in object oriented software design process

DETAILED SYLLABUS

UNIT I: ENVISIONING ARCHITECTURE

(Periods: 12)

Introduction to Software Architecture, Software Processes and the ABC, Architectural Patterns, Reference Model and Reference Architecture.

Pattern System – Introduction to Pattern System, Pattern Classification, Pattern Selection, Introduction to Architectural Patterns, Pipes & Filter, Model-View-Controller.

UNIT II: CREATING ARCHITECTURE

(Periods: 14)

Understanding the Requirements: Functionality and Architecture, Architecture and Quality Attributes (QA), System QAs, QA Scenarios in Practice, Business and Architecture Qualities, Achieving Qualities.

Designing the Architecture: Architecture in the Life Cycle, Attribute Driven Design (ADD).

Documenting Software Architectures: Uses of Architectural Documentation, Views.

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

UNIT III: ANALYZING ARCHITECTURES AND MOVING FROM ONE SYSTEM TO MANY

(Periods: 10)

Evaluating the Architecture: The ATAM, The CBAM. The World Wide Web-A Case Study in Interoperability

Moving From one System to Many: Software Product Lines, Celsius Tech- A Case Study in Product Line Development, Building Systems from off the shelf components, Software Architecture in the future.

UNIT IV: INTRODUCTION TO DESIGN PATTERN AND CREATIONAL PATTERNS

(Periods: 08)

Introduction to Design Patterns: Design Patterns in Smalltalk MVC, Describing DPs, The Catalog of DPs, Organizing the Catalog, Design Pattern to Solve Design Problem, Select and Use of a DP.

Creational Patterns: Abstract Factory, Builder, Factory Method, Prototype.

UNIT V: STRUCTURAL AND BEHAVIORAL PATTERNS

(Periods: 10)

Structural Patterns: Adapter, Composite, Decorator, Flyweight.

Behavioral patterns: Command, Iterator, Mediator, Observer, State.

[Total Periods: 54]

TEXT BOOKS:

1. Len Bass, Paul Clements and Rick Kazman, "*Software Architecture in Practice*," 2nd Edition, Addison-Wesley, 2003.
2. Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides "*Design Patterns: Elements of Reusable Object-Oriented Software*," Pearson Education, 1995.

REFERENCE BOOKS:

1. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad and Michael Stal, "*Pattern-Oriented Software Architecture: A System of Pattern*," Volume 1, John Wiley & Sons, 2001.
2. Eric Freeman and Elisabeth Freeman, "*Head First Design patterns*," O'REILLY, 2004.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. II Semester
(16MT22504) SOFTWARE TESTING TECHNIQUES

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A course on "Software Development Methodologies".

COURSE DESCRIPTION: Basic concepts of Software Testing; Testing Techniques – Levels of Testing; Testing Process – Test Planning; Test Metrics and Reports; Software Test Automation.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on the Software Testing types and Process for different programming environments.
2. Analyze and apply the appropriate testing techniques suitable for testing the software.
3. Design and apply appropriate strategies for selecting test cases to meet requirements of the product.
4. Apply efficient modern software testing tools for automation.
5. Write test cases and perform defect reporting.

DETAILED SYLLABUS

UNIT-I: BASIC CONCEPTS OF SOFTWARE TESTING (Periods: 12)

Fundamentals of software testing - software verification and validation – V test model: V model for software, testing during proposal stage, testing during requirements stage, testing during test-planning phase, test during design phase, VV model, critical roles and responsibilities.

UNIT-II: TESTING TECHNIQUES (Periods: 12)

Levels of testing – Acceptance testing – feature based testing (special tests part – I) – Application based testing (special tests part – II)

UNIT-III: TESTING PROCESS (Periods: 10)

Test planning –test policy, contents, strategy, test plan, Quality plan, test plan template, guidelines, test administration and estimation, standards, building test data, test cases, scenarios, templates for test cases, test scripts, effective test cases, building test data, generation of test data, roles and responsibilities in testing life cycle, test process monitoring.

