ACADEMIC REGULATIONS, COURSE STRUCTURE
AND DETAILED SYLLABUS

M.Tech (SOFTWARE ENGINEERING)

FOR
MASTER OF TECHNOLOGY TWO YEAR POST GRADUATE COURSE
(Applicable for the batches admitted from 2015-2016)

R15

ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)
SCHOOL OF ENGINEERING
Venkatapur, Ghatkesar, Hyderabad – 500088
Applicable for the students of M. Tech. (Regular) programme from the Academic Year 2015-16 and onwards

The M. Tech. Degree of Jawaharlal Nehru Technological University Hyderabad shall be conferred on candidates who are admitted to the programme and who fulfill all the requirements for the award of the Degree.

1.0 ELIGIBILITY FOR ADMISSIONS

Admission to the above programme shall be made subject to eligibility, qualification and specialization as prescribed by the University from time to time.

Admissions shall be made on the basis of merit/rank obtained by the candidates at the qualifying Entrance Test conducted by the University or on the basis of any other order of merit as approved by the University, subject to reservations as laid down by the Govt. from time to time.

2.0 AWARD OF M. Tech. DEGREE

2.1 A student shall be declared eligible for the award of the M. Tech. Degree, if he pursues a course of study in not less than two and not more than four academic years, failing which he shall forfeit his seat in M. Tech. programme.

2.2 The student shall register for all 88 credits and secure all the 88 credits.

2.3 The minimum instruction days in each semester are 90.

3.0 COURSES OF STUDY

The following specializations are offered at present for the M. Tech. programme of study.

1. CAD/CAM
2. Computer Networks and Information Security
3. Computer Science
4. Computer Science and Engineering
5. Construction Management
6. Electrical Power Systems
7. Electronics and Communication Engineering
8. Embedded Systems
9. Machine Design
10. Power Electronics and Electrical Drives
11. Software Engineering
12. Structural Engineering
13. VLSI System Design
14. Wireless and Mobile Communication
4 Course Registration

4.1 A ‘Faculty Advisor or Counselor’ shall be assigned to each student, who will advise him on the Post Graduate Programme (PGP), its Course Structure and Curriculum, Choice/Option for Subjects/ Courses, based on his competence, progress, pre-requisites and interest.

4.2 Academic Section of the College invites ‘Registration Forms’ from students with in 15 days from the commencement of class work through ‘ON-LINE SUBMISSIONS’, ensuring ‘DATE and TIME Stamping’. The ON-LINE Registration Requests for any ‘CURRENT SEMESTER’ shall be completed BEFORE the commencement of SEE (Semester End Examinations) of the ‘PRECEDING SEMESTER’.

4.3 A Student can apply for ON-LINE Registration, ONLY AFTER obtaining the ‘WRITTEN APPROVAL’ from his Faculty Advisor, which should be submitted to the College Academic Section through the Head of Department (a copy of it being retained with Head of Department, Faculty Advisor and the Student).

4.4 If the Student submits ambiguous choices or multiple options or erroneous entries during ON-LINE Registration for the Subject(s) / Course(s) under a given/ specified Course Group/ Category as listed in the Course Structure, only the first mentioned Subject/ Course in that Category will be taken into consideration.

4.5 Subject/ Course Options exercised through ON-LINE Registration are final and CANNOT be changed, nor can they be inter-changed; further, alternate choices will also not be considered. However, if the Subject/ Course that has already been listed for Registration (by the Head of Department) in a Semester could not be offered due to any unforeseen or unexpected reasons, then the Student shall be allowed to have alternate choice - either for a new Subject (subject to offering of such a Subject), or for another existing Subject (subject to availability of seats), which may be considered. Such alternate arrangements will be made by the Head of Department, with due notification and time-framed schedule, within the FIRST WEEK from the commencement of Class-work for that Semester.

5 ATTENDANCE

The programmes are offered on a unit basis with each subject being considered a unit.

5.1 Attendance in all classes (Lectures/Laboratories etc.) is compulsory. The minimum required attendance in each theory / Laboratory etc. is 75% including the days of attendance in sports, games, NCC and NSS activities for appearing for the End Semester examination. A student shall not be permitted to appear for the Semester End Examinations (SEE) if attendance is less than 75%.

5.2 Condonation of shortage of attendance in each subject up to 10% (65% and above and below75%) in each semester shall be granted by the College Academic Committee on genuine medical grounds and valid reasons on representation by the candidate with supporting evidence.
5.3 Shortage of Attendance below 65% in each subject shall not be condoned.

5.4 Students whose shortage of attendance is not condoned in any subject are not eligible to write their end semester examination of that subject and their registration shall stand cancelled.

5.5 A prescribed fees hall be payable towards condonation of shortage of attendance.

5.6 A candidate shall get minimum required attendance at least in three (3) theory subjects in the present semester to get promoted to the next semester. In order to qualify for the award of the M.Tech Degree, The candidate shall complete all the academic requirements of the subjects, as per the course structure.

5.7 A student shall not be promoted to the next semester unless he satisfies the attendance requirement of the present Semester, as applicable. They may seek readmission into that semester when offered next. If any candidate fulfills the attendance requirement in the present semester, he shall not be eligible for readmission in to the same class.

