KAKATIYA INSTITUTE OF TECHNOLOGY & SCIENCE

(An Autonomous Institute Under Kakatiya University, Warangal)

WARANGAL – 506015, TELANGANA, INDIA

Postgraduate Program

M.TECH (SOFTWARE ENGINEERING)- PRR14

Rules, Regulations, Scheme of instruction & Evaluation and Syllabi
With effect from
2014-15

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

RULES AND REGULATIONS FOR POSTGRADUATE PROGRAMME – 2-YEAR M.TECH. DEGREE PROGRAMME (PRR-14)

(Applicable from the academic year 2014-15)

1. INTRODUCTION:

1.1 The provisions contained in these regulations given the conditions for imparting course of instructions, conducting examinations and evaluation of students performance leading to 2-year M.Tech. degree programme to be offered by Kakatiya Institute of Technology & Science, Warangal and awarded by Kakatiya University, Warangal

1.2 These regulations shall be called the ***"Kakatiya Institute of Technology & Science, Warangal (KITSW) regulations for the award of 2-year M.Tech. degree programme"*** *by Kakatiya University, Warangal*

1.3 They shall come into effect from the date of getting approval from the Academic Council of the Kakatiya Institute of Technology & Science, Warangal

1.4 They shall be applicable for all students enrolling for 2-year M.Tech. degree programme at the Kakatiya Institute of Technology & Science, Warangal from the academic year 2014-15.

2. DEFINITIONS:

2.1 ***"M.Tech."*** means Master of Technology, a Post-Graduate Degree awarded by Kakatiya University, Warangal

2.2 ***"University"*** means Kakatiya University, Warangal

2.3 ***"Institute"*** means Kakatiya Institute of Technology & Science, Warangal

2.4 ***"UGC"*** means University Grants Commission, New Delhi

2.5 ***"AICTE"*** means All India Council for Technical Education, New Delhi

2.6 ***"MHRD"*** means Ministry of Human Resource & Development, Govt. of India, New Delhi

2.7 ***"TSCHE"*** means Telangana State Council for Higher Education, Govt. of Telangana, Hyderabad

2.8 ***"GB"*** means Governing Body of the Institute

2.9 ***"AC"*** means Administrative Committee of the Institute

2.10 ***"FC"*** means Finance Committee of the Institute

2.11 ***"Council"*** means Academic Council of the Institute

2.12 ***"Principal"*** means Principal of the Institute

2.13 ***"Dean"*** means Dean of specific affairs of the Institute

2.14 ***"HoD"*** means Head of the Department of specific programme offered by the Institute

2.15 ***"BoS"*** means Board of Studies in the engineering of a specific programme offered by the Institute

2.16 ***"CoE"*** means Controller of Examinations of the Institute.

3. POST GRADUATE PROGRAMMES:

3.1 The Institute shall offer the following Post Graduate Programmes:

1. ***Structural & Construction Engineering (offered by the Dept. of Civil Engineering)***
2. ***Design Engineering (offered by the Dept. of Mechanical Engineering)***
3. ***Digital Communications (offered by the Dept. of Electronics & Communication Engineering)***
4. ***Software Engineering (offered by the Dept. of Computer Science & Engineering)***
5. ***VLSI & Embedded Systems (offered by the Dept. of Electronics & Instrumentation Engineering)***
6. ***Power Electronics (offered by the Dept. of Electrical & Electronics Engineering)***

3.2 The provisions of these regulations shall also be applicable to any new postgraduate programmes that are introduced from time to time with approval from appropriate bodies such as MHRD / AICTE / UGC, etc.

4. ADMISSION:

4.1

Course	Specialization	Eligibility		
		Qualifying Degree	GATE	PGECET
M.Tech.	Structural & Construction Engg.	B.E. / B.Tech. / AMIE in Civil Engineering / Construction Engineering or equivalent. They should have qualified at GATE/ PGECET	CE	CE
M.Tech.	Design Engineering	B.E. / B.Tech. / AMIE in Mechanical Engineering / Production Engineering / Industrial Engineering / Aeronautical Engineering / Marine Engineering or equivalent. They should have qualified at GATE / PGECET	ME	ME
M.Tech.	Digital Communications	B. E. / B.Tech. / AMIE in ECE, AMIE (Electronics & Telecommunication Engg. / B.E. / B.Tech. in Electrical or Electrical & Electronics Engg. EIE and Bio-medical Engg. or equivalent. They should have qualified at GATE / PGECET	EC / IN	EC
M.Tech.	Software Engineering	B.E. / B.Tech. / AMIE in any branch of Engg. / Tech. (Or) equivalent Master's Degree in Physics, Statistics, Mathematics, Applied Mathematics, Applied Statistics, Applied Physics, Geophysics, M.Sc. (Computer Science), M.Sc. (Information Systems), M.Sc. (Computer Applications & Electronics) and MCA or equivalent. They should have qualified at GATE / PGECET	CS	CS
M.Tech.	VLSI & Embedded Systems	B.E. / B.Tech. / AMIE in ECE, EIE, EEE, CSE, IT (Or) equivalent. They should have qualified at GATE / PGECET	CS / EC / IN / EE	EC
M.Tech.	Power Electronics	B.E. / B.Tech. / AMIE in Electrical & Electronics Engg. / Electrical Engg. or equivalent. They should have qualified at GATE / PGECET	EE	EE

4.2 **For GATE candidates**

The candidates should have passed 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. Sc.), M.Sc. (Information Systems) (Computer Applications and Electronics) and MCA or equivalent. They should have qualified at the GATE and possess a valid GATE score. The seats will be assigned purely on the basis of merit in GATE.

For Sponsored candidates

The candidates should have passed BE/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. Sc.), M.Sc. (Information Systems) (Computer Applications and Electronics) and MCA or equivalent.

The criterion for selection of sponsored candidates shall be by their merit at the entrance examination to be conducted by the PGECET

Admission shall be made into sponsored category only with the candidates who are qualified either in GATE/ PGECET or as decided by the admission committee.

1. His/ Her application shall be duly recommended by the sponsoring agency for admission to the course and forwarded to the Convener, PGECET
2. He/ She must be permanent employee with the sponsoring agency for at least two years, after obtaining the qualifying degree.
3. The sponsoring agency must be a Government Establishment or a Public-Sector undertaking, or a reputed Private Engineering College
4. The sponsoring agency shall certify that the candidates will be granted leave for pursuing the M.E./ M.Tech. as regular course of study.
5. The candidates who are working in Research Projects approved by the competent authority are also required to fulfill the above conditions before they are sponsored for admission

4.3 The Admissions shall be made in accordance with the rules and guidelines issued by TSCHE

5 ACADEMIC YEAR:

5.1 Each academic year is divided into two semesters (odd and even), each of 15 weeks including two Mid Semester Examinations. Academic session of the first semester will be decided based on counseling schedule declared by the TSCHE / Convener, PGECET

5.2 The Institute shall announce the schedule for all the academic activities for both the semesters (odd & even semesters) well before the commencement of the academic year and take all the necessary steps to follow them scrupulously.

5.3 The academic activities in a semester normally include registration, course work, Continuous Internal Evaluation (CIE), End Semester Examination (ESE) and declaration of results.

6. REGISTRATION:

6.1 All the students are required to register in person at the beginning of each academic year on the dates specified in the academic calendar / almanac.

6.2 The sole responsibility for registration rests with the student concerned.

6.3 Registration of students will be centrally organized by the Academic Section of the Institute.

6.4 The Registration procedure involves:

a) Filling of the prescribed registration form
b) Payment of fees and clearance of outstanding dues (if any).
c) Submitting undertaking (undertaking for regular attendance, discipline and against ragging) along with the parents.

6.5 If for any compelling reasons like illness, etc., a student is unable to register on the announced day of registration, he/she can register within 12 working days from the beginning of the academic year on payment of an additional late fee as prescribed by the Institute.

6.6 **No late registration shall be permitted after 12th working day** from the scheduled date of commencement of class work for that academic year.

6.7 Only those students will be permitted to register who have

a) cleared all institute and hostel dues of previous semesters.
b) paid all required prescribed fees for the current academic year.
c) not been debarred / detained from registering for a specified period on disciplinary or any other grounds.
d) cleared the minimum academic requirement as detailed in Regulation No. 14.

7. CURRICULUM

7.1 The duration of the programme leading to 2-year M.Tech. degree will be 4 semesters (2 academic years).

7.2 The curricula for 2-year M.Tech. degree programme with specializations as proposed by the department concerned and recommended by its BoS shall have the approval of the Academic Council.

7.3 The curricula to be followed for all the M.Tech. programmes is as specified and approved by the BoS of the department concerned.

7.4 The courses offered would have a ***Lecture – Tutorial - Practical (L-T-P)*** component to indicate contact hours/ periods. Separate laboratory (practical) course may exist ***(0-0-P)*** in certain cases as decided.

7.5 The academic programmes of the Institute shall follow the credit system.

7.6 Each course shall have an integer number of credits(C), which reflects its weightage. The number of credits of a course in a semester shall ordinarily be calculated as under:

$$\textit{Number of credits of a course, } C = L + (T+P)/2$$

- where L, T, P represent the No. of Lecture, Tutorial and Practical hours /periods per week.
- The fraction to be rounded off to next integer value.

7.7 **Course Code:** Each course offered in the Postgraduate (M.Tech.) curriculum at this institute shall be listed by using a total of 8 digits, as follows:

Ex: **P14SC101**

1. The first letter, to represent the **P**ost Graduate Programme
   Ex. **P** for Postgraduate Course
2. The next two numericals, to represent the year in which the syllabus is proposed / revised.
   Ex. **14** for the year 2014 from which syllabus is applicable for the batches admitted from academic year 2014-15.
3. The next two letters, to represent the post graduate specialization offered.
   Ex. **SC** for Structural & Construction Engineering
4. The last three numericals, to represent the course number and semester in which it is being offered.
   Ex. **XYZ**; X - Semester number ; YZ – Course number
   101 represents course number 01 offered in first semester

In general, a **course code** "**P14SC101**" represents a **Postgraduate Course number-01 offered for the batches admitted from the year 2014 in Structural & Construction Engineering in first semester.**

7.8 The syllabus of each course in the M.Tech. curriculum shall be divided into four units.

8. ATTENDANCE:

8.1 All the students are normally required to have full (100%) attendance.

8.2 However, the attendance in no case shall be less than 75% of the total classes held in all the courses offered in a semester for that academic year.

8.3 Students having attendance less than 75% in aggregate will be detained and are not allowed to appear for the end semester examination of that semester.

8.4 All such students who are detained have to repeat the entire semester when it is offered.

9. CONDUCT AND DISCIPLINE:

9.1 All students shall be required to conduct themselves in a manner befitting the reputation of the institution, within and outside the premises of the Institute; and are expected to complete their studies without any break.

9.2 As per the order of Hon'ble Supreme Court of India, ragging in any form is strictly banned. Involvement of a student in ragging will be considered as a gross indiscipline and may lead to his / her expulsion from the Institute.

9.3 Detailed rules regarding the conduct and discipline (code of conduct) are given in Appendix – I.

10 EVALUATION PROCEDURE:

10.1 The evaluation of students in a course for all 2-year M.Tech. programme (4 semesters) is a continuous process and is based on their performance in different examinations as mentioned below:

a) Sessional, involving **Continuous Internal Evaluation (CIE)** conducted all through the semester which includes **Mid-Semester Exams (MSE)** and **Teachers Assessment (TA)** through assignments.

b) Terminal, often designated as **End Semester Examination (ESE)** which includes a written examination for theory courses, practical, comprehensive viva-voce, dissertation examination with built-in oral part for laboratory / comprehensive viva-voce / dissertation courses.

10.2 A student's performance in a course (subject) shall be judged by taking into account the result of Continuous Internal Evaluation (CIE) and End Semester Examination (ESE) together.

10.3 Continuous Internal Evaluation (CIE) and End Semester Examination (ESE) shall have 40:60 weightage.

i.e. Continuous Internal Evaluation (CIE) carrying 40% weightage and End Semester Examination (ESE) carrying 60% weightage.

10.4 **Continuous Internal Evaluation (CIE) for Theory courses**:

10.4.1 The Continuous Internal Evaluation (CIE) throughout the semester shall consist of Teachers Assessment (TA) and Mid Semester Examination (MSE).

10.4.2 For assigning marks in Teachers Assessment (TA), performance in assignments is to be considered. Teacher shall give at least 2 assignments per each unit of syllabus covering the entire contents of that unit.

10.4.3 There shall be two mid semester examinations (MSE-I and MSE-II) of two hour duration for each course.

The average of the marks scored in MSE-I and MSE-II will be considered for evaluation under MSE. Hence, ***it is mandatory for the student to take both the mid semester examinations.***

10.4.4 The distribution given to each component of Continuous Internal Evaluation (CIE) for a theory course is given below:

S. No.	*Particulars*	*Weightage*
1.	*Teacher's Assessment (TA) (Assignments)*	*15%*
2.	*Mid Semester Examination (MSE) (MSE-I & MSE-II)*	*25%*
	Total Weightage::	***40%***

10.4.5 The marks obtained by the students in Mid Semester Examination (MSE) must be submitted to the Controller of Examination (CoE) by the teachers within 10 days from the date of conduct of the examination.

10.4.6 The dates for Mid Semester Examination (MSE) and End Semester Examination (ESE) will be declared by the CoE in consultation with the Dean, Academic Affairs.

10.5 **End Semester Examination (ESE) for Theory Course**:

There shall be an End Semester Examination (ESE) at the end of each semester for three hour duration for each course.

10.6 **Continuous Internal Evaluation (CIE) for Practical (Laboratory) Course:**

10.6.1 Continuous Internal Evaluation (CIE) for practical course shall carry 40% Weightage.

10.6.2 The Continuous Internal Evaluation (CIE) throughout the semester shall consist of the following:

Assessment	Weightage
Regular Experimentation / Job work	10%
Regular submission of record	10%
Quiz / Skill Test at the end of semester	10%
Viva-voce at the end of semester	10%
Total Weightage	**40%**

10.7 **End Semester Examination (ESE) for Practical (Laboratory) Course:**

10.7.1 There shall be an End Semester Examination (ESE) at the end of each semester for three hour duration for each practical course.