UNIT-IV: TEST METRICS AND REPORTS**(Periods: 10)**

Testing related data, defect data, efficiency data, categories of test metrics, estimated, budgeted, approved and actual, resources, effectiveness in testing, defect density, defect leakage ratio, residual defect density, test team efficiency, test case efficiency, rework, MTBF/ MTTR, test reports, status reports, integration test reports, system test reports, final test reporting, test status report, Bench marking

UNIT-V: SOFTWARE TEST AUTOMATION**(Periods: 10)**

Test Automation: Scope of Automation, Design and Architecture of automation, Process Model for Automation, challenges in automation. Load Runner, Selenium, QTP, RFT and RQM.

[Total Periods: 54]**TEXT BOOKS:**

1. M. G. Limaye, "*Software Testing: Principles and Techniques and Tools*," Tata McGraw – Hill Education, 1st Edition, 2012.
2. Srinivasan Desikan, Gopaldaswamy Ramesh, "*Software Testing: Principles and Practices*", Pearson 2012

REFERENCE BOOKS:

1. Dr. K. V. K. K. Prasad, "*Software Testing Tools*," Dreamtech, 1st Edition, 2004.
2. Aditya P. Mathur, "*Foundations of Software Testing*", Pearson, 2008

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. II Semester
(16MT22505) WEB TECHNOLOGIES
(Common to M.Tech. (SE and CS))

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A course on "Object Oriented Programming".

COURSE DESCRIPTION: Web Technologies: HTML5, CSS, JavaScript, JQuery; Open source server-side scripting language- PHP; MySQL database concepts; and AJAX.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on web technologies.
2. Analyze the functionality of client as well as server side web technologies for validating web pages.
3. Gain programming skills to design and develop novel web applications
4. Apply web technologies to make web pages more interactive, scalable and user friendly web applications.

DETAILED SYLLABUS

UNIT-I: HTML5 AND CSS3

(Periods: 14)

HTML5: Overview of HTML and XHTML, HTML5 - Introduction, HTML5 Document Structure, Creating Editable Content, Checking Spelling Mistakes, Exploring Custom Data Attributes, Microdata, Client-Side Storage, Drag and Drop Feature, ARIA Accessibility, Offline Web Applications, Web Communications, Cross-Document Messaging and Desktop Notifications, 2D and 3D Graphics. **CSS3:** Introduction, Features of CSS3, Syntax of CSS, Exploring CSS selectors, Inserting CSS in HTML Document, State of CSS3.

UNIT-II: JAVASCRIPT AND JQUERY

(Periods: 10)

JavaScript: Overview of JavaScript, JavaScript Functions, Events, Image Maps and Animations, JavaScript Objects. **JQuery:** Fundamentals of JQuery, JQuery Selectors, JQuery Methods to Access HTML Attributes and Traversing, JQuery Manipulators, Events and Effects.

UNIT-III: INTRODUCTION TO PHP

(Periods: 10)

Introduction, Data Types, Variables, Constants, Expressions, String Interpolation, Control Structures, Functions, Arrays, Embedding PHP Code in Web Pages, Object Oriented PHP.

UNIT-IV: PHP AND MYSQL

(Periods: 10)

PHP and Web Forms, Sending Form Data to a Server, Authenticating Users with PHP, Session Handlers, PHP with MySQL, Interacting with the Database, Database Transactions.

UNIT-V: AJAX

(Periods: 08)

Exploring Different Web Technologies, Exploring AJAX, Creating a Sample AJAX Application, Displaying Date and Time using AJAX, Creating the XMLHttpRequest Object, Reading a File Synchronously and Asynchronously, Reading Response Headers, Loading List Boxes Dynamically using XMLHttpRequest Object, JQuery with AJAX, Validating a Field using AJAX and PHP.

[Total Periods: 52]

TEXT BOOKS:

1. Kogent Learning Solutions Inc, "*HTML 5 Black Book: Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP and JQuery*," 1st Edition, Dreamtech Press, 2011.
2. W. Jason Gilmore, "*Beginning PHP and MySQL*," 4th Edition, APress, 2011.

REFERENCE BOOKS:

1. Andrea Tarr, "*PHP and MySQL*," 1st Edition, Willy India, 2012.
2. Thomas A. Powell, "*The Complete Reference: HTML and CSS*," 5th Edition, Tata McGraw Hill, 2010.
3. Steve Suehring, Tim Converse and Joyce Park, "*PHP6 and MySQL*," 1st Edition, Willy India, 2009.
4. P. J. Deitel and H. M. Deitel, "*Internet & World Wide Web How to Program*," 4th Edition, Pearson, 2009.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT10502) ADVANCED DATABASE MANAGEMENT
SYSTEMS
(PROFESSIONAL ELECTIVE-II)

Int.	Ext.	Total	L	T	P	C
Marks	Marks	Marks				
40	60	100	4	-	-	4

PREREQUISITES: A Course on "Database Management Systems".