6 EVALUATION

The performance of the candidate in each semester shall be evaluated subject-wise, with a maximum of 100 marks for theory and 100 marks for practicals, on the basis of Internal Evaluation and End Semester Examination.

6.1 For the theory subjects 60 marks shall be awarded for the performance in the Semester End Examination and 40 marks shall be awarded for Continuous Internal Evaluation (CIE). The Continuous Internal Evaluation shall be made based on the average of the marks secured in the two Mid Term-Examinations conducted, one in the middle of the Semester and the other, immediately after the completion of Semester instructions. Each mid-term examination shall be conducted for a total duration of 120 minutes with Part A as compulsory question (10 marks) consisting of 5 sub-questions carrying 2 marks each, and Part B with 3 questions to be answered out of 5 questions, each question carrying 10 marks. The details of the Question Paper pattern for End Examination (Theory) are given below:

- The Semester End Examination will be conducted for 60 marks. It consists of two parts. i). Part-A for 20 marks, ii). Part-B for 40 marks.

- Part-A is a compulsory question consisting of 5 questions, one from each unit and carries 4 marks each.

- Part-B to be answered 5 questions carrying 8 marks each. There will be two questions from each unit and only one should be answered.

6.2 For practical subjects, 60 marks shall be awarded for performance in the Semester End Examinations and 40 marks shall be awarded for day-to-day performance as Internal Marks.
6.3 The practical end semester examination shall be conducted with an external examiner and the laboratory teacher. The external examiner shall be appointed by the Principal from the panel of examiners recommended by Chairman, Board of Studies in respective Branches.

6.4 There shall be two seminar presentations during I year I semester and II semester. For seminar, a student under the supervision of a faculty member, shall collect the literature on a topic and critically review the literature and submit it to the department in a report form and shall make an oral presentation before the Departmental Academic Committee consisting of Head of the Department, Supervisor and two other senior faculty members of the department. For each Seminar there will be only internal evaluation of 50 marks. A candidate has to secure a minimum of 50% of marks to be declared successful. If he fails to fulfill minimum marks, he has to reappear during the supplementary examinations.

6.5 There shall be a Comprehensive Viva-Voce in II year I Semester. The Comprehensive Viva-Voce is intended to assess the students’ understanding of various subjects he has studied during the M. Tech. course of study. The Head of the Department shall be associated with the conduct of the Comprehensive Viva-Voce through a Committee. The Committee consisting of Head of the Department, one senior faculty member and an external examiner. The external examiner shall be appointed by the Principal from the panel of 3 examiners recommended by Chairman, Board of Studies in respective Branches. There are no internal marks for the Comprehensive Viva-Voce and evaluates for maximum of 100 marks. A candidate has to secure a minimum of 50% of marks to be declared successful. If he fails to fulfill minimum marks, he has to reappear during the supplementary examinations.

6.6 A candidate shall be deemed to have secured the minimum academic requirement in a subject if he secures a minimum of 40% of marks in the Semester End Examination and a minimum aggregate of 50% of the total marks in the Semester End Examination and Continuous Internal Evaluation taken together.

6.7 In case the candidate does not secure the minimum academic requirement in any subject (as specified in 6.6) he has to reappear for the Semester End Examination in that subject.

6.8 A candidate shall be given one chance to re-register for the subjects if the internal marks secured by a candidate is less than 50% and failed in that subject for maximum of two subjects and should register within four weeks of commencement of the class work. In such a case, the candidate must re-register for the subjects and secure the required minimum attendance. The candidate’s attendance in the re-registered subject(s) shall be calculated separately to decide upon his eligibility for writing the Semester End Examination in those subjects. In the event of the student taking another chance, his Continuous Internal Evaluation (internal) marks and Semester End Examination marks obtained in the previous attempt stands cancelled.

6.9 In case the candidate secures less than the required attendance in any subject, he shall not be permitted to write the Semester End Examination in that subject. He shall re-register for the subject when next offered.
7 Examinations and Assessment - The Grading System

7.1 Marks will be awarded to indicate the performance of each student in each Theory Subject, or Lab/Practicals, or Seminar, or Project, etc., based on the % marks obtained in CIE + SEE (Continuous Internal Evaluation + Semester End Examination, both taken together) as specified in Item 6 above, and a corresponding Letter Grade shall be given.