10.7.2 The End Semester Examination (ESE) for practical course shall carry 60% Weightage.

10.7.3 The marks distribution at End Semester Examination (ESE) shall be as follows:

Assessment	Weightage
Procedure / Experimentation / Tabulation / Result, as applicable …	40%
Viva-voce	20%
Total Weightage	**60%**

10.8 The **Department Post Graduate Review Committee (DPGRC)** shall be constituted with HoD as a Chairman, M.Tech. Coordinator as a Convener and Three to five other faculty members representing various specializations in that particular programme as members.

10.9 **Evaluation for Seminar :**

10.9.1 There shall be only Continuous Internal Evaluation (CIE) for Seminar, which includes Report Submission & Presentation

10.9.2 A teacher will be allotted to a student for guiding in

(i) selection of topic

(ii) report writing

(iii) presentation (PPT) before the DPGRC

10.10 **Evaluation for Comprehensive Viva-voce :**

There shall be only external oral examination for Comprehensive Viva-voce on a pre-notified date. The oral examination shall cover the entire content of courses covered in First and Second Semesters.

10.11 **Evaluation for Industrial Training:**

10.11.1 M.Tech. Coordinator in consultation with the Training & Placement Section has to procure training slots, for the students before the last day of instruction of 2nd semester.

10.11.2 The students shall confirm their training slots by the last day of 2nd semester

10.11.3 The students after 8 weeks Industrial Training shall submit a certificate, a report in the prescribed format before the last date specified by the Department Post Graduate Review Committee (DPGRC). The DPGRC shall evaluate their submitted reports and oral presentations.

10.12 **Continuous Internal Evaluation (CIE) for Dissertation:**

10.12.1 **Dissertation** shall be normally conducted in two stages, spread over two sequential semesters i.e. third and fourth semester.

10.12.2 **Registration Seminar** shall be arranged within four weeks after completion of the Industrial Training and Seminar in the 3rd semester. The Registration Seminar shall include a brief report and presentation focusing the identified topic, literature review, time schedule indicating the main tasks, and expected outcome.

10.12.3 **Progress Seminar-I:** At the end of first stage (third semester), student shall be required to submit a preliminary report of work done for evaluation to the project coordinator and present the same before the DPGRC. The Continuous Internal Evaluation (CIE) for the third semester is as follows:

Assessment	Weightage
Dissertation Supervisor Assessment	50%
DPGRC Assessment	50%
Total Weightage:	**100%**

10.12.4 **Progress Seminar-II** shall be arranged during the 6th week of IV semester.

10.12.5 **Progress Seminar-III** shall be arranged during the 15th week of IV semester.

10.12.6 **Synopsis Seminar** shall be arranged two weeks before the final thesis submission date. The student shall submit a synopsis report covering all the details of the works carried out duly signed by the Dissertation Supervisor.

10.12.7 At the end of second stage (fourth semester), student shall be required to submit two bound copies, one being for the department and other for the Dissertation Supervisor. The Dissertation report shall be evaluated by the DPGRC and external examination shall be conducted on a pre-notified date. The Dissertation evaluation for the fourth semester is as follows:

Assessment	Weightage
Dissertation Supervisor Assessment	20%
DPGRC Assessment	20%
ESE (Presentation & Viva-voce)	60%
Total Weightage:	**100%**

11 MINIMUM REQUIREMENT FOR PASSING A COURSE

11.1 **Theory Course**: A student is deemed to have passed in a theory course, if he / she secures

(a) 35 percent of marks assigned to End Semester Examination (ESE) and

(b) 35 percent of marks assigned to the Mid Semester Examination (MSE) and End Semester Examination (ESE) of the course taken together.

11.2 The average of the marks scored in both Mid Semester Examination (MSE) (as per the Regulation No. 10.4.4) will be considered for the evaluation under Mid Semester Examination (MSE).

11.3 **Laboratory Course**: A student is deemed to have passed in a laboratory course, if he / she secures

(a) 35 percent of marks assigned to End Semester Examination (ESE) and

(b) 35 percent of marks assigned to the Teachers Assessment (TA) and End Semester Examination (ESE) of the laboratory course taken together.

12 GRADING SYSTEM

12.1 At the end of the semester a student is awarded a letter grade in each of his / her courses taking into account his / her performance in Continuous Internal Evaluation (CIE) and End Semester Examination (ESE).

12.2 The typical grades and their numerical equivalents on 10-point scale (called Grade Points) are as follows:

Performance	Letter Grade	Grade Points (G_i)
Superior	S	10
Excellent	A	9
Very Good	B	8
Good	C	7
Average	D	6
Pass	P	4
Fail	F	0

12.3 F-Grade is a Fail Grade. The course in which the student has earned F-Grade will be termed as backlog Course.

12.4 In addition, there shall be a transitional M-grade.
M-Grade for "Debarred" due to malpractice / indiscipline during examination.

12.5 The Institute shall follow absolute grading system. The grades will be awarded as under:

Grade	Percentage Score (X)
S	$X \geq 90$
A	$80 \leq X < 90$
B	$70 \leq X < 80$
C	$60 \leq X < 70$
D	$45 \leq X < 60$
P	$35 \leq X < 45$
F	$X < 35$

12.6 For arriving at a grade obtained by a student in a particular course (subject), initially numeric marks obtained by the student out of 100 are to be determined. Once a numeric mark is obtained, the same is to be converted to a letter grade following the guidelines given in 12.5.

12.7 A Semester Grade Point Average (SGPA) will be computed for each semester. The SGPA will be calculated as follows:

$$\text{SGPA} = \sum_{i=1}^{n} C_i G_i \Big/ \sum_{i=1}^{n} C_i$$

where 'n' is the no. of courses (subjects) offered for the semester, 'C_i' is the credits allotted to a particular course, 'G_i' is the grade-points carried by the letter corresponding to the grade awarded to the student for the course as illustrated in 12.2.

12.8 The SGPA would indicate the performance of the student in the semester to which it refers. SGPA will be rounded off to the second place of decimal and recorded as such.

12.9 Starting from the second semester, at the end of each semester, a Cumulative Grade Point Average (CGPA) will be computed for every student as follows:

$$\text{CGPA} = \sum_{i=1}^{m} C_i G_i \Big/ \sum_{i=1}^{m} C_i$$

where 'm' is the total number of courses (subjects) the student has been offered from the first semester onwards upto and including the present semester, 'C_i 'and 'G_i' are as explained in 12.7.

12.10 The CGPA would indicate the cumulative performance of the student from the first semester up to the end of the semester to which it refers. CGPA will be rounded off to the second place of decimal and recorded as such.

12.11 SGPA and CGPA are calculated in consideration of only credits cleared, i.e. F-grade credits are not included for calculation.

13 SUPPLEMENTARY EXAMINATIONS

13.1 End Semester Examination (ESE) for each semester shall be conducted once in an academic year.

13.2 A student who obtained the F-grade in a course (theory or practical) can appear in a subsequent End Semester Examination (ESE) in the same course as supplementary candidate.

13.3 However the marks secured in Continuous Internal Evaluation (CIE) by the student in that course during the semester study shall remain unaltered.

13.4 The students those who have passed in the supplementary examination will be awarded grade with '*' marked on the courses passed in the supplementary.

13.5 Any candidate appearing for ESE in any course, after 2 years from his admission, shall be governed by the syllabus in force.

14 CONDITIONS FOR PROMOTION

14.1 A student shall have to satisfy the attendance requirements for the semester (as per the regulation No. 8) for promotion to the next higher semester.

15 GRADUATION REQUIREMENT

15.1 A student shall be declared to be eligible for award of the M.Tech. degree, if he / she has registered and completed all the courses with a minimum P-grade scored in every course

15.2 Normally a student should complete all the requirements consecutively in 4 semesters (2 academic years) for the award of M.Tech. degree. However, the students who fail to fulfill all the requirements for the award of M.Tech. degree within a period of 8 consecutive semesters (4 academic years from the registration in 1st semester) shall forfeit his / her enrolment to the program.

15.3 CGPA to Percentage (%) and Class Conversion is as follows:

S.No.	Division	Eligibility Criteria
1	First Division with Distinction	a) Student should secure CGPA≥8.0. b) Student should pass all the courses along with the batch of students admitted with him / her within 8 consecutive semesters. c) The failed candidate in any course shall not be awarded Distinction.
2	First Division	Student should secure CGPA, which is 6.5 ≤ CGPA < 8.0 within the time frame of the programme i.e. 8 semesters.
3	Second Division	Student should secure CGPA, which is 5.0 ≤ CGPA < 6.5 within the time frame of the programme i.e. 8 semesters.
4.	Pass Division	Student should secure CGPA, which is 4.0 ≤ CGPA < 5.0 within the time frame of the programme i.e. 8 semesters.

15.4 The University will award the post-graduate degrees to the students who are evaluated and recommended by the Institute.

16 MALPRACTICE IN EXAMINATION

16.1 Malpractice in examination is an illegal activity and is prohibited.

16.2 Mobile phones are strictly prohibited in the examination hall.

16.3 Exchange of question paper and material like pen, pencil, sharpener, eraser, scale, calculator, etc., during examination is strictly prohibited.

16.4 Malpractice in examination is viewed very seriously. Malpractice includes oral communication between candidates, possessing forbidden material, mobile phones (switched off/on) etc.

16.5 Any malpractice or engaging in any improper conduct and violation of the examination code by the student during examinations is liable for the punishment as given below:

S. No	Nature of Malpractice	S. No	Punishment
1.	Taking help from others, consulting and or helping other examinees during the examination period inside the examination hall or outside it, with or without their consent or helping other candidates to receive help from anyone else.	a)	Cancelling the examination of the paper in which he / she indulged in malpractices.
2	If the examinee attempts to disclose his / her identity to the valuer by writing his / her Hall-Ticket Number at a place other than the place prescribed for it or any coded message including his / her name or addressing the valuer in any manner in the answer book.		Cancelling the examination of the paper in which he / she indulged in malpractices.
3.	Candidate is found in possession of forbidden material; relevant or not relevant but not used.	b)	Cancellation of the result of all examinations taken or proposed to be taken during that session. However, he/ she shall be promoted to next semester/ year as per the promotion rules in vogue.
4.	Destroying the material found in his / her possession or acting in any other manner with a view to destroying evidence.	c)	Cancellation of the result of all examinations taken or proposed to be taken during that session and prohibiting his/her admission to or continuation in any course of the Institute for a period of one year. The student will be eligible to appear for the next corresponding semester / year examination in the succeeding academic year.
5.	Smuggling main answer book / additional answer book / question paper / matter in to or out of the examination hall & Conspiring to interchange Hall Ticket Numbers.		-do-
6.	Candidate is found in possession of forbidden material, relevant or not relevant but used.		-do-
7.	In case of (i) impersonation, (ii) misbehavior with the invigilators/any person related to examination work, (iii) insertion of written sheets in different hand writing in the main/additional answer book, and (iv) creation of disturbance in and around the examination hall during or before the examination.	d)	Cancellation of the result of all examinations taken or proposed to be taken during that session and prohibiting his/her admission in to or continuation in any course of the Institute for a period of two years. Further, the candidate shall not be allowed to appear for any examination during the period of punishment.

17 ROLL NUMBERS ALLOTMENT

The Roll Number given to the student shall have a total 8 digits as follows:

Ex: **M14SC007**

1. The first letter, to represent Masters (M.Tech.) degree programme.
   Ex: **M** for Masters programme
2. The next two numericals, to represent the year in which the student admitted into I semester.
   Ex: **14** for 2014
3. The next two letters, to represent the concerned specialization to which the student belongs.
   Ex: **SC** for Structural & Construction Engineering
4. The last three numericals, to represent the three digit roll number of the student.
   In general, a **student with roll number** "**M14SC007**" represents a **Masters student with a specialization of Structural & Construction Engineering admitted in the year 2014 bearing a roll number of 007.**

18 AMENDMENTS

Notwithstanding anything contained in this manual, the Academic Council of the Institute reserves the right to modify / amend the curricula, requirements and rules & regulations pertaining to its postgraduate programmes, without any notice.