COURSE DESCRIPTION: Concepts of Database System Concepts and Architectures, Data modeling using ER-Model;SQL, Objects Relational Database and XML, Database Design and File Organizations, Query Processing, Concurrency and Recovery, Distributed DBMS Architecture and Design.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain Advanced knowledge in:
 - Database System Concepts , Languages , Interfaces and Architectures
 - Query Languages , Relational Databases and XML
 - Database Design and File Organization.
 - Query Processing and Recovery
 - Distributed Database Architecture and Design
2. Analyze database management architecture and categorize languages and database objects.
3. Design a wide range of potential solutions for the database problems using ER-diagrams, SQL, Normalization and XML.
4. Apply higher order skill and contribute for the development of technical knowledge to solve the problems innovatively.
5. Apply appropriate modern techniques, resources and tools for the real world problems in databases.

DETAILED SYLLABUS:

UNIT-I: DATABASE SYSTEM CONCEPTS AND ARCHITECTURES, DATA MODELING USING ER-MODEL (Periods: 11)

Database System Concepts and Architectures: Architecture And Data Independence, Database Languages and Interfaces, Database System Environment, Centralized and Client/server Architectures for DBMS.

Data modeling using ER-Model: Using High-Level Conceptual data Model for Database Design, A Sample Database Application, Entity Types, Entity Sets, Attributes, and Keys, Relationship Types, Relationship Sets, Roles, and Structural Constraints, Weak Entity Types, Refining the ER Design for the COMPANY Database, ER Diagrams, Naming Conventions, and Design Issues, Relationship Types of Degree Higher than Two, Relational Database Design using ER-to-Relational Model.

UNIT-II: SQL, OBJECTS RELATIONAL DATABASE AND XML (Periods: 12)

SQL: Schema Definition, Constraints, Queries, Joins, Assertions, Triggers and Views

Object Relational Databases: Concepts for Object Databases, Standards, Languages and Design.

XML: Hierarchical data model, Documents, DTD, XML Schema, Documents and Databases, Querying.

UNIT-III: DATABASE DESIGN AND FILE ORGANIZATIONS (Periods: 11)

Database Design: Functional Dependencies, Types of Normal Forms, properties of relational decompositions, Algorithms for Relational Database Design.

File Organizations: Introduction, Secondary Storage Devices, Buffering of Blocks, Placing File Records on Disk, Operations on Files, Files of Unordered Records, Files of Ordered Records, Hashing Techniques, Other Primary File Organizations, Parallelizing Disk Access Using RAID Technology, New Storage Systems.

UNIT-IV: QUERY PROCESSING, CONCURRENCY AND RECOVERY (Periods: 13)

Query Processing: Problem, Objectives, Characterization, Layers and Query Optimization and Query Optimization Algorithms: INGRES, System R, Distributed INGRES, R*, SDD-1.

Concurrency Control: Transaction management types and properties, Algorithms, Deadlock Management.

Recovery: Concepts, Techniques Based on Deferred Update and Immediate Update, Shadow paging and ARIES Algorithm.

UNIT-V: DISTRIBUTED DBMS ARCHITECTURE AND DESIGN (Periods: 09)

Distributed DBMS Architecture: Architectural Models and Architectures.

Distributed Database Design: Alternative Design Strategies, Distribution Design Issues, Fragmentation and Allocation.

[Total Periods: 56]

TEXT BOOKS:

1. Ramez Elmasri & Shamkant B. Navathe, "*Database Systems: Models, Languages, Design and Application Programming*," 6th Edition, New Delhi, Pearson Education, 2013.
2. M. Tamer Ozsu, Patrick Valduriez, "*Principles of Distributed Database System*," 2nd Edition, New Delhi, Pearson Education, 2006.