7.2 As a measure of the student’s performance, a 10-point Absolute Grading System using the following Letter Grades (UGC Guidelines) and corresponding percentage of marks shall be followed:

<table>
<thead>
<tr>
<th>% of Marks Secured (Class Intervals)</th>
<th>Letter Grade (UGC Guidelines)</th>
<th>Grade Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>80% and above (≥ 80%, ≤ 100%)</td>
<td>O (Outstanding)</td>
<td>10</td>
</tr>
<tr>
<td>Below 80% but not less than 70%</td>
<td>A+ (Excellent)</td>
<td>9</td>
</tr>
<tr>
<td>(≥ 70%, &lt; 80%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Below 70% but not less than 60%</td>
<td>A (Very Good)</td>
<td>8</td>
</tr>
<tr>
<td>(≥ 60%, &lt; 70%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Below 60% but not less than 55%</td>
<td>B+ (Good)</td>
<td>7</td>
</tr>
<tr>
<td>(≥ 55%, &lt; 60%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Below 55% but not less than 50%</td>
<td>B (Above Average)</td>
<td>6</td>
</tr>
<tr>
<td>(≥ 50%, &lt; 55%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Below 50% (≤ 50%)</td>
<td>F (Fail)</td>
<td>0</td>
</tr>
<tr>
<td>Absent</td>
<td>Ab</td>
<td>0</td>
</tr>
</tbody>
</table>

7.3 A student obtaining F Grade in any Subject shall be considered ‘failed’ and is be required to reappear as ‘Supplementary Candidate’ in the Semester End Examination (SEE), as and when offered. In such cases, his Internal Marks (CIE Marks) in those Subjects will remain the same as those he obtained earlier.

7.4 A student not appeared for examination then ‘Ab’ Grade will be allocated in any Subject shall be considered ‘failed’ and will be required to reappear as ‘Supplementary Candidate’ in the Semester End Examination (SEE), as and when offered.

7.5 A Letter Grade does not imply any specific Marks percentage and it will be the range of marks percentage.

1. In general, a student shall not be permitted to repeat any Subject/ Course (s) only for the sake of ‘Grade Improvement’ or ‘SGPA/ CGPA Improvement’. 
2. A student earns Grade Point (GP) in each Subject/ Course, on the basis of the Letter Grade obtained by him in that Subject/ Course. The corresponding ‘Credit Points’ (CP) are computed by multiplying the Grade Point with Credits for that particular Subject/ Course.

\[
\text{Credit Points (CP)} = \text{Grade Point (GP)} \times \text{Credits} \quad \text{.... For a Course}
\]

3. The Student passes the Subject/ Course only when he gets \( \text{GP} \geq 6 \) (B Grade or above).

4. The Semester Grade Point Average (SGPA) is calculated by dividing the Sum of Credit Points (ΣCP) secured from ALL Subjects/ Courses registered in a Semester, by the Total Number of Credits registered during that Semester. SGPA is rounded off to TWO Decimal Places. SGPA is thus computed as

\[
\text{SGPA} = \left\{ \frac{\sum_{i=1}^{N} C_i G_i}{\sum_{i=1}^{N} C_i} \right\} \quad \text{For each Semester,}
\]

where ‘i’ is the Subject indicator index (takes into account all Subjects in a Semester), ‘N’ is the no. of Subjects ‘REGISTERED’ for the Semester (as specifically required and listed under the Course Structure of the parent Department), C is the no. of Credits allotted to the ith Subject, and G represents the Grade Points (GP) corresponding to the Letter Grade awarded for that ith Subject.

5. The Cumulative Grade Point Average (CGPA) is a measure of the overall cumulative performance of a student over all Semesters considered for registration. The CGPA is the ratio of the Total Credit Points secured by a student in ALL registered Courses in ALL Semesters, and the Total Number of Credits registered in ALL the Semesters. CGPA is rounded off to TWO Decimal Places. CGPA is thus computed from the I Year Second Semester onwards, at the end of each Semester, as per the formula

\[
\text{CGPA} = \left\{ \frac{\sum_{j=1}^{M} C_j G_j}{\sum_{j=1}^{M} C_j} \right\} \quad \text{... for all S Semesters registered}
\]

(ie., upto and inclusive of S Semesters, \( S \geq 2 \), where ‘M’ is the TOTAL no. of Subjects (as specifically required and listed under the Course Structure of the parent Department) the Student has ‘REGISTERED’ from the 1st Semester onwards upto and inclusive of the Semester S (obviously \( M > N \)), ‘j’ is the Subject indicator index (takes into account all Subjects from 1 to S Semesters), C is the no. of Credits allotted to the jth Subject, and G represents the Grade Points (GP) corresponding to the Letter Grade awarded for that jth Subject. After registration and completion of I Year I Semester however, the SGPA of that Semester itself may be taken as the CGPA, as there are no cumulative effects.

7.11 For Calculations listed in Item 7.6 – 7.10, performance in failed Subjects/ Courses (securing F Grade) will also be taken into account, and the Credits of such Subjects/ Courses will also be included in the multiplications and summations.
8. **EVALUATION OF PROJECT/DISSERTATION WORK**

Every candidate shall be required to submit a thesis or dissertation on a topic approved by the Project Review Committee.

8.1 A Project Review Committee (PRC) shall be constituted with Head of the Department as Chairperson, Project Supervisor and one senior faculty member of the Departments offering the M. Tech. programme.

8.2 Registration of Project Work: A candidate is permitted to register for the project work after satisfying the attendance requirement of all the subjects, both theory and practical.

8.3 After satisfying 8.2, a candidate has to submit, in consultation with his Project Supervisor, the title, objective and plan of action of his project work to the PRC for approval. Only after obtaining the approval of the PRC the student can initiate the Project work.

8.4 If a candidate wishes