KAKATIYA INSTITUTE OF TECHNOLOGY & SCIENCE, WARANGAL
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Scheme of Instruction and Evaluation for 2 year Postgraduate program

M.TECH (SOFTWARE ENGINEERING)

SEMESTER – I

Course Code	**Course Name**	*Periods per Week*			*Credits*	*Evaluation Scheme*				
						CIE			*ESE*	*Total Marks*
		L	***P***	***T***		***TA***	***MSE***	***Total***		
P14SE101	Discrete Mathematics & Optimization Techniques	3	-	1	4	15	25	40	60	100
P14SE102	Object Oriented Software Engineering	3	-	1	4	15	25	40	60	100
P14SE103	Software Requirement and Estimation	3	-	1	4	15	25	40	60	100
P14SE104	Advanced Data Structures & Algorithms	3	-	1	4	15	25	40	60	100
P14SE105	Elective-I	3	-	1	4	15	25	40	60	100
P14SE106	Elective-II	3		1	4	15	25	40	60	100
P14SE107	Object Oriented Software Engineering Laboratory	-	3	-	2	40	-	40	60	100
P14SE108	Advanced Software Laboratory		3	-	2	40	-	40	60	100
P14SE109	Seminar			-	2	100	-	100	-	100
	Total	***18***	***6***	***6***	***30***	***270***	***150***	*420*	***480***	***900***

Elective-I

P14SE105A	Secure Software Engineering
P14SE105B	Component Based Software Engineering
P14SE105C	Software Project Management
P14SE105D	Service Oriented Architecture

Elective-II

P14SE106A	Human Computer Interaction
P14SE106B	Advanced Operating Systems
P14SE106C	Real Time Systems
P14SE106D	Information Systems and Auditing

KAKATIYA INSTITUTE OF TECHNOLOGY & SCIENCE, WARANGAL
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Scheme of Instruction and Evaluation for 2 year Postgraduate program

M.TECH (SOFTWARE ENGINEERING)

SEMESTER – II

Course Code	**Course Name**	*Periods per Week*			*Credits*	*Evaluation Scheme*				
						CIE			*ESE*	*Total Marks*
		L	*P*	*T*		*TA*	*MSE*	*Total*		
P14SE201	Software Architecture & Design Patterns	3	-	1	4	15	25	40	60	100
P14SE202	Software Quality Assurance & Testing	3	-	1	4	15	25	40	60	100
P14SE203	Advanced Data Mining	3	-	1	4	15	25	40	60	100
P14SE204	Cloud Computing	3	-	1	4	15	25	40	60	100
P14SE205	Elective-III	3	-	1	4	15	25	40	60	100
P14SE206	Elective-IV	3		1	4	15	25	40	60	100
P14SE207	Software Testing Laboratory	-	3	-	2	40	-	40	60	100
P14SE208	Data Engineering Laboratory	-	3	-	2	40	-	40	60	100
P14SE209	Comprehensive Viva-Voce	-	-	-	2		-		100	100
	Total	***18***	***6***	***6***	***30***	***170***	***150***	*320*	***480***	***900***

Elective-III

P14SE205A	Model Driven Software Development
P14SE205B	Information Retrieval System
P14SE205C	Machine Learning
P14SE205D	Semantic Web & Social networks

Elective-IV

P14SE206A	Big Data Analytics
P14SE206B	Mobile Computing
P14SE206C	Soft Computing
P14SE206D	Distributed Computing

KAKATIYA INSTITUTE OF TECHNOLOGY & SCIENCE, WARANGAL
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Scheme of Instruction and Evaluation for 2 year Postgraduate program
M.TECH (SOFTWARE ENGINEERING)

SEMESTER – III

Course Code	**Course Name**	*Period*	***Credits***	*Evaluation Scheme*				
				CIE			*ESE*	***Total Marks***
				TA	***MSE***	***Total***		
P14SE301	Industrial Training	08 Weeks	*4*	*100*	-	*100*	-	*100*
P14SE302	Dissertation	16 Weeks	*8*	*100*	-	*100*	-	*100*
		Total	*12*	*200*	-	*200*	-	*200*

SEMESTER – IV

Course Code	**Course Name**	*Period*	***Credits***	*Evaluation Scheme*				
				CIE			*ESE*	***Total Marks***
				TA	***MSE***	***Total***		
P14SE401	Dissertation	24 Weeks	*12*	*40*	-	*40*	*60*	*100*
		Total	*12*	*40*	-	*40*	*60*	*100*

P14SE101 DISCRETE MATHEMATICS & OPTIMIZATION TECHNIQUES

M.Tech. Semester: I **Specialization: Software Engineering**

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To introduce the methods of optimization of both linear and non-linear objectives under a set of constraints.*
- *To introduce the techniques of solving decision making problems and analyze them in a competitive situations to get optimal output.*
- *To introduce the concepts and determination of optimal flow in a transport network and analysis of network scheduling by CPM-PERT with their practical applications.*
- *To introduce the basic concepts of Fuzzy sets, Fuzzy operations, Fuzzy logic and their Engineering applications.*

UNIT – I (9+3)

Constrained optimization: Linear programming concepts: Simplex method, Artificial variables method, Duality and Dual simplex method, **Integer Linear Programming:** Branch and Bound algorithm, Cutting plane algorithm, **Non-linear Programming concepts:** NLPP with equality and inequality constraints, Lagrange's method of multipliers, Kuhn-Tucker Conditions and Penalty function method.

UNIT – II (9+3)

Decision Analysis and Game Theory: Introduction to decision making problems, Decision making under uncertainty, Laplace criterion, Max-min criterion, Savage criterion and hurwitz criterion, Introduction to game theory, Games with pure strategies. Max-min and min- max principle, Optimal solution of two person zero-sum game, Dominance property, Solutions of mixed strategy games using graphical and linear programming methods.

UNIT-III (9+3)

Network Flows: Transport Networks, Flows in a network and maximal flows, Max flow- min cut theorem , Augmenting path method, Representation of project network, Network scheduling by CPM/PERT, Resource analysis in network scheduling.

UNIT-IV (9+3)

Fuzzy Sets and Fuzzy Logic: Basic concepts of fuzzy set and examples, Operations on fuzzy sets, Fuzzy complements, Fuzzy intersections, Fuzzy union and their properties, α-cuts and representation fuzzy sets, Generalized fuzzy operations, Complement, t-norms and T-Conorms, Simple theorems on fuzzy operations, Basic concepts of fuzzy logic, Fuzzy propositions and types of fuzzy propositions, Fuzzy quantifiers, Inferences from conditional fuzzy propositions, Qualified propositions and quantified propositions.

TEXT BOOKS:

1. Kandell, J.L Mott and Backer, "Discrete Mathematics", Prentice Hall of India, Second Edition, 81-203-1502-2, 1986.
2. George J.Kilr , Boyuan, "Fuzzy Sets and Fuzzy logic", Prentice Hall of India, 2003.
3. Kanti Swaroop, P.K. Gupta, ManMohan, "Operations Research", S.Chand Publications, Eleventh Edition,978-81-8054-909-0, 2010.
4. H.A. Taha, "Operations Research an Introduction", Prentice Hall of India, Sixth Edition, 81-7808-757-X, 2006.

REFERENCE BOOKS:

1. J.C. Panth, Introduction to optimization and operation research, Jain Brothers, 7th edition, 81-86321-88-8, 2006.
2. S.S.Rao, Engineering Optimization, Theory and Practice. New Age International (P) Ltd Publishers. 978-81-224-2723-3, Third Edition 2013.

Course Learning out Comes:

After attending the course the student will be able to

- *Solve any type of LPP and discuss the nature of the solution.*
- *Solve a class of non-linear programming problems with different types of constraints.*
- *Identify the importance of decision making systems and find an optimal solution of the problem given different types of nature of states.*
- *Analyze different strategies of a Game between two objects under conflicting situations.*
- *Develop an algorithm for solving problems of Game theory.*
- *Find a maximal flow of commodities in a transport network using different methods.*
- *Discuss different network based methods designed to assist in the planning, scheduling and control of projects.*
- *Identify the differences between Crisp sets and Fuzzy sets and the related properties.*
- *Differentiate between Classical systems and Fuzzy systems in order to solve the problems based on Fuzzy logic.*

P14SE102 OBJECT ORIENTED SOFTWARE ENGINEERING

M.Tech. **Semester**: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course learning objectives:

- *To make perfect the students in modeling the systems.*
- *To understand project requirements elicitation.*
- *To develop and design a simplified system with reduced complexity.*
- *To understand the various testing methods.*

UNIT – I (9+3)

Introduction to Software Engineering: Software engineering failures, What is software engineering, Software engineering concepts, Software engineering development activities, Managing software development, Object oriented paradigm, **Modeling with Unified Modeling Languages:** Introduction, An overview of UML, Modeling concepts and deeper view into UML, **Project Organization and Communication-Introduction:** A rocket example, An overview of projects, Project organization concepts, Project communication concepts, Organizational activities.

UNIT – II (9+3)

Requirements Elicitation-Introduction: Usability examples, An overview of requirements elicitation, Requirements elicitation concepts, Requirements elicitation activities, Managing requirements elicitation, **Analysis-Introduction:** An optical illusion, An overview of analysis, Analysis concepts, **Analysis Activities:** From use cases to objects, Managing analysis.

UNIT – III (9+3)

System Design: Decomposing the System-Introduction: A floor plan example, An overview of system design, System design concepts, **System Design Activities:** From objects to subsystems, **System Design:** Addressing design goals, Introduction, A redundancy example, An overview of system design activities, **Concepts:** UML deployment diagrams, **System Design Activities:** Addressing design goals, Managing system design.

UNIT – IV (9+3)

Object Design Reusing Pattern Solutions: Introduction- Bloopers, An overview of object design, **Reuse Concepts:** Solution objects, Inheritance and design patterns, **Reuse activities:** Selecting design patterns and components, Managing reuse, **Object Design Specifying Interfaces:** Introduction, A relational example, An overview of interface specification, Interface specification concepts, Interface specification activities, Managing object design, **Mapping Models to Code, Introduction:** A book example, An overview of mapping, Mapping concepts, Mapping activities and managing implementation, Testing**, Introduction:** Testing the space shuttle, An overview of testing, Testing concepts, Testing activities, Managing testing.

TEXT BOOKS:

1. Bernd Bruegge, Allen H.Dutoit, "Object Oriented Software Engineering Using UML, Patterns and Java", Second Edition, Pearson Education, 2004.
2. Stephen R Schach "Object Oriented & Classical Software Engineering" Fifth Edition TMH-2002

REFERENCE BOOKS:

1. Timothy C.Lethbridge, Robert Laganiere "Object Oriented Software Engineering Practical Software Development using UML & Java", TMH Edition 2004.
2. Grady Booch, James Rambaugh, Ivar Jacobson "The Unified Modeling Language user guide "Pearson education, 2006.

Course learning outcomes:

After completion of the course, the student will be able to

- *model the systems effectively.*
- *elicit the project requirements.*
- *design the system in a simplified and understandable.*
- *test the systems effectively using appropriate testing methods.*

P14SE103 SOFTWARE REQUIREMENTS AND ESTIMATION

M.Tech. Semester: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To make students understand the knowledge of software requirements elicitation*
- *To improve students capability in defining project objectives*
- *To make the students develop appropriate design solutions to a given problem*
- *To improve students capability in developing quality software artifacts and that should be satisfied by the client*

UNIT-I (9+3)

Software Requirements: 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 (9+3)

Software Requirements Management: 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 diagram, State transition diagram, Class diagrams, Object analysis, Problem frames.

UNIT III (9+3)

Software Estimation: 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, Conversion between size measures.

UNIT-IV (9+3)

Effort, Schedule and Cost Estimation: Productivity, Estimation factors, Approaches to effort and schedule estimation, COCOMO II, Putnam estimation model, Algorithmic models, Cost estimation, **Requirements and Estimation Management Tools:** Benefits of using a requirements management tool, commercial requirements management tool, Rational requisite pro, Caliber requirements management, Implementing requirements management automation, **Software Estimation Tools:** Desirable features in software estimation tools, International function point users group, USC's COCOMO II, Software life cycle management tools.

Text Books

1. Rajesh Naik and Swapna Kishore: Software Requirements and Estimation, 1st Edition, Tata Mc Graw Hill,ISBN-10, 0070403120, 2010
2. Karl E. Weigers: Software Requirements, 2nd edition Microsoft Press,ISBN-10, 073568798, 2008

Reference Books

1. Soren Lausen: Software Requirements Styles and Techniques, 1st edition, Addison-Wesley Professional, ISBN-10: 0201745704, 2009
2. Karl E.Weigers: Software Requirements Practical Techniques for gathering and Managing requirements through the product development life cycle, 2nd Edition, Microsoft Press, 2008

Course Learning Outcomes:

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

- *model, analyze and measure the software artifacts*
- *analyze, specify and document software requirements for a software system*
- *verify, validate, assess and assure the quality of software artifacts*
- *understand the impact of computing solutions in a global and societal context*

P14SE104 ADVANCED DATA STRUCTURES AND ALGORITHMS

M.Tech. Semester: I **Specialization: Software Engineering**

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To implement various operations on linear and non-linear data structures.*
- *To apply the suitable data structure to implement various sorting and searching techniques.*
- *To analyze the performance of different algorithms in terms of space and time.*
- *To understand various algorithm design methods and their usage in solving real world problems.*

UNIT-I (9+3)

Algorithms: Definition**,** Properties**, Performance Analysis:** Time complexity and space complexity, Asymptotic notations, **Data Structures:** Definition, Linear and non linear data structures, Abstract data type concept, **Trees:** Basic terminology, Binary search trees, Traversal methods, AVL trees, Splay trees, Red-black trees, Skip lists, **Graphs:** Graphs terminology, Representations, Graph traversals methods –Depth first search and breadth first search.

UNIT-II (9+3)

Searching- Linear and binary search methods, **Sorting**: Insertion sort, Heap sort and radix sort, **Internet Algorithms:** Strings and pattern matching algorithms, Cryptographic computations, Information security algorithms and Protocols, Network algorithms-Complexity measures and models, fundamental distributed algorithms, Broadcast and unicast routing, multi cast routing.

UNIT-III (9+3)

Algorithm Design Methods: Introduction, **Divide and Conquer:** General method, Merge sort, Quick sort, Sets and Disjoint sets, **Greedy method:** General method, Optimal storage on tapes, Knapsack problem, Minimum spanning trees, **Dynamic Programming:** Multistage graphs, Optimal binary search trees, Traveling sales person problem.

UNIT-IV (9+3)

Back Tracking: General method, 8- queens's problem, Graph coloring problem, **Branch and Bound:** Introduction, 0/1 knapsack problem, Traveling sales person problem, **Non-Polynomial-Hard and Non-Polynomial Complete Problems**: Basic concepts, Nondeterministic algorithms, the classes NP-Hard and NP-Complete, Cook's theorem.

Text Books:

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", Pearson Education Inc, 3rd Edition, ISBN 978-81-317-1474-4, 2009.
2. M T Goodrich, Roberto Tamassia, "Algorithm Design", John Wiley, 2nd edition, ISBN 812 65 098 64, 2008.
3. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms",Galgotia publications pvt Ltd, ISBN 81-7515-257-5, 2010.

Reference Books:

1. Debasis Samantha,"Classic Data Structures", PHI Learning Pvt. Ltd., 2nd Edition, ISBN-978-81-203-3731-2, 2009.
2. Aho, Hopcroft, Ulman, "The Design and Analysis of Computer Algorithms", Pearson Education Inc., ISBN 978-81-317-0205-5, 2009.
3. Sartaj Sahni, "Data Structures, Algorithms, and Applications in C++", Mc Graw-Hill, ISBN 978-00-711-8457-1, 2000.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know various linear and non-linear data structures, their operations and applications.*
- *analyze the performance of different algorithms in terms of space and time.*
- *implement various sorting and searching algorithms efficiently.*
- *select appropriate algorithm design method to solve a given real time problem.*

P14SE105A SECURE SOFTWARE ENGINEERING

M.Tech Semester: I **Specialization**: Software Engineering

Teaching Scheme:

L	T	P	C
3	1	-	4

Examination Scheme:

Continuous Internal Evaluation	40 Marks
End Semester Exam:	60 Marks

Course Learning Objectives:

- *To make students capable of understanding the specification and design of secure software.*
- *To make students capable of developing secure software*
- *To make students capable of testing security levels of an software*
- *To make students capable of managing secure software's.*

UNIT-I (9+3)

Software Security Issues: introduction, the problem, Software Assurance and Software Security, Threats to software security, Sources of software insecurity, Benefits of Detecting Software Security, **Secure Software Properties:** Properties of Secure Software, Influencing the security properties of Software, Asserting and specifying the desired security properties.