REFERENCE BOOKS:

1. Abraham Silberchatz, Henry F. Korth, S. Sudarsan, "*Database System Concepts*," 5th Edition, N.Y, McGraw-Hill, 2006.
2. Thomas M. Connolly, Carolyn E. Begg, "*Database Systems – A Practical Approach to Design, Implementation and Management*," 3rd Edition, New Delhi, Pearson Education, 2003.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT22506) SOFTWARE PROCESS AND PROJECT MANAGEMENT
(PROFESSIONAL ELECTIVE-II)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A course on "Software Engineering"

COURSE DESCRIPTION: Software Process Maturity Levels-Initial and the repeatable process, The Defined Process, The Managed Process and the Optimizing Process; Software management Renaissance, Software Management Disciplines and Framework, Next Generation Software Economics and Case Studies.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on:
 - Software process maturity levels and activities
 - Software project management strategies
2. Perform analysis of project management techniques for deploying software systems quickly.
3. Apply the skills to solve problems in modern software process and project management.
4. Conduct research to improve software process and project Controlling activities.
5. Apply software process management techniques to measure the quality of the software.
6. Learn how to optimize software project cost and schedule estimation techniques

DETAILED SYLLABUS:

UNIT-I: SOFTWARE PROCESS MATURITY AND THE REPEATABLE PROCESS

(Periods: 12)

A Software Maturity Framework, The Principles of Software Process Change, Software Process Assessment, The Initial Process, The Project Plan, Software Configuration Management-I.

UNIT-II: THE DEFINED PROCESS, THE MANAGED PROCESS AND THE OPTIMIZING PROCESS

(Periods: 14)

Software Standards, Software Configuration Management-II, Defining the Software Process, Data Gathering and Analysis, Managing Software Quality, Defect Prevention, Automating the software Process.

UNIT-III: SOFTWARE MANAGEMENT RENAISSANCE AND PROCESS FRAMEWORK

(Periods: 10)

Conventional Software Management, Evolution of Software Economics, Improving software Economics, Life cycle phases, Workflows of the process, Checkpoints of the Process.

UNIT-IV: SOFTWARE MANAGEMENT DISCIPLINES**(Periods: 08)**

Iterative process planning, Project organizations and responsibilities, Process Automation, Project control and Process Instrumentation, Tailoring the Process.

UNIT-V: NEXT GENERATION SOFTWARE ECONOMICS AND CASE STUDIES**(Periods: 08)**

Modern Project Profiles, Next Generation Software Economics, Modern Process Transitions.

Case studies: CCPDS-R and Process Improvement and Mapping to the CMM.

[Total Periods: 52]**TEXT BOOKS:**

1. Watts S. Humphrey, "*Managing the Software Process*," Pearson Education, 2009
2. Walker Royce, "*Software Project Management*," Pearson Education, 2005.

REFERENCE BOOKS:

1. Bob Hughes and Mike Cotterel, "*Software Project Management*," Tata McGraw- Hill, 1st Edition, 2006.
2. Joel Henry, "*Software Project Management*," Pearson Education, 2003.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT22507) SOFTWARE REVERSE ENGINEERING
(PROFESSIONAL ELECTIVE-II)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Software Engineering," and "Object Oriented Concepts".

COURSE DESCRIPTION: Foundations – Software Reverse Engineering, Applications and Tools, Low level software; Reverse Engineering Tools and applied Reversing; Object Flow Graph, Class and Object diagrams; Interaction, State and Package diagrams; Reversing Malware and Anti-reversing Techniques.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on:
 - Finding malicious code.
 - Discovering unexpected flaws and faults.
 - Finding the use of others code.
 - Learning from others products of a different domain or purpose.
2. Analyze Software System and discover new opportunities for improving existing system.
3. Apply Anti-reversing techniques for Code obfuscation.
4. Design and develop Object Flow Graph, UML diagrams necessary for code reversing.
5. Apply Disassemblers, Debuggers and Decompilations tools and algorithms to implementing Reverse Engineering.
6. Exhibit ethical attitude in software reverse engineering.

DETAILED SYLLABUS

UNIT-I: FOUNDATIONS

(Periods: 10)

Need for Reverse Engineering, Software Reverse Engineering, Reverse Applications, Low Level Software, The Reversing Process, The Tools, Is Reversing Legal.

Low Level Software: High Level Perspectives, Low Level Perspectives, Assembly Language, A Primer on Compilers and Compilation, Execution Environments.

UNIT-II: REVERSE ENGINEERING TOOLS AND APPLIED REVERSING **(Periods: 12)**

Reverse Engineering Tools: Different Reversing Approaches, Disassemblers, Debuggers, Decompilers, System-Monitoring Tools, Patching Tools, Miscellaneous Reversing Tools.

Beyond the Documentation: Reversing and Interoperability, Laying the Ground Rules, Locating Undocumented APIs, Case Study.

UNIT-III: OBJECT FLOW GRAPH, CLASS AND OBJECT DIAGRAMS **(Periods: 10)**

Object Flow Graph: Abstract Language, Object Flow Graph, Containers, Flow Propagation Algorithm, Object Sensitivity, The elib Program.

Class Diagram: Class Diagram Recovery, Declared Vs Actual Types, Containers, The elib Program.

Object Diagram: The Object Diagram, Object Sensitivity, Dynamic Analysis, The elib Program.

UNIT-IV: INTERACTION, STATE AND PACKAGE DIAGRAMS (Periods: 13)

Interaction Diagram: Interaction Diagram, Interaction Diagram, Interaction Diagram Recovery, Dynamic Analysis, The elib Program.

State Diagram: State Diagram, Abstract Interpretation, State Diagram Recovery, The elib Program.

Package Diagram: Package Diagram Recovery, Clustering, Concept Analysis, The elib Program, Tool Architecture, The elib Program, Perspectives.

UNIT-V: REVERSING MALWARE AND ANTI-REVERSING TECHNIQUES (Periods: 10)

Reversing Malware: Types of malware, Sticky software, Future malware, Uses of malware, Malware vulnerability, Polymorphism, Metamorphism, establishing a secure environment.

Anti Reversing Techniques : Anti reversing?, Basic approaches to anti reversing, Eliminating symbolic information, Code encryption, Active anti debugger techniques, Confusing Disassemblers, Code obfuscation, Control flow transformations, Data transformations.

[Total Periods: 55]

TEXT BOOKS:

1. Paolo Tonella & Alessandra Potrich, *"Reverse Engineering of Object Oriented Code,"* Springer-2005.
2. Eldad Eilam, *"Reversing: Secrets of Reverse Engineering,"* Wiley, 2005.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. II Semester
(16MT22508) SOFTWARE SECURITY
(PROFESSIONAL ELECTIVE-II)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Software Engineering," and "Network Security".

COURSE DESCRIPTION: Importance of Security in Software - Security a Software Issue, Secure Software; Requirements Engineering for Secure Software; Security Principles in SDLC - Secure Software Architecture and Design, Secure Coding and Testing; Security and Complexity - System Assembly Challenges; Governance and Managing for more Secure Software.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on security issues in:
 - Requirement Engineering
 - Architecture and Design
 - Coding and Testing
2. Analyze complex software projects to describe security risks and mitigation techniques.
3. Applying methods to detect software security defects, SQUARE process model for requirement gathering and coding practices & security testing for identifying security failures.
4. Initiate research issues in code analysis techniques to improve software security.

DETAILED SYLLABUS

UNIT-I: IMPORTANCE OF SECURITY IN SOFTWARE (Periods: 11)

Security a Software Issue: Introduction, The problem, Software Assurance and Software Security, Threats to software security, Sources of software insecurity, Benefits of detecting software security defects early, managing secure software development.

Secure Software: Introduction, Properties of Secure Software, Influencing the security properties of software, Asserting and specifying the desired security properties.

UNIT-II: REQUIREMENTS ENGINEERING (Periods: 10)

Requirements Engineering for Secure Software: Introduction, Misuse and abuse cases, the SQUARE process Model, SQUARE sample outputs, Requirements elicitation, Requirements prioritization.

UNIT-III: SECURITY PRINCIPLES IN SDLC (Periods: 11)

Secure Software Architecture and Design: Introduction, Software Security practices for Architecture and Design - architectural risk analysis, Software security knowledge for Architecture and Design - Security principles, Security guidelines and Attack patterns.

Secure Coding and Testing: Introduction, Code analysis, Coding Practices, Software Security testing, Security testing considerations throughout the SDLC.

UNIT-IV: SECURITY AND COMPLEXITY**(Periods: 10)**

System Assembly Challenges: Introduction, Security failures, functional and attacker perspectives for security analysis in web services and identity management, system complexity drivers and security, Deep technical problem complexity.