UNIT-II (9+3)

Requirements engineering for secure software: Introduction, the SQUARE process Model, Requirements elicitation and prioritization, **Secure Software Architecture and Design:** Introduction, software security practices for architecture and design, Architectural risk analysis.

UNIT-III (9+3)

Knowledge for secure software design: security principles, security guidelines and attack patterns. **Secure coding and Testing:** Code analysis, Software Security testing, Security testing, Considerations throughput the SDLC.

UNIT –IV (9+3)

Secure Systems Assembling Challenges: introduction, security failures, functional and attacker perspectives for security analysis, system complexity drivers and security, **Managing Secure Software's:** Governance and security, Adopting an enterprise software security framework, Deciding how much security is enough, Security and project management, Maturity of Practices.

Text Books:

1. Julia H. Allen, Nancy R. Mead, Sean J. Barnum, Robert J. Ellison,Gary," Software Security Engineering: A Guide for Project Managers", McGraw Edition , ISBN 978-0-321-50917, Addison- Wesley Professional, 2004.

Reference books:

1. Jason Grembi, "Developing Secure Software", Cengage Learning, ISBN:9788131508886, 2009.
2. Richard Sinn, "Software Security ", Cengage Learning, ISBN10: 142831945X, 2008.

Course Learning Outcomes:

After the completion of course, the student will be able to:

- *understand the specification and design of secure software.*
- *develop secure software*
- *test security levels of an software*
- *managing secure software's*

P14SE105B COMPONENT BASED SOFTWARE ENGINEERING

M.Tech. **Semester**: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To expose the students to the concepts of component-based software engineering*
- *To make students capable of using software engineering practices for component building*
- *To make students capable of manage component based software systems.*
- *To make students capable of understanding real-time component technologies*

UNIT I (9+3)

Component Introduction: Definition of a software component and its elements, The component industry metaphor, Component models and component services, An example specification for implementing a temperature regulator software component, **The Case for Components:** The business case for components, COT'S myths and other lessons learned in component-based software development.

UNIT II (9+3)

Software Engineering Practices for Component Building: Planning team roles for component development, Common high-risk mistakes, Integrating architecture, Process and organization, Practices of software engineering, Component-based software development, **The Design of Software Components:** Software components and the UML, Component infrastructures, Business components, Components and connectors, An open process for component based development, Designing models of modularity and integration.

UNIT III (9+3)

The Management of Component-Based Software Systems: Measurement and metrics for software components, Implementing a practical reuse program for software components, Selecting the right COT'S software, Building instead of buying, Software component project management, The trouble with testing components, Configuration management and component libraries, The evolution, Maintenance and management of component based systems.

UNIT IV (9+3)

Component Technologies: Component Technologies, Overview of the CORBA component model, Overview of COM+ component model, Overview of the EJB component model, Bonobo and free software GNOME components, Choosing between COM+, EJB, and GNOME, Software Agents as Next Generation Software Components.

TEXT BOOKS:

1. George T. Heineman, William T. Councill,"Component-based Software Engineering: Putting the Pieces", Addison-Wesley, ISBN 0201704854, 9780201704853, 2001.

REFERENCE BOOKS:

1. C. Szyperski, D. Gruntz and S. Murer, "Component Software, Second Edition", Pearson Education, ISBN 978-81-317-0523-0, 2002.
2. Ian Sommerville, "Software Engineering", Pearson education, seventh edition, ISBN 978-81-317-2461-3, 2007.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know essentials concepts component-based software engineering*
- *apply software engineering practices for component-based systems*
- *manage projects of component based software systems.*
- *utilize the real-time component technologies in software building*

P14SE105C SOFTWARE PROJECT MANAGEMENT

M.Tech. **Semester**: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To make students capable of understanding project management concepts and principles*
- *To make students capable of selecting the appropriate project development approach.*
- *To make students capable of performing costing and estimation of projects.*
- *To make students capable of performing risk assessment of projects.*

UNIT – I (9+3)

Introduction to Software Project Management: Introduction to project, Project versus product, Product versus process, Software projects versus other types of project, Contract management and technical project management, Activities covered by software project management, Plans, methods and methodologies, Some ways of categorizing software projects, The Management spectrum, Problems with software projects, Setting objectives, Stakeholders, The business case, Requirement specification, Management control, **Overview of Project Planning:** Introduction to step wise project planning, Select project, Identify project scope and objectives, Identify project infrastructure, Analyze project characteristics, Identify project products and activities, Estimate effort for each activity, Identify activity risks, Allocate resources, Review/publicize plan, Execute plan and lower levels of planning, Project Evaluation: Strategic assessment, Technical assessment, Cost-benefit analysis, Cash flow forecasting, Cost-benefit evaluation techniques, Risk evaluation.

UNIT – II (9+3)

Selection of an Appropriate Project Approach: Choosing technologies, Technical plan contents list, Choice of process models, Structure versus speed of delivery, The waterfall model, The V-process model, The Spiral Model, Software prototyping, Other ways of categorizing prototypes, Controlling changes during prototyping, Incremental delivery, Dynamic systems development method, Extreme programming, Managing iterative processes, Selecting the most appropriate process model, **Software Effort Estimation:** Observations on Estimation, Problems with over-and under-estimates, The basis for software estimating, Software estimating techniques, Expert judgement, Estimating by analogy, Albrecht function point analysis, Function points Mark II, Object points, A procedural code-oriented approach, COCOMO model, **Activity Planning:** The objectives of activity planning, When to plan, Project schedules, Projects and activities, Sequencing and scheduling activities, Network planning models, Formulating a network model, Adding the time dimension, The forward

pass, The backward pass, Identifying the critical path, Activity float, Shortening the project duration, Identifying critical activities, Activity-on-arrow networks.

UNIT – III (9+3)

Risk Management: The nature of risk, Types of risk, Managing risk, Hazard identification, Hazard analysis, Risk planning and control, Evaluating risks to the schedule, **Resource Allocation:** The nature of resources, Identifying resource requirement, Scheduling resources, Creating critical paths, Counting the cost, Being specific, Publishing the resource schedule, Cost Schedules, The scheduling sequence, **Monitoring and Control:** Creating the framework, Collecting the data, Visualizing progress, Cost monitoring, Earned value, Prioritizing monitoring, Getting the project back to target, Charge control.

UNIT – IV (9+3)

Managing Contracts: Types of contract, Stages in contract placement, Typical terms of a contract, Contract management, Acceptance, **Managing People and Organizing Teams:** Understanding behavior, Organizational behavior, Selecting the right person for the job, Instruction in the best methods, Motivation, The oldham-hackman job characteristics model, Working in groups, Becoming a team, Decision making, Leadership, Organizational structures, Stress, Health and safety, **Software Quality:** The importance of software quality, Defining software quality, ISO 9126, Practical software quality measures, Product versus Process quality management, External standards, Techniques to help enhance software quality, Quality plans.

TEXT BOOKS:

1. Bob Hughes and Mike Cotterell, "Software Project Management", Tata McGraw Hill, Third Edition, ISBN-13: 978-0077122799, 2002.
2. Walker Royce, "Software Project Management", Pearson Education, ISBN-13: 978-0321734020, 2006.

REFERENCE BOOKS:

1. Pressman, " Software Engineering", Tata Mc Graw Hill, Seventh Edition, ISBN 007-124083-7, 2011
2. Somerville, "Software Engineering", Tata Mc Graw Hill, Seventh Edition, 2009.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *understand project management concepts and principles*
- *select the appropriate project development approach.*
- *perform costing and estimation of projects.*
- *perform risk assessment of projects.*

P14SE106A HUMAN COMPUTER INTERACTION

M.Tech. Semester: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To expose students to the concepts, terminology, facts and principles in HCI.*
- *To know the relationships between specific instances and broader generalizations.*
- *To use concepts and principles to explain, analyze and solve specific situations.*
- *To of applying course content in coping with real life situations.*

UNIT-I (9+3)

Introduction: Importance of user Interface Definition, Importance of good design, Benefits of good design, A brief history of screen design, **The Graphical User Interface:** Popularity of graphics, The concept of direct manipulation, Graphical system, Characteristics, Web user, Interface popularity, Characteristics, principles of user interface.

UNIT-II (9+3)

Design Process: Human interaction with computers, Importance of human characteristics human consideration, Human interaction speeds and understanding business junctions, **Screen Designing:** Design goals, Screen planning and purpose, Organizing screen elements, Ordering of screen data and content, Screen navigation and flow , Visually pleasing composition, Amount of information, Focus and emphasis, Presentation information simply and meaningfully, Information retrieval on web, Statistical graphics , Technological consideration in interface design.

UNIT-III (9+3)

Windows: New and navigation schemes selection of window, Selection of devices based and screen based controls, **Components:** Text and messages, Icons and increases , Multimedia, Colors, Uses problems, Choosing colors.

UNIT-IV (9+3)

Software Tools: Specification methods, Interface, Building tools, **Interaction Devices:** Keyboard and function keys, Pointing devices, Speech recognition digitization and generation, Image and video displays, Drivers.

Text Books:

1. Wilbert O Galitz, "The Essential Guide to User Interface Design", Wiley Dream Tech, 2nd Edn., ISBN: 0-471-084646, 2002.
2. Ben Shneiderman, "Designing the User Interface", Pearson Education Asia, 3rd Edition, ISBN-10: 0-201-69497-2, 1998.

Reference Books:

1. Alan Dix, Janet Fincay, Gre Goryd, Abowd and Russell Bealg, "Human Computer Interaction", Pearson Education, 3rd Edition, ISBN-13: 978-0130461094, 2003.
2. Jenny Preece, Yvonne Rogers and Helen Sharp, "Interaction Design: Beyond Human – Computer Interaction", Wiley Dreamtech, 3rd Edition, ISBN-13: 978-0470665763, 2007.
3. Soren Lauesen , "User Interface Design: A Software Engineering Perspective", Addison Wesley, ISBN 10: 0321181433, 2005.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know the basics of human and computational abilities and limitations.*
- *understand basic theories, tools and techniques in HCI.*
- *learn the fundamental aspects of designing and evaluating interfaces.*
- *practice a variety of simple methods for evaluating the quality of a user interface.*
- *apply appropriate HCI techniques to design systems*

P14 SE106B ADVANCED OPERATING SYSTEMS

M.Tech. **Semester**: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning objectives:

- *To make students capable of understanding deadlocks and its recovery in distributed environment*
- *To make students known about load distribution requirements and algorithms*
- *To make students capable of system resource management and utilization*
- *To make students capable of understanding multiprocessor and data base operating systems*

UNIT-I (9+3)

Process Synchronization: Introduction, Functions of an operating system, Design Approaches, **Distributed Operating Systems:** Introduction, Motivations, System architecture types, Distributed operating systems, Issues in distributed operating systems, Communication networks, Communication primitives, **Distributed Mutual Exclusion:** The Classification of mutual exclusion algorithms, **Non_Token_Based Algorithms:** Lamport's algorithm, The ricart-agrawala algorithm, Maekawa's algorithm, **Token Based Algorithms**: Suzuki-kasami's broadcast algorithm, Singhal's heuristic algorithm, Raymond's tree based algorithm.

UNIT – II (9+3)

Distributed Deadlock Detection: Preliminaries, Deadlock handling strategies in distributed systems, Issues in deadlock detection and resolution, Control organizations for distributed deadlock detection, Centralized deadlock detection algorithms, Distributed deadlock detection algorithms, Hierarchical deadlock detection algorithms, **Distributed Resource Management:** Distributed file systems, Introduction, Architecture, Mechanism for building distributed file systems, Design issues, Log structured file systems.

UNIT-III (9+3)

Distributed Shared Memory: Architecture, Motivation, Algorithms for implementing DSM, Memory coherence, Coherence protocols, **Distributed Scheduling:** Issues in load distributing, Components of a load distributing algorithm, Load distributing algorithms, Task migration, **Failure Recovery:** Backward and forward error recovery, Recovery in concurrent systems, **Checkpoints:** Consistent, Synchronous check pointing and recovery, Asynchronous check pointing and recovery, Check pointing for distributed database systems, An algorithm for site recovery, **Fault Tolerance:** Introduction, Issues, Atomic actions and committing, Commit **Protocols:** Two-phase, Non-blocking commit protocols, **Voting Protocols:** Static and dynamic voting protocols.

UNIT – IV (9+3)

Multiprocessor Operating Systems: Motivations for multiprocessor systems, Basic multiprocessor System architectures: Tightly coupled versus loosely coupled, Uniform memory access Vs nonuniform memory Access Vs no remote memory access, **Interconnection Networks for Multiprocessor Systems:** Bus, Cross bar switch and multi stage inter connection network, Hyper cube architectures, **Case Studies:** The mach operating system, The sequoia system, **Database Operating Systems:** Introduction to database operating systems, Requirements of a database operating, **Concurrency Control:** The problem of concurrency control, Serializability Theory: Logs, Serial logs, Log equivalence, **Distributed Database Systems:** Data replication, Complications due to data replication, **Concurrency Control Algorithms:** Lock based algorithms, Timestamp based algorithms, Optimistic algorithms.

TEXT BOOK:

1) Mukesh Singhal, Niranjan G.Shivaratri, "Advanced Concepts In Operating Systems", Tata McGraw Hill Edition, 2001, ISBN 0-07-047268-8.

REFERENCE BOOKS:

1. Sinha, "Distributed Operating Systems Concepts and Design", IEEE Computer Society Press, 1997, ISBN - 0-7803-1119-1;
2. Tanenbaum and Steen, "Distributed Systems Principles and Paradigms", Prentice Hall Of India, 2002.