UNIT-V: GOVERNANCE AND MANAGING**(Periods: 10)**

Governance and Managing for more Secure Software: Introduction, Governance and security, adopting an enterprise software security framework, Defining adequate security, Risk Management framework for software security, Security and Project Management, Maturity of Practice.

[Total Periods: 52]**TEXTBOOK:**

1. Julia H. Allen, Sean Barnum, Robert J. Ellison, Gary McGraw, and Nancy R. Mead, *"Security Engineering: A Guide for Project Managers,"* Pearson Education, 2009.

REFERENCE BOOKS:

1. Gary McGraw, *"Software Security: Building Security In,"* Addison-Wesley, 2006.
2. Mark Dowd, John McDonald and Justin Schuh, *"The Art of Software Security Assessment: Identifying and Preventing Software Vulnerabilities,"* 1st Edition, Addison-Wesley, 2006.
3. John Viega and Gary McGraw, *"Building Secure Software: How to Avoid Security Problems the Right Way,"* Addison-Wesley, 2001.
4. G. Hoglund and G. McGraw, *"Exploiting Software: How to Break Code,"* Addison-Wesley, 2004.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT22531) ADVANCED SOFTWARE ENGINEERING LAB-2

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	4	2

PREREQUISITES: Courses on “Software Development Methodology” and “Object Oriented Programming”

COURSE DESCRIPTION: Software development life cycle activities- Implementation of design patterns using enterprise architect; Creation of web service client; Implementation of Orchestration with BPEL; Test plan document; Regression testing, functional testing using QTP, RFT and Selenium; Performance testing using Load Runner, RPT and Web Performance Tool.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain hands-on experience on:
 - Web services
 - Design issues and patterns
2. Analyze software design problems, complex business process and identify architectural styles, patterns and appropriate service model logic to solve.
3. Gain software testing skills and practical experience by conducting software testing processes.
4. Apply various testing phases and automate testing process for the given application using Software Engineering concepts and practices to:
 - i. Identify customer’s needs.
 - ii. Evaluate system for feasibility.
 - iii. Perform economic and technical analysis.
 - iv. Allocate functions to system elements.
 - v. Establish schedule, constraints and estimate cost.
 - vi. Create system definitions.
5. Apply QTP and RFT tools for automation testing of software process.
6. Work individually and in teams collaboratively in implementing mini projects.
7. Gain communication skills both oral and written for preparing and presenting reports.

LIST OF EXERCISES:

1. Study and prepare a report on the Enterprise Architect 12.0.
2. Write the Java program for Abstract Factory design pattern.
3. Write the Java program for Decorator design pattern.
4. Creation of web services and client applications to consume the following services:
(i) Addition operation (ii) Finding factorial number
5. Implementation of orchestration with BPEL for authenticating user credentials.
6. Create a test plan document for a Desktop based application.
7. Write the Functional test cases for a Desktop based application.
8. Conduct Functional testing for a Desktop based application using QTP.
9. Conduct Functional testing for a Desktop based application using RFT.
10. Conduct Performance testing for a Desktop based application using Load Runner.
11. Write the Regression test cases for a Web based application.
12. Conduct Functional Testing using Selenium for a Web based application.
13. Develop a mini project for any web based application.

Note: Use above tools in the process of developing the project

REFERENCE BOOKS:

1. James W.Cooper, "Java Design Patterns- A Tutorial," Pearson Education, 2000
2. Eric Newcomer and Greg Lomow, "Understanding SOA with Web Services," Pearson Education, 2007.
3. Ilene Burnstein, "Practical Software Testing," Springer-Verilog Internation Edition, 2003.
4. Dr. K. V. K. K.Prasad, "Software Testing Tools," Dreamtech, 1st Edition, 2004.
5. "Introduction to IBM Rational Functional Tester 6.1,"
<http://www.ibm.com/developerworks/rational/library/04/r-3228/3228.html>, drafted on July 01, 2016 at 2:30 PM.
6. "Selenium Documentation," <http://docs.seleniumhq.org/docs/>, drafted on July 10, 2016 at 3:30 PM.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT22532) BIG DATA TECHNOLOGIES LAB

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	4	2

PREREQUISITES: Courses on "Database Management Systems", "Data Mining and Data Warehousing", and "Big Data Technologies".

COURSE DESCRIPTION: Hands on Java Programs; Data-parallel programming model-Hadoop, Hadoop I/O; MapReduce features, HDFS; Hive, HBase, Zookeeper; Sqoop and Case studies.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain hands-on experience on:
 - Map-Reduce, Sqoop, HBase and Mahout
2. Apply Big Data Technologies to solve contemporary problems
3. Gain skills on HDFS and MapReduce programming model concepts.
4. Initiate research insights into latest technologies of Big Data Analytics.
5. Apply various Big Data tools: Sqoop, HBase, MapReduce and Mahout.
6. Work individually and in teams collaboratively in implementing mini projects.
7. Demonstrate communication skills both oral and written for preparing and presenting reports.

LIST OF EXERCISES:

1. Practice Hadoop installation, Hadoop Cluster Configuration, Name node Safe mode, Secondary Name node, Hadoop File system Shell, Java API ,Configuration, Reading Data and Writing Data in Hadoop Distributed File System.
2. Practice on MapReduce to Implement first MapReduce Job, Running MapReduce Locally, Running MapReduce on Cluster, Packaging MapReduce Jobs, MapReduce CLASSPATH, Submitting Jobs, Logs and Web UI, Input and Output Formats, Implement a Streaming Job, Contrast with Java Code and Create counts in Streaming application.
3. Practice on Pig Programming with Execution Modes, Installation, Pig Latin Basics, Developing Pig Script: Most Occurred Start Letter, Resources, Joining data-sets and User Defined Functions (UDF).
4. Practice on Hive Installation, Table Creation and Deletion, Loading Data into Hive, Partitioning, Bucketing and Joins.
5. Practice on Sqoop with Importing and Exporting data from using RDBMS.
6. Practice on HBase Management Console, HBase Shell: Define Schema and Create, Read, Update and Delete, create via Put method, Read via Get method, Update via Put method, Delete via Delete method, Create Table, Drop Table, Scan API, Scan Caching, Scan Batching and Filters.

7. Practice on Oozie Installation, Write Oozie Workflow, Deploy and Run Oozie Workflow.
8. Practice on Flume properties, Flume sinks and Flume Channels
9. Practice on Zookeeper with Stand alone operation, Managing zookeeper storage and Programming to Zookeeper
10. **Case Study 1: Insurance Domain:** A US-based insurance provider has decided to launch a new medical insurance program targeting various customers. To help this customer understand the current realities and the market better, perform a series of data analytics tasks using Hadoop. The customer has provided pointers to the data set that can be used. For the Insurance company data set, perform the following analysis tasks
 - i) Find maximum insurance
 - ii) Find minimum insurance
 - iii) Find average insurance
 - iv) Find Total insurance
11. **Case Study 2: Retail Domain:** An Indian-based online retailer wants to launch a new product category and wants to understand the potential growth areas and areas that have stagnated over a period of time. It wants to use this information to ensure its product focus is aligned to opportunities that will grow over the next 5–7 years. The customer has also provided pointers to the data set that can be used to:
 - i) Find maximum retail
 - ii) Find minimum retail
 - iii) Find average retail
 - iv) Find Total retail
12. **Case Study 3: Education Domain:** The company has recently bagged a large assignment from a US-based customer that is into training and development. The larger outcome deals with launching a suite of educational and skill development programs to consumers across the globe. As part of the project, the customer wants the company to analyze a series of data sets to arrive at a prudent product mix, product positioning, and marketing strategy that will be applicable for at least a decade.
 - i) Find maximum students
 - ii) Find minimum students
 - iii) Find average students
 - iv) Find Total students

REFERENCE BOOKS:

1. Tom White, "*Hadoop: The Definitive Guide*," O'Reilly and Yahoo Press, 3rd Edition, 2012.
2. Frank J. Ohlhorst, "*Big Data Analytics: Turning Big Data into Big Money*," Wiley Publication, December 2012.
3. Kevin Roebuck, "*Big Data: High-Impact Strategies - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*," Tebbo Publisher, 2011.
4. Alex Holmes, "*Hadoop in Practice*," Manning Publications Publisher, 2012.