Course learning outcomes:

- *After the completion of course, the student will be able to*
- *understand deadlocks and its recovery in distributed environment*
- *known about load distribution requirements and algorithms*
- *perform system resource management and utilization*
- *understand multiprocessor and data base operating systems*

P14SE106C REAL-TIME SYSTEMS

M.Tech. **Semester**: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To know the concept of a real-time systems*
- *To know the role and the design process of a real-time operating system*
- *To know the using generic process architectures for monitoring control and data acquisition systems*

UNIT-I (9+3)

Typical Real-Time Application: Digital control, High-level controls, Signals processing, Other real-time application, **Hard Versus Soft Real-Time Systems:** Jobs and processors, Release times, Deadlines, and timing constraints, Hard and soft timing constraints, Hard real-time systems, Soft real-time systems, **A Reference Model of Real-Time Systems:** Processor and resources, Temporal parameters of real-time workload, Periodic task model, Precedence constraints and data dependency, Other types of dependencies, Functional parameters, Resources parameters of jobs and parameters of resources, Scheduling hierarchy.

UNIT-II (9+3)

Commonly Used Approaches to Real-Time Scheduling: Clock driven approach, Weighted round-robin approach, Priority driven approach, Dynamic versus static systems, Effective release times and deadlines, Optimality of the EDF and LST algorithms, Non-optimality of the EDF and the LST algorithms, Challenges in validating timing constraints in priority-driven systems, Off-line Vs On-line scheduling.
Clock-Driven Scheduling: Notations and assumptions, Static timer-driven scheduler, General structure of cyclic schedules, Cyclic executives, Improving the average response time of periodic jobs, Scheduling sporadic jobs, Practical considerations and generalizations, Algorithms for constructing static schedules, Pros and cons of clock-driven scheduling, **Priority driven scheduling of periodic task:** Static assumption, Fixed-priority versus dynamic priority algorithms, Maximum schedulable utilization, Optimality of the RM and DM algorithms, A schedulability test for fixed-priority tasks with short response times, Schedulability test for priority tasks with arbitrary response times. Sufficient schedulability conditions for the RM and DM algorithms.

UNIT-III (9+3)

Scheduling Periodic and Sporadic Jobs In Priority-Driven System : Assumptions and approaches, Deferrable servers, Sporadic servers, Constant utilization, Total bandwidth and weighted fair –queuing server, Slack stealing in deadline-driven systems, Slack stealing in fixed priority system, Scheduling of sporadic jobs, Real-time performance for jobs with soft timing constraints, A two-level scheme for integrated scheduling,

Resources and Resource Access Control: Assumptions on resources and their usage, Effects of resource contention and resource access control, Non-preemptive critical sections, Basic priority-inheritance protocol, Basic priority-ceiling protocol, Stack-based, Priority-ceiling protocol, Use of priority-ceiling protocol in dynamic-priority systems, Preemptive-ceiling protocol, Controlling access to multiple-unit resources, Controlling concurrent access to data objects.

UNIT-IV(9+3)

Real-Time Databases: Basic Definitions, Real-time vs. general purpose databases, Main memory databases, Transaction priorities, Transaction aborts, Concurrency control issues, Disk scheduling algorithm, A two-phase approach to improve predictability, Maintaining serialization consistency, Databases for hard real-time systems, **Fault-Tolerance Techniques:** What causes failures, Fault types, Fault detection, Fault and error containment, Redundancy, Data diversity, Reversal checks, Malicious or byzantine failures, Integrated failure handling.

TEXT BOOK:

1. Jane W. Liu, "Real-Time Systems" Pearson Education, Ist Edition, ISBN-13: 9780130996510, 2001.
2. Rajib Mall, "Real-Time Systems: Theory and Practice," Pearson Education, 2nd Edition, ISBN 978-81-317-0069-3, 2008.

REFERENCE BOOK:

1. C.M.Krishna, Kang G. Shin, “Real-Time Systems”, Tata McGraw Hill International Edition, ISBN-13: 9780070570436, 1997.
2. Philip Laplante, “Real-Time Systems Design and Analysis”, Prentice Hall of India,2nd Edition, ISBN 13: 9788120316843, 2005

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know the fundamental concepts in applications of computer science*
- *apply knowledge in advanced computer science to formulate the analyze problems in computing and solve them*
- *apply knowledge to the design and conduct experiments as well as to analyze and interpret data*
- *gain knowledge on emerging concepts in theory and applications of computer science*

P14 SE106D INFORMATION SYSTEMS AND AUDITING

M.Tech. **Semester**: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *This course helps to learn minute of information systems auditing organization and evaluating the different phases in systems development measures.*
- *It provides better understanding on security Mechanisms quality assures management and controls.*
- *It drives students to work on communication controls and processing controls.*
- *It gives better perception of concurrent auditing techniques and evaluating the system in an efficient way.*

UNIT-I (9+3)

Overview of Information Systems Auditing: Need for control and Audit of computers, Effect of computers on Internet Controls, Effects of Computers on Auditing, Foundations of Information Systems Auditing, **Conducting an Information Systems Audit:** The Nature of Controls, Dealing with Complexity, Audit Risks, Types of Audit Procedures, overview of Steps in an Audit, Auditing auditing Around or through the computers, **Top Management Controls:** Evaluating the Planning function, Evaluating the Organizing function, Evaluating the Leading Function, Evaluating the Controlling Function, **Systems Development Management Controls:** Approaches to Auditing Systems Development, Normative Models of the Systems Development Process, evaluating the major Phases in the Systems Development Process, **Programming Management Controls:** The Program Development life Cycle, Organizing the Programming Team, Managing the System Programming Group.

UNIT-II (9+3)

Data Resource Management Controls: Motivations toward the DA and DBA Roles, Functions of the DA and DBA, Some Organizational issues, Data Repository Systems, control Over the DA and DBA, **Security Management Controls:** Conducting a security Program , Major Security threats and Remedial measures, Controls of Last Resort, Some Organizational Issues, **Operations Management Controls:** Computer operations, Network operations, Data Preparation and Entry, Production Control, File Library, Documentation and Program Library, Help Desk/Technical Support, Capacity Planning and Performance Monitoring, Management of Outsourced Operations, **Quality Assure Management Controls :** Motivations To Ward the QA Role , QA Functions, Organizational Considerations, **Boundary Controls:** Cryptographic Controls, Access controls, Personal Identification Numbers, Digital Signatures, Plastic Cards , Audit Trail Controls , Existence Controls.

UNIT-III (9+3)

Input Controls: Data input methods, Source document design, Data –entry screen design, Data code controls, Check digits, Batch controls, Validation of data input, Instruction input,

Validation of instruction input, Audit trail controls, Existence controls, **Communication Controls:** Communication subsystem exposures, Physical component controls, Line error controls, Flow controls, Link controls, Topological controls, Channel accesses controls, Controls over subversive threats, Internetworking controls, Communication architectures and controls, Audit trail controls, Existence controls, **Processing Controls:** Processor controls, Real memory controls, Virtual memory controls, Operating system integrity, Application software controls, Audit trail controls, Existence controls, **Database Controls:** Access controls, Integrity controls, Application software controls, Concurrency controls, Cryptographic controls, File handling controls, Audit trail controls, Existence controls, **Output Controls:** Inference controls, Batch output production and distribution controls, Batch report design controls, Online output production and distribution controls, Audit trail controls, Existence controls.

UNIT-IV (9+3)

Audit Software: Generalized audit software, Industry-specific audit software, High-level languages, Utility software, Expert systems, Neural network software, Specialized audit software control audit software, **Code Review, Test Data and Code Comparison**: Program source –code review , Test data , Program code comparison, **Concurrent Auditing Techniques**: Basic nature of concurrent auditing techniques, Need for concurrent auditing techniques, Types of concurrent auditing techniques, Implementing concurrent auditing techniques, Strengths/limitations of concurrent auditing techniques, **Evaluating System Efficiency**: The evaluation process, Performance indices, workload models, System models, **Managing the Information System Audit Function:** Planning function, Organizing function, Staffing function, Leading function, Controlling function.

Text Book:

1. Ron Weber, "Information Systems Control and Audit", Pearson Education ISBN: 978-81-317-0472-1, 1999.

Reference Book:

1. Richard E.Cascarino, "Information Systems and auditing ", John Ailey Publications.

Course Learning Out Comes:

After completion of the course, the student will be able to

- *get knowledge on the information systems auditing and different audit procedures.*
- *understands the security management, operations management and quality assurance management controls for organizational issues.*
- *realize the process controls and database controls in information auditing.*
- *acquire insights on audit software and code review mechanism.*

P14SE107 OBJECT ORIENTED SOFTWARE ENGINEERING LABORATORY

M.Tech. **Semester**: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
-	-	3	2

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To understand modeling the systems.*
- *To learn various UML diagrams.*
- *To expose effectively using UML diagrams in object oriented software design.*
- *To know the importance of case tools in software development and maintenance.*

The following Experiments are suggested in this laboratory.

List of Experiments:

1. Developing Use-Case Analysis.
2. Developing Class & Object Diagrams.
3. Developing Interaction & State Chart Diagrams.
4. Developing Activity Diagrams.
5. Developing Component & Deployment Diagrams.
6. Developing Forward Engineering and Reverse Engineering Diagrams.
7. Case study on simple watch system & Library Information system.
8. Case study on Railway Reservation System & 2-floor elevator simulator system.
9. Case study on ATM (Automatic Teller Machine) System & Online Examination System.
10. Case study on Hospital Management System & Online Shopping System.
11. Requirement Specification & Project Estimation Tools.
12. Software Maintenance Tools.

Course Learning Outcomes:

After completion of the course, the student will be able to,

- *model the system.*
- *design the object oriented software systems effectively using UML diagrams.*
- *use case tools effectively.*

P14SE 108 ADVANCED SOFTWARE LABORATORY

M.Tech. **Semester**: I **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
-	-	3	2

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To learn windows applications in .net programming.*
- *To learn working with the data stored in databases in .net programming.*
- *To know the implementation of windows services and web services in .net programming.*
- *To create various web forms using Java scripts and Java Server Pages.*
- *To develop programs using java beans.*

PART-A

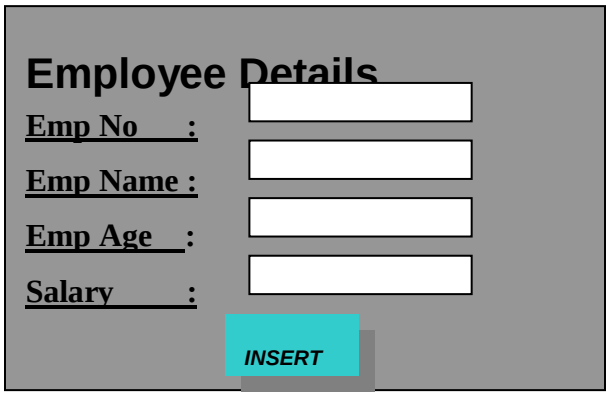
List of experiments on .Net Programming:

1. Program to implementation of scientific calculator.
2. Program to implement the bouncing ball.
3. Program to draw Circle, Rectangle, Line, Ellipse and to fill them.
4. Program for creation of common dialog controls.(Open, Save, Font, Color).
5. Program to scroll the image using Scrollbars.
6. Program to handle the printer operations.
7. Program to Read/Write from/to Text files and Binary files.
8. Program to create a web form using the validation controls.
9. Program for create the Database table and perform the following operations.
   i) Insertion ii) Deletion iii) Updation iv) Editing
10. Write the above program in Web Application.
11. Program to Synchronize the threads.
12. Program to create a Windows service and a Web service.

PART-B

List of experiments on Java 2 Enterprise Edition (J2EE):

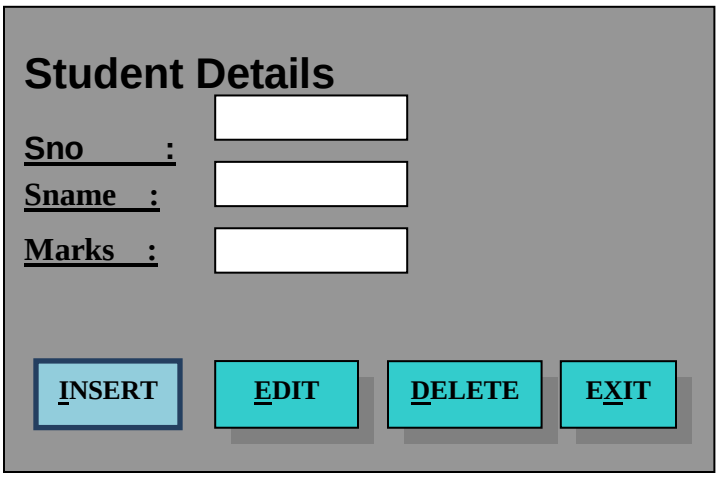
1. Write a JSP program for displaying Employee details in a tabular format.
2. Write a JSP program to generate the following Employee form



Provide validations using Java Script for

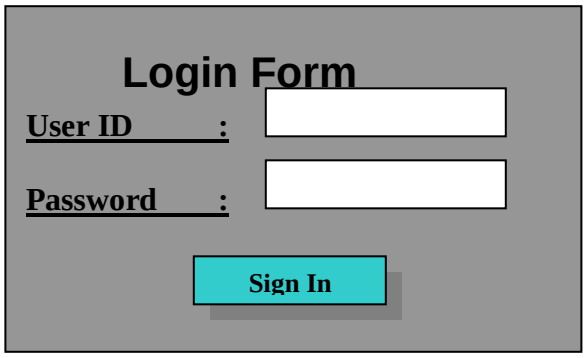
1) All fields are mandatory
2) Empno, age, salary should be numeric
3) Age should be between 20 and 30
4) Employee name should be alphanumeric and should start with an upper case letter.

3. Design a form as follows



i) Provide validations using Java Script
ii) Provide multiple buttons and call appropriate JSP files.

4. Design form as follows



- Provide all validations
- Display appropriate messages like
- –“ welcome to user” if user id exists and password is correct
- –“Welcome password “ if user id exists and password is wrong
- “ Invalid user” if user id does not exists
- Create a login table in oracle and connect to it.

5. Create a Bean for displaying welcome message, Invoke the Bean from JSP.

6. Create a Bean for implementing Account operations.

Account Consists:

Datamembers : Accno, Balance, Account Type

Methods **:** Deposit, withdraw, getBalance

- Write a JSP program which invoke the Account Bean and provide interface for invoking the methods of it.

7. Write a stateless Session Bean for accepting a string and returns “Welcome to “ followed by the accepted string.

8. Write a stateless Session Bean, which provides a remote interface consisting following interfaces.
   i. void store(int a, int b)
   ii. int add();
   iii. int mul();

9. Write implementation file which implements the above methods.

10. Write a Client program for locating Session object and invokes the above methods.

11. Write a Stateless Session Bean for the above problem and observe the difference.

12. Create an Entity Bean which implements the Account Entity.
    i. Data Members:
       accountNo, balance, accountType.
    ii. Data Methods:
        1. void deposit(double amt);
        2. void withdraw(double amt);
        3. double getBalance();

13. Use Bean Managed Persistence as persistent-type.
14. Create an Entity Bean which Implements the Account Entity as above problem by using Container Managed Persistence as persistent-type

Course Learning Outcomes:

After completion of the course, the student will be able to

- *write .Net programs to develop windows applications.*
- *establish the connection with the database in .Net programming.*
- *implement web services and windows services.*
- *to create web forms Java scripts and JSP.*
- *develop java beans for the user requirements.*

P14SE109 SEMINAR

Class: **M.Tech I Semester** **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
-	-	-	2

Examination Scheme :

Continuous Internal Evaluation	100 marks
End Semester Exam	--

Guidelines:

The Department Post Graduate Review Committee (DPGRC) shall be constituted with HoD as a Chairman, M.Tech. Coordinator as a Convener and Three to five other faculty members representing various specializations in that particular programme as members.

There shall be only Continuous Internal Evaluation (CIE) for Seminar, which includes Report Submission & Presentation

A teacher will be allotted to a student for guiding in

(i) Selection of topic
(ii) Report writing
(iii) Presentation (PPT) before the DPGRC

P14SE201 SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation :	40 marks
End Semester Exam :	60 marks

Course Learning Objectives:

- *To learn about the Software architecture design and evaluation processes.*
- *To understand the concept of patterns and the Catalog.*
- *To know the Software architectures for product lines*
- *To understand the behavioral pattern and iterator pattern*
- *To understand design patterns to keep code quality high without overdesign.*

UNIT I (9+3)

Envisioning Architecture: The architecture business cycle, What is software architecture, Architectural patterns, Reference models, Reference architectures, Architectural structures and views, **Creating Architecture:** Quality attributes, Achieving qualities, Architectural styles and patterns, Designing the architecture, Documenting software architectures, Reconstructing software architecture.

UNIT II (9+3)

Analyzing Architectures: Architecture evaluation, Architecture design decision making, Architecture Tradeoff Analysis Method, Cost-Benefit Analysis Method, Moving **from one system to many:** Software product lines, Building systems from off the shelf components, Software architecture in future.

UNIT III (9+3)

Patterns: Pattern description, Organizing catalogs, Role in solving design problems, Selection and usage, **Creational and Structural Patterns:** Abstract factory, Builder, Factory method, Prototype, Singleton, Adapter, Bridge, Composite, Façade, and Flyweight.

UNIT IV (9+3)

Behavioral Patterns: Chain of responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Strategy, Template method, Visitor, **Case Studies:** A-7E – A case study in utilizing architectural structures, The world wide web - a case study in interoperability, Air traffic control – a case study in designing for high availability, Celsius tech – a case study in product line development.

TEXT BOOKS:

1. Len Bass, Paul Clements & Rick Kazman, "Software Architecture in Practice", second edition, Pearson Education, ISBN-13: 078-5342154955, 2003.
2. Erich Gamma, "Design Patterns", Pearson Education, ISBN-13: 078-5342633610, 1995.

REFERENCE BOOKS:

1. Luke Hohmann, Addison wesley, "Beyond Software architecture", 2003.
2. David M. Dikel, David Kane and James R. Wilson, "Software architecture", Prentice Hall PTR, 2001
3. David Budgen, "Software Design", Second edition, Pearson education, 2003
4. Eric Freeman & Elisabeth Freeman," Head First Design patterns", O'Reilly, 2007.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *design software architecture for large scale software systems*
- *describe a software architecture using various documentation approaches and architectural description languages*
- *identify and assess the quality attributes of a system at the architectural level*
- *communicate program structures using design patterns.*
- *select appropriate design patterns for design problems.*

P14SE202 SOFTWARE QUALITY ASSURANCE AND TESTING

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation :	40 marks
End Semester Exam :	60 marks

Course Learning Objectives:

1. *To understand the scope of software testing and quality assurance in software development life cycle*
2. *To know testing and quality assurance activities using modern software tools*
3. *To prepare test plans, schedules and budget for a testing and quality assurance projects*
4. *To manage testing and quality assurance projects*

UNIT-I (9+3)

Software Quality: Perspective and expectations, Historical perspective of quality, Quality frameworks, Quality assurance as dealing with defects, Defect prevention detection and containment strategies, **Quality Assurance Process and Quality Engineering:** QA activities in software processes, Verification and validation perspectives, Reconciling the two views quality engineering, Activities and process quality planning, Goal setting and strategy formation quality assessment and improvement quality engineering in software processes.

UNIT-II (9+3)

Software Testing Background: Infamous software case studies, Bug, Why do bugs occur, The cost of bugs, What exactly does a software testing do, What makes a good software tester, **The Realities of Software Testing:** Testing axioms, Software testing terms and definitions, Precision and accuracy, Testing and quality assurance, **Examining the Specification:** Black-box and white-box testing, Static and dynamic testing, Performing a high level review of the specification, Low level specification test techniques, **Testing the Software with Blinders:** Dynamic black-box testing, Test-to-pass and test-to-fail, Equivalence partitioning data testing, State testing, Other black-box test techniques, **Examining the Code:** Static white-box testing, Examining the design code, Formal reviews, Peer reviews, Walk through, Inspectors, Coding standards and guidelines, Examples of programming standards and guidelines, Obtaining standards, Generic code review checklist.

UNIT-III (9+3)

Testing the Software with Dynamic White-Box Testing: Dynamic white-box testing, Dynamic white-box testing Vs. debugging, Testing the pieces, Data coverage, Code coverage, **Configuration Testing:** An overview of configuration testing, Approaching the task, Obtaining the hardware, Identifying hardware standards, Configuration

testing other hardware, **Compatibility Testing:** Compatibility testing overview, Platform and application versions, Standards and guidelines data sharing compatibility, **Usability Testing:** User interface testing, What makes good user interface testing, Guidelines, Intuitive consistent, Flexible, Comfortable, Correct, Useful, Accessibility testing, Accessibility features in software.

UNIT-IV (9+3)

Testing the Documentation: Types of software documentation, The importance of documentation testing, Reviewing documentation, The realities of documentation testing, **Web Site Testing:** Black-box testing, Gray-box testing, White-box testing, Configuration and compatibility testing, Usability testing, Introducing automation, **Automated Testing and Test Tools:** The benefits of automation and tools, Test tools, software test automation, Random testing, **Planning Test Effort:** The goal of test planning, Test planning topics, **Writing and Tracking Test Cases:** The goals of test case planning, Test case planning overview, Test case organization and tracking, **Reporting:** Bugs fixation, Isolating and reproducing bugs, A bug's life cycle and bug tracking systems, **Measuring Testing Results:** Metrics for testing, Common project-level metrics.

Text Books:

1. Jeff Tian, "Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement", John Wiley and Sons, Inc., and IEEE Computer Society Press, ISBN 0-471-7345-7, 2005.
2. Ron Patton, "Software Testing", Pearson Education, Second Edition, ISBN 978-81-7758-031-0, 2004.

Reference Books:

1. Edwar.Dkit. "Software testing in the Real World", Pearson Education, Third Edision, ISBN 978-81-7758-572-8, 2003.
2. M.G.Limaye, "Software Testing: Principles Techniques and Tools", Tata McGraw-Hill Education Pvt. Ltd., First Edison, ISBN 10: 0070139903 / ISBN 13: 9780070139909, 2009.
3. Cem Kaner, Jack Falk, Hung Quoc Nguyen, "Testing Computer Software", Second Edition, International Thomson Computer Press, ISBN 1850328471, 9781850328476, 1993.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know the scope of software testing and quality assurance in software development life cycle*
- *capable of performing testing & quality assurance activities using modern software tools*
- *develop test plans, schedules and budget for a testing & quality assurance projects*
- *effectively manage a testing & quality assurance projects*

P14SE203 ADVANCED DATA MINING

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation :	40 marks
End Semester Exam :	60 marks

Course Learning Objectives:

- *To apply data mining techniques in real world applications*
- *To know advances in classification and clustering algorithms*
- *To develop web and text mining applications*
- *To know temporal and spatial mining applications*

UNIT I (9+3)

Data mining overview: Mining frequent patterns, Associations and correlations, Classification and regression for predictive analysis, Cluster analysis, Outlier analysis, **Pattern Mining Overview:** Pattern mining in multilevel, Multidimensional space mining multilevel Associations, Mining multidimensional associations, Mining quantitative association rules, Mining rare patterns and negative patterns.

UNIT II (9+3)

Advance Classification: Classification by back propagation, Support vector machines, Classification using frequent patterns, Classification using rough sets and fuzzy sets.
Advance Clustering: Density - based methods –DBSCAN, OPTICS, DENCLUE, Grid-Based methods – STING, CLIQUE, **Exceptions:** Maximization algorithm, Clustering Dimensional Data.

UNIT III (9+3)

Graph Mining: Introduction, Clustering graph and network data, **Web Mining:** Introduction, Web content mining, Web structure mining and web usage mining,
Text mining: Unstructured text, Episode rule discovery for texts, Hierarchy of categories, Text clustering.

UNIT IV (9+3)

Temporal Data Mining: Temporal association rules, Sequence mining, GSP algorithm, SPADE, SPIRIT episode discovery, Time series analysis, **Spatial Mining**: Spatial mining tasks, Spatial clustering, Data mining applications.

TEXT BOOKS:

1. Jiawei Hang and Micheline Kamber, "Data Mining Concepts and Techniques", MorganKaufmannn, ISBN 978-0-12-381479-1, 2nd Edition, 2010.
2. Arun K pujari, "Data Mining Techniques", ISBN 81-7371-380-4,Universities Press, 2001.

REFERENCE BOOKS:

1. Pang-Ning Tan, Vipin kumar, Michael Steinbach, "Introduction to Data Mining", Pearson.
2. T.V Sveresh Kumar, B.Esware Reddy, "Data Mining Principles & Applications", Elsevier.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *apply the data mining algorithms for real world problems*
- *analyze advances in classification and clustering algorithms*
- *able to build web and text mining applications*
- *gain knowledge in temporal and spatial mining applications*

P14SE 204 CLOUD COMPUTING

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To introduce various cloud computing models.*
- *To understand cloud services and solutions.*
- *To know about cloud virtualization technologies and cloud management.*
- *To understand the relevance of cloud and SOA.*

UNIT-I (9+3)

Introduction: Introduction, Essentials, Benefits, Why cloud, Business and IT perspective, Cloud and virtualization ,Cloud services requirements , Cloud and dynamic infrastructure , Cloud computing characteristics , Cloud adoption, **Cloud Models:** Cloud characteristics, Measured service, Cloud models, Security in a public cloud, Public versus private clouds, Cloud infrastructure self Service.

UNIT-II (9+3)

Cloud as a Service: Gamut of cloud solutions, Principal technologies, Cloud strategy, Cloud design and implementation using SOA, Conceptual cloud model, Cloud service defined, **Cloud Solutions**: Introduction, Cloud ecosystem, Cloud business process management, Cloud service management, Cloud stack, Computing on demand (CoD), Cloud sourcing.

UNIT-III (9+3)

Cloud Offerings: Information storage, Retrieval, Archive and protection, Cloud analytics, Testing under cloud, Information security, Virtual desktop infrastructure, Storage cloud, **Cloud Management**: Introduction, Resiliency, Provisioning, Asset management, Cloud Governance, High availability and disaster recovery, Charging models, Usage reporting, Billing and metering.

UNIT-IV (9+3)

Cloud Virtualization Technology: Virtualization defined, Virtualization benefits, Server virtualization, Virtualization for x 86 architecture, Hypervisor management software, Logical partitioning (LPAR), VIO server, Virtual infrastructure requirements, **Cloud Virtualization:** Introduction, Storage virtualization, Storage area networks, Network, Attached storage, Cloud server virtualization, Virtualized data center, **Cloud and SOA**: Introduction, SOA journey to infrastructure, SOA defined, SOA and IAAS, SOA based cloud infrastructure steps, SOA business and IT services.

Text Books:

1. Kumar Saurabh, "Cloud Computing: Insights into New-Era Infrastructure", Wiley India, ISBN 8126528834,2011.

Reference Books:

1. Herbert Schildt, "Complete Reference with C", Tata McGraw Hill, 4th Edition, ISBN-13: 9780070411838, 2000
2. Barrie Sosinsky, " Cloud Computing Bible", John Wiley & Sons, 2010

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know the different cloud models.*
- *understand various services of cloud.*
- *gain knowledge on cloud virtualization technologies.*
- *learn cloud and SOA concepts*

P14SE205A MODEL DRIVEN SOFTWARE DEVELOPMENT

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To expose the students to the concepts of model driven software development*
- *To understand model driven software development relationships with other practices*
- *To manage projects based on model driven software development*
- *To know utilizing modern technologies for model driven software development*

UNIT-I (9+3)

Model Driven Software Development (MDSD): basic ideas, terminology, challenges, goals, approaches and architecture, case study on web application, **Model Concept formation:** Common model driven software development concepts and terminology, model driven architecture, architecture centric model driven software development, Generative programming.