M. Tech. (SE) – II Semester (16MT22533) SEMINAR

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
--	100	100	--	--	--	2

PRE-REQUISITES: -Nil-

COURSE DESCRIPTION: Identification of seminar topic; literature survey; preparation of technical report and presentation.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Demonstrate capacity to identify an advanced topic for seminar in core and allied areas.
2. Extract information pertinent to the topic through literature survey.
3. Comprehend extracted information through analysis and synthesis critically on the topic.
4. Plan, organize, prepare and present effective written and oral technical report on the topic.
5. Adapt to independent and reflective learning for sustainable professional growth in Software Engineering.
6. Contribute to multidisciplinary scientific work in the field of Software Engineering.
7. Understand ethical responsibility towards environment and society in the field of Software Engineering.
8. Engage in lifelong learning for development of technical competence in the field of Software Engineering.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT23810) INTELLECTUAL PROPERTY RIGHTS
(Common to all M. Tech. Programs)
(Audit Course)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
-	-	-	-	2	-	-

PREREQUISITES: --

COURSE DESCRIPTION: Introduction to Intellectual Property; Trade Marks; Law of Copy Rights; Law of Patents; Trade Secrets; Unfair Competition; New Development of Intellectual Property.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Demonstrate in-depth knowledge on
 - Intellectual Property
 - Trade Marks & Secrets
 - Law of Copy Rights, Patents
 - New development of Intellectual Property
2. Analyze the different forms of infringement of intellectual property rights.
3. Solve problems pertaining to Intellectual Property Rights.
4. Stimulate research zeal for patenting of an idea or product.
5. Write effective reports required for filing patents.
6. Develop life-long learning capabilities.
7. Develop awareness of the relevance and impact of IP Law on their academic and professional lives.
8. Develop attitude for reflective learning.

DETAILED SYLLABUS

UNIT - I: INTRODUCTION TO INTELLECTUAL PROPERTY

Introduction, types of intellectual property, international organizations, agencies and treaties, importance of intellectual property rights.

UNIT - II: TRADE MARKS

Purpose and function of trademarks, acquisition of trade mark rights, protectable matter, selecting and evaluating trade mark, trade mark registration processes.

UNIT - III: LAW OF COPY RIGHTS

Fundamental of copy right law, originality of material, rights of reproduction, rights to perform the work publicly, copy right ownership issues, copy right registration, notice of copy right, international copy right law.

Law of patents: Foundation of patent law, patent searching process, ownership rights and transfer

UNIT - IV: TRADE SECRETS

Trade secret law, determination of trade secret status, liability for misappropriations of trade secrets, protection for submission, trade secret litigation.

Unfair competition: Misappropriation right of publicity, False advertising.

UNIT - V: NEW DEVELOPMENT OF INTELLECTUAL PROPERTY

New developments in trade mark law; copy right law, patent law, intellectual property audits.

International overview on intellectual property, international - trade mark law, copy right law, international patent law, international development in trade secrets law.

REFERENCE BOOKS:

1. Deborah, E. Bouchoux, "*Intellectual property right*", Cengage learning.
2. Prabuddha ganguli, "*Intellectual property right - Unleashing the knowledge economy*", Tata Mc Graw Hill Publishing Company Ltd.

**M. Tech. (SE) – III & IV Semesters
(16MT32531 & 16MT42531) PROJECT WORK**

Int. Marks 200	Ext. Marks 200	Total Marks 400	L	T	P	C
			--	--	--	28

PRE-REQUISITES: -Nil-

COURSE DESCRIPTION: Identification of topic for the project work; Literature survey; Collection of preliminary data; Identification of implementation tools and methodologies; Performing critical study and analysis of the topic identified; Time and cost analysis; Implementation of the project work; Writing of thesis and presentation.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Demonstrate capacity to identify an advanced topic for project work in core and allied areas.
2. Extract information pertinent to the topic through literature survey.
3. Comprehend extracted information through analysis and synthesis critically on the topic.
4. Solve engineering problems pertinent to the chosen topic for feasible solutions.
5. Use the techniques, skills and modern engineering tools necessary for project work.
6. Do time and cost analysis on the project.
7. Plan, prepare and present effective written and oral technical report on the topic.
8. Adapt to independent and reflective learning for sustainable professional growth.
9. Contribute to multidisciplinary scientific work in the field of Software Engineering.
10. Understand ethical responsibility towards environment and society in the field of Software Engineering.
11. Engage lifelong learning for development of technical competence in the field of Software Engineering.