UNIT-II (9+3)

MDSD Classifications: Model driven software development verses computer aided software engineering, 4GL, wizard, roundtrip engineering, Model driven software development and Patterns, Model driven software development and domain driven design, **MDSD Capable Target Architecture:** Software architecture in the context of MDSD, Building blocks of software architecture, Architecture reference model, balancing the MDSD platform, MDSD and component based development and Service oriented architecture.

UNIT-III (9+3)

Building domain architecture: Domain Specific Language construction, General transformation architecture, technical aspects of building transformations, and the use of interpreters, **Code generation techniques:** categorization, generation techniques Model transformations with Query view transformation, Model to model language requirements. MDSD tools, roles, architecture, selection criterion and pointers.

UNIT-IV (9+3)

Modular-based software design: Model-driven Architecture, Meta modeling, Meta levels vs Levels of abstraction, **Model Driven Architectures Framework:** Platform Independent Model, Platform Specific Model, System modeling- MOF's Meta modeling.

Text Book:

1. Thomas Stahl, Markus Voelter, "Model-Driven Software Development: Technology, Engineering, Management", Wiley, ISBN: 978-0-470-02570-3, 2006.

Reference Book:

1. Anne Kleppe, Jos Warmer and Wim Bast, "The Model Driven Architecture, Practice and Promise", Pearson Education, ISBN-13: 978-8177589702, 2003.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know essentials concepts model driven software development*
- *apply model driven software development for real time practices*
- *manage projects of model driven software development.*
- *utilize the real-time technologies for model driven software development*

P14SE205B INFORMATION RETRIVAL SYSTEM

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation :	40 marks
End Semester Exam :	60 marks

Course Learning Objectives:

- *To expose the concepts of Information retrieval systems*
- *To analyze advances in information retrieval algorithms*
- *To know advances in web searching technologies*
- *To develop text classification based applications*

Unit I (9+3)

Boolean retrieval, Term vocabulary and postings lists, Dictionaries and tolerant retrieval, Index construction, Index compression.

UNIT II (9+3)

Scoring, term weighting and the vector space model, Computing scores in a complete search system, Evaluation in information retrieval, Relevance feedback and query expansion.

UNIT III (9+3)

XML retrieval, Probabilistic information retrieval, Language models for information retrieval, Web search basics, Web crawling and indexes, Link analysis.

UNIT IV (9+3)

Text classification, Vector space classification, Support vector machines and machine learning on documents, flat clustering, Hierarchical clustering, Matrix decompositions and latent semantic indexing.

TEXT BOOKS:

1. Christopher D. Manning and Prabhakar Raghavan and Hinrich Schütze, "Introduction to Information Retrieval ", Cambridge University Press, ISBN 1139472100, 2008.

REFERENCE BOOKS:

1. Kowalski, Gerald, Mark T Maybury, "Information Storage and Retrieval Systems: Theory and Implementation", Springer, ISBN 0-972-37924-1, 2002.
2. Ricardo Baeza-Yates, Modern Information Retrieval, Pearson Education, Ist edition, ISBN 978-81-317-0977-1, 2007.
3. David A Grossman and Ophir Frieder, "Information Retrieval: Algorithms and Heuristics", 2nd Edition, Springer, ISBN 1-4020-3004-5, 2004.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know essentials concepts Information retrieval systems*
- *analyze advances in information retrieval algorithms*
- *gain knowledge in advances of web searching technologies*
- *build text classification based application*

P14SE205C MACHINE LEARNING

M.Tech. Semester: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation :	40 marks
End Semester Exam :	60 marks

Course Learning Objectives:

- *To understand the basic theory in machine learning.*
- *To understand a range of machine learning algorithms along with their strengths and weakness.*
- *To understand the machine learning algorithms implemented in different fields of computers.*
- *To formulate machine learning problems corresponding to different applications.*
- *To apply machine learning algorithms to solve problems of moderate complexity out code.*
- *To read current research papers and understands the issues raised by current research out code.*

UNIT-I (9+3 Hrs)

Introduction: Learning, Types of machine learning, Supervised learning, Designing a learning system, Perspectives and issues in machine learning, **Concept learning and the general to specific ordering –** Introduction, A concept learning task, Concept learning as search, Find-S: finding a maximally specific hypothesis, Version spaces and the candidate elimination algorithm, Remarks on version spaces and candidate elimination, Inductive bias.

UNIT-II (9+3 Hrs)

Linear Discriminants: Preliminaries, The perceptron, Linear separability, Linear regression, **Artificial Neural Networks**: Introduction, Neural network representation, Appropriate problems for neural network learning, Perceptions, An illustrative example face recognition, Advanced topics in artificial neural networks, **Multi-Layer Perceptron**: Going forwards, Going Backwards, Back propagation of error, The multi-layer perceptron in practice examples of using the MLP, Radial basis functions and splines concepts, The radial basis function (RBF) network, The curse of dimensionality, Interpolation and basic functions.

UNIT-III (9+3 Hrs)

Decision Tree learning – Introduction, Decision tree representation, Appropriate problems for decision tree learning, The basic decision tree learning algorithm, Hypothesis space search in decision tree learning, Inductive bias in decision tree learning, Issues in decision tree learning, **Computational learning Theory:** Introduction, Probability learning an approximately correct hypothesis, Sample complexity for finite hypothesis space, Sample complexity for infinite hypothesis spaces, The mistake bound model of instance based learning, k -Nearest neighbor learning, Locally weighted regression, Radial basis functions, Case-based reasoning, Remarks on lazy and eager learning.

UNIT-IV (9+3 Hrs)

Unsupervised Learning: The k-Means algorithm, Vector quantisation, Self-organizing features map, **Dimensionality Reduction:** Linear discriminant analysis, Principles components analysis, Factor analysis, Independent components analysis, Locally linear embedding, Multi-dimensional scaling, **Evolutionary Learning:** Motivation, Genetic algorithms, An illustrative example, Hypothesis space search, Genetic programming, Models of evolution and learning, Parallelizing genetic algorithms. Case study: Optical character recognition, Speech recognition using hidden markov model.

TEXT BOOKS:

1. Tom M. Mitchell, "Machine Learning", MGH, ISBN-13: 978-0070428072, 1997
2. Stephen Marsland, Taylor & Francis,"Machine Learning: An Algorithmic Perspective", CRC, ISBN-13: 978-1420067187, 2009

REFERENCE BOOKS:

1. William W Hsieh,"Machine Learning Methods in the Environmental Sciences, Neural Networks", Cambridge Univ Press, ISBN-13: 978-0805822410, 2009
2. Richard o. Duda, Peter E. Hart and David G. Stork, Pattern Classification, John Wiley & Sons Inc, ISBN-13-978-0471056690, 2001
3. Chris Bishop, Neural Networks for Pattern Recognition, Oxford University Press, ISBN-13 978-0198538646, 1995

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know fundamental issues and challenges of machine learning: data, model selection, model complexity.*
- *know Strengths and weaknesses of many popular machine learning approaches.*
- *appreciate the underlying mathematical relationships within and across Machine Learning algorithms and the paradigms of supervised and un-supervised learning.*
- *design and implement various machine learning algorithms in a range of real-world applications.*

P14 SE205D SEMANTIC WEB AND SOCIAL NETWORKS

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation :	40 marks
End Semester Exam :	60 marks

Course Learning Objectives:

- *To learn Web Intelligence.*
- *To learn Ontology Engineering.*
- *To learn Knowledge Representation for the Semantic Web.*
- *To expose Semantic Web Applications, Services and Technology.*
- *To expose Social Network Analysis and semantic web.*

UNIT-I (9+3)

Web Intelligence: Thinking and intelligent web applications, The Information age ,The world wide web, Limitations of today's web, The next generation web, **Machine Intelligence:** Artificial intelligence, Ontology, Inference engines, Software agents, Limitations and capabilities, **Berners-Lee:** WWW, Semantic road map, Semantic web services, Logic on the semantic web.

UNIT-II: (9+3)

Ontology Engineering: Ontology engineering, Constructing ontology, Ontology development tools, Ontology methods, Ontology libraries and ontology mapping, **Knowledge Representation for the Semantic Web:** Ontologies and their role in the semantic web, Ontologies languages for the semantic web – resource description framework(RDF), RDF schema, Ontology web language(OWL), UML, XML/XML schema.

UNIT-III (9+3)

Semantic Web Applications, Services and Technology: Semantic web applications and services, Semantic search, e-learning, Semantic bioinformatics, Knowledge base , XML based web services, Creating an OWL-S ontology for web services, Semantic search technology, Web search agents and semantic methods.

UNIT-IV (9+3)

Social Network Analysis and semantic web: What is social networks analysis, development of the social networks analysis, Electronic sources for network analysis – electronic discussion networks, Blogs and online communities, Web based networks. Building semantic web applications with social network features.

Text Books:

1. H. Peter Alesso, Craig F. Smith, "Thinking on the Web: Berners-Lee, Gödel and Turing", Wiley-Blackwell, 1st Edition, ISBN-13: 978-0-471-76866-1, 2008.
2. Peter Mika, "Social Networks and the Semantic Web", Springer, 1st Edition, ISBN-13: 978-0-387-71000-6, 2007.

Reference Books:

1. Rudi Studer, Stephan Grimm, Andreas Abecker, "Semantic Web Services: Concepts, Technologies, and Applications", Springer, 2007 Edition, ISBN 978-3-540-70893-3, 2007.
2. Liyang Yu, "Semantic Web and Semantic Web Services", Chapman and Hall/CRC Publishers, 1st Edition, ISBN-13: 978-15848893 35, 2007.
3. Heiner Stuckenschmidt, Frank Van Harmelen, "Information Sharing on the Semantic Web", Springer Publications, 1st Edition, ISBN-13: 978-3642058233, 2010.
4. John Hebeler , Matthew Fisher , Ryan Blace , Andrew Perez-Lopez , Mike Dean, "Semantic Web Programming", Wiley, 1st Edition, ISBN: 047041801X, 2011.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *understand different techniques in web semantics.*
- *understand different tools, methods and mapping in Ontology engineering.*
- *analyze web services, semantic search techniques to develop semantic web applications.*
- *analyze social network structure and different sources for it.*

P14 SE206A BIG DATA ANALYTICS

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *Students will be able to learn about Big data analytic processes and tools.*
- *Students will be able to know about Big data architecture and reports.*
- *Students will be able to use Map reduce for building Big data applications.*
- *Students will be able to learn Frequent Item sets and Clustering.*
- *Students will be able learn Frameworks and Visualization.*

UNIT I (9+3)

Introduction:, Velocity, Variety, Veracity, and Drivers for Big Data, Sophisticated consumers, Automation, Monetization, **Big Data Analytics Applications**: Social Media command center, Product knowledge hub, Infrastructure and operations studies, Product selection, Design and engineering, Location-based services, Online advertising, Risk management.

UNIT II (9+3)

Architecture Components: Massively parallel processing platforms, **Unstructured Data Analytics and Reporting:** Search and count, Context-sensitive and domain-specific searches, Categories and ontology, Qualitative comparisons, Data privacy protection, Real-Time adaptive analytics and decision engines, **Advanced Analytics Platform:** Real-Time architecture for conversations, Orchestration and synthesis using analytics engines, Entity resolution, Model management, Discovery using data at rest, Integration strategies.

UNIT III (9+3)

Implementation of Big Data Analytics: Revolutionary, Evolutionary or hybrid, Big Data governance, Integrating Big Data with MDM, Evolving maturity levels, **Map-Reduce and New Software Stack**: Distributed file systems, Physical organization of compute nodes, Large-scale file-system organization, Map-reduce features: Map tasks, Grouping by key, Reduce tasks, Combiners, Map-reduce execution, Coping with node failures, Algorithms using map-reduce for matrix multiplication, Relational algebra operations, Workflow systems, Recursive extensions to map-reduce.

UNIT IV (9+3)

Communication Cost Models: Complexity theory for map-reduce, Reducer size and replication rate, Graph model and mapping schemas, Lower bounds on replication rate, **Mining Data Streams**: Stream data mode l and management stream source, Stream queries and issues, Sampling data in a stream , Filtering streams, Counting distinct elements in a stream, Estimating moments, Counting ones in a window, Decaying windows, **Link Analysis**: Page ranking in web search engines, Efficient computation of page rank using map-reduce and other approaches, Topic-sensitive page rank, Link spam, Hubs and authorities.

TEXT BOOKS:

1. *Dr. Arvind Sathi,* "Big Data Analytics:Disruptive Technologies for Changing the Game" , IBM Corporation, First Edition, ISBN: 978-1-58347-380-1,2012.
2. Anand Rajarama, Jure Leskovec, Jeffrey D. Ullman. "Mining of Massive Datasets", Prime, ISBN-13: 978-1107015357, 2013.

REFERENCES:

1. Soumendra Mohanty, Madhu Jagadeesh, Harsha Srivatsa, Apress, "Big Data Imperatives", Apress, ISBN: 978-1-4302-4872-9, 2012.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *learn about Big data analytic processes and tools.*
- *know about Big data architecture and reports.*
- *use Map reduce for building Big data applications.*
- *learn Frequent Item sets and Clustering.*
- *learn Frameworks and Visualization.*

P14SE206B MOBILE COMPUTING

M.Tech. Semester: II **Specialization: Software Engineering**

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course learning objectives:

- *To learn about the concepts and principles of mobile computing*
- *To distinguish between types of mobility*
- *To study the specifications and functionalities of various protocols of mobile networks*
- *To make students expertise in working with application protocols to develop mobile applications*
- *To explore both theoretical and practical issues of mobile computing*

UNIT I (9+3)

Introduction to Mobile Computing: Introduction to mobile computing, Mobile Computing, Functions, Devices, Applications and services, **Mobile Computing Architecture:** Architecture for mobile computing, Three-tier architecture -presentation tier, Application tier, Data tier, Design considerations, **Mobile Computing:** Client context manager, Context aware systems.

UNIT II (9+3)

Global System for Mobile Communications (GSM): GSM architecture, GSM entities call routing in GSM, PLMN interfaces, GSM addresses and identifiers, Network aspects in GSM, GSM frequency allocation, Authentication and security, **General Packet Radio Service (GPRS):** GPR, GPRS and packet data network, GPRS network architecture & operations, Applications and limitations GPRS, Wireless Application Protocol: WAP Architecture, Wireless markup language, WML script.

UNIT III (9+3)

Mobile Network Layer: Goals, assumptions, entities and terminology, IP packet deliver agent advertisement and discovery, registration, tunneling and encapsulation, optimizations. Dynamic host configuration protocol, **Mobile Ad-Hoc Networks:** Routing, Destination sequence distance vector dynamic source routing, **Mobile Transport Layer:** Traditional TCP, Indirect TCP, Snooping TCI mobile TCP, Fast retransmit/fast recovery, Transmission /time-out freezing, Selective retransmission, Transaction oriented TCP.

UNIT IV: (9+3)

Broadcast Systems: Overview, Cyclical repetition of data, Digital audio broadcasting: Multimedia object transfer protocol. **Digital Video Broadcasting:** DVB data broadcasting, DVB for high-speed internet access, Convergence of broadcasting and mobile communications, **Emerging Technologies:** Introduction, Bluetooth, Radio

Frequency identification, Wireless: Broadband, IPv6, J2ME, **Security Issues in Mobile Computing:** Introduction information security, Security models, Security frameworks for mobile environment.

TEXT BOOKS:

1. Jodien Schiller, "Mobile Communications", Pearson Education, Second Edition, ISBN 81-7808-560-7, 2005.
2. Asoke K Talukder, "Mobile Computing", Tata McGraw Hill, ISBN: 0-07-014457-5, 2008.

REFERENCE BOOKS:

1. Reza Behravanfar, "Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML", ISBN: 0521817331, Cambridge University Press, October 2004.
2. Frank Adelstein et al, "Fundamentals of Mobile and Pervasive computing", TMH. 2005.
3. Gary S.Rogers, et al, "An Introduction to Wireless Technology", Pearson Education, 2007.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *know the basic concepts and principles of mobile computing*
- *know the characteristics and limitations of mobile hardware devices including their user-interface modalities.*
- *understand the positioning techniques and location based services and applications*
- *know the structure and components for Mobile IP and Mobility management*
- *organize the functionalities and components of mobile computing systems into different layers.*

P14SE206C SOFT COMPUTING

M.Tech I Semester **Specialization:** Software Engineering

Teaching Scheme: **Examination Scheme**:

L	T	P	C
3	1	-	4

Continuous Internal Evaluation:	40 Marks
End Semester Exam:	60 Marks

Course Learning Objectives:

- *To learn the key aspects of soft computing.*
- *To know about the components and building block hypothesis of Genetic algorithm.*
- *To understand the features of neural network and its applications.*
- *To study the fuzzy logic components.*
- *To gain insight onto hybrid systems and Neuro Fuzzy modeling*

UNIT-I (9+3)

Introduction to Soft Computing and Genetic Algorithms: Evolution of computing, Soft computing constituents and conventional artificial intelligence, Soft computing characteristics, Introduction to GA, Building block hypothesis, Introduction to Genetics-based machine learning applications of GA.

UNIT-II (9+3)

Fuzzy Logic: Fuzzy sets, Operations on fuzzy sets, Fuzzy relations, Membership functions, Fuzzy rules and fuzzy reasoning, Fuzzy inference systems, Fuzzy expert systems, Fuzzy decision making

UNIT-III (9+3)

Neural Networks: Basics of artificial neuron model, Adaptive networks, Feed forward networks, Supervised learning neural networks, Radial basis function networks, Reinforcement learning, Unsupervised learning neural networks.

UNIT-IV (9+3)

Hybrid Systems: Integration of neural networks, Fuzzy logic, Genetic algorithms, Neuro fuzzy modeling, ANFIS architecture, CANFIS architecture, Classification and regression trees, Data clustering algorithms, Rule base structure identification.

Text Books:

1. Jyh – Shing Roger Jang, Chuen – Tsai Sun, Eiji Mizutani, "Neuro – Fuzzy and Soft Computing", Prentice – Hall of India, 2008,ISBN:978-81-317-1109-5.
2. David E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning", Addison Wesley, 2007,ISBN: 9780201157673.
3. George J. Klir and Bo Yuan, "Fuzzy Sets and Fuzzy Logic – Theory and Applications", Prentice Hall, 1995,ISBN:81-203-1136-1.
4. Neural Networks, Fuzzy logic and Genetic Algorithms, Synthesis and applications by S. Rajsekharan, Vijayalaxmi Pai,2006,ISBN:81-203-2186-3.

Reference Books:

1. Kwang H.Lee, "First course on Fuzzy Theory and Applications", Springer, Verlag Berlin Heidelberg, 2005,ISBN:978-3-540-22988-95.
2. N.K.Bose and P.Liang "Neural Network Fundamentals with graphs, Algorithms and Applications" Tata Mcgraw -Hill, 1998,ISBN: 0-07-463529-8.
3. James A. Freeman and David M. Skapura, "Neural Networks Algorithms, Applications, and Programming Techniques", Pearson Edn., 2003.
4. Mitsuo Gen and Runwei Cheng,"Genetic Algorithms and Engineering Optimization", Wiley Publishers 2000.
5. Mitchell Melanie, "An Introduction to Genetic Algorithm", Prentice Hall, 1998.

Course Learning Outcomes:

After the completion of course, the student will be able to

- *implement machine learning through neural networks.*
- *gain knowledge to develop genetic algorithm .*
- *develop genetic algorithm to solve the optimization problem.*
- *develop a fuzzy expert system to derive decisions.*
- *model neuro fuzzy system for data clustering and classification*

P14SE206D DISTRIBUTED COMPUTING

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
3	1	-	4

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To know the issues involved in distributed systems.*
- *To learn distributed computing paradigms.*
- *To learn computer networks architecture relevant to distributed computing.*
- *To understand basic knowledge of net-centric computing.*

UNIT-I (9+3)

Introduction: Definition of distributed System, Characteristics of distributed systems, Goals of distributed system, Hardware concepts, Software concepts, Client-server model, Model of a distributed computation, **Communication:** Layered protocols, Remote procedure call, Remote object invocation, Message oriented communication, Stream oriented communication.

UNIT-II (9+3)

Processes: Threads, Clients, Servers, Code migration, Software agents, **Naming:** Naming Entities, Name resolution, Implementation of namespace, DNS, X.500, Locating mobile entities, Naming Vs locating entities, Home-based approaches, Hierarchical approaches, Removing unreformed entities, **Distributed Based Systems:** CORBA processes, Naming, synchronization, caching and replication, Fault tolerance, security, Distributed COM, GLOBE and comparison.

UNIT-III (9+3)

Distributed Document Based Systems: WWW, Lotus notes and comparisons, **Distributed Coordination Based Systems:** TIB/Rendezvous Overview, Communication, Processes, Naming, Synchronization, Caching and replication, Security; JINI, Comparisons of JINI and TIB/Rendezvous, **Software Agents**: Definition, Terminology, Agent Technology, Mobile Agents.

UNIT-IV (9+3)

Distributed Multimedia Systems: Characteristics of multimedia data, Quality of service management, Resource management, Stream Adaptation, **Computing Technologies:** Cluster/parallel computing, Coordination/Scheduling, Distributed object computing, Peer-to-peer computing, Service-oriented computing.

Text Books:

1. Andrew S. Tanenbaum and Marteen Van Steen, "Distributed Systems: Principles and Paradigms", Prentice Hall, 2nd Edition, ISBN: 0-13-088893-1, 2002.
2. Colouris G., Dollimore Jean, Kindberg Tim, "Distributed Systems Concepts and Design", Pearson education, 3rd Edition, ISBN-10: 0201619180, 2001.

Reference Books:

1. Singhal M, Shivaratri N.G, "Advanced concepts in operating systems". Mc-Graw-Hill Intl., ISBN: 007057572X, 1994.
2. Eric Newcomer, "Understanding Web Services: XML, WSDL, SOAP, and UDDI", Addison-Wesley Professional, ISBN-13: 078-5342750812, 2002.
3. James Edward Keogh, "J2EE: The complete Reference", Mc-Graw-Hill, ISBN: 007222472X, 2002.
4. Rajkumar Buyya, "High Performance Cluster Computing: Architectures and Systems", Vol. 1, Pearson Education, ISBN: 9788131716939, 1999.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *problem solving skills to distributed application.*
- *identify and decompose complex systems into its components parts*
- *integrate OS and programming language concepts to solve distributed components of the system.*
- *practice a variety of simple methods for develop suites of networking protocols for implementing the communicating components.*

P14SE 207 SOFTWARE TESTING LABORATORY

M.Tech. Semester: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
-	-	3	2

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

To learn to use the following (or similar) automated testing tools to automate testing:

- *Win Runner/QTP for functional testing.*
- *Load Runner for Load/Stress testing.*
- *Test Director for test management.*
- *JUnit, HTMLUnit, CPPUnit.*

List of experiments on testing:

1. Write programs in 'C' Language to demonstrate the working of the following constructs:
   i) do...while ii) while….do iii) if…else iv) switch v) for
2. "A program written in 'C' language for Matrix Multiplication fails" Introspect the causes for its failure and write down the possible reasons for its failure.
3. Take any system (e.g. ATM system) and study its system specifications and report the various bugs.
4. Write the test cases for any known application (e.g. banking application)
5. Create a test plan document for any application (e.g. Library Management System)
6. Study of any testing tool (e.g. Win runner)
7. Study of any web testing tool (e.g. Selenium)
8. Study of any bug tracking tool (e.g. Bugzilla, bugbit)
9. Study of any test management tool (e.g. Test Director)
10. Study of any open source-testing tool (e.g. Test Link)
11. Take a mini project (e.g. University admission, Placement Portal) and execute it. During the Life cycle of the mini project create the various testing documents
12. Take a mini project (e.g. Library Management, Student register for a Course) and execute it. During the Life cycle of the mini project create the various testing documents.

Course Learning Outcomes:

After completion of the course, the student will be able to

- *exposure on Win-runner and QTP for functional testing.*
- *use load runner for load and stress testing.*
- *use test director for test management.*
- *work with JUnit, HTMLUnit, CPPUnit.*

P14SE208 DATA ENGINEERING LABORATORY

M.Tech. **Semester**: II **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
-	-	3	2

Examination Scheme :

Continuous Internal Evaluation	40 marks
End Semester Exam	60 marks

Course Learning Objectives:

- *To make students aware of real-time data warehousing tools*
- *To make students able to implement data mining algorithms*
- *To make students able to build data mining applications for credit risk analysis*
- *To make students capable to use WEKA tool for testing data mining algorithms*

List of experiments:

1. Evolution of data management technologies, introduction to data warehousing concepts.
2. Develop an application to implement defining subject area, design of fact dimension table, data mart.
3. Develop an application to implement Extract, Transform and Load operations on a data warehouse
4. Develop an application to implement OLAP, roll up, drill down, slice and dice operation
5. Develop an application to construct a multidimensional data.
6. Develop an application to implement data generalization and summarization technique.
7. Develop an application to extract association rule of data mining.
8. Develop an application to extract data pattern
9. Develop an application for classification of data.
10. Develop an application for decision tree.
11. Develop an application for clustering technique
12. Develop an application to credit risk assessment

Course Learning Outcomes:

After completion of the course, the student will be able to

- adopt real-time data warehousing tools
- implement data mining algorithms
- build data mining applications for credit risk management
- use WEKA tool for testing data mining algorithms

P14SE209 COMPREHENSIVE VIVA-VOCE

Class: **M.Tech II Semester** **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
-	-	-	2

Examination Scheme :

Continuous Internal Evaluation	--
End Semester Exam	100 marks

Guidelines:

There shall be only external oral examination for comprehensive viva-voce on a pre-notified date. The oral examination shall cover the entire content of courses covered in first and second semesters.

P14SE301 INDUSTRIAL TRAINING

M.Tech III Semester **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
-	-	-	4

Examination Scheme :

Continuous Internal Evaluation	100 marks
End Semester Exam	--

Guidelines for Industrial Training:

Coordinator in consultation with the Training & Placement Section has to procure training slots, for the students before the last day of instruction of 2nd semester. The students shall confirm their training slots by the last day of 2nd semester.

The students after 8 weeks Industrial Training shall submit a certificate, a report In the prescribed format before the last date specified by the Department Post Graduate Review Committee (DPGRC). The DPGRC shall evaluate their submitted reports and oral presentations.

P14SE302 DISSERTATION

Class : **M.Tech III Semester** **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
-	-	-	8

Examination Scheme :

Continuous Internal Evaluation	100 marks
End Semester Exam	--

Guidelines for Dissertation:

Dissertation shall be normally conducted in two stages, spread over two sequential semesters i.e. third and fourth semester.

Registration Seminar shall be arranged within four weeks after completion of the Industrial Training and Seminar in the 3rd semester. The registration seminar shall include a brief report and presentation focusing the identified topic, literature review, time schedule indicating the main tasks, and expected outcome.

Progress Seminar-I: At the end of first stage (third semester), student shall be required to submit a preliminary report of work done for evaluation to the project coordinator and present the same before the DPGRC. The Continuous Internal Evaluation (CIE) for the third semester is as follows:

Assessment	Weightage
Dissertation Supervisor Assessment	50%
DPGRC Assessment	50%
Total Weightage:	**100%**

P14SE401 DISSERTATION

M.Tech IV Semester **Specialization:** Software Engineering

Teaching Scheme :

L	T	P	C
-	-	-	12

Examination Scheme :

Continuous Internal Evaluation	100 marks
End Semester Exam	--

Guidelines for Dissertation:

Progress Seminar-II shall be arranged during the 6th week of IV semester.

Progress Seminar-III shall be arranged during the 15th week of IV semester.

Synopsis Seminar shall be arranged two weeks before the final thesis submission date. The student shall submit a synopsis report covering all the details of the works carried out duly signed by the dissertation supervisor.

At the end of second stage (fourth semester), student shall be required to submit two bound copies, one being for the department and other for the dissertation supervisor. The dissertation report shall be evaluated by the DPGRC and external examination shall be conducted on a pre-notified date. The dissertation evaluation for the fourth semester is as follows:

Assessment	Weightage
Dissertation Supervisor Assessment	20%
DPGRC Assessment	20%
ESE (Presentation & Viva-voce)	60%
Total Weightage:	**100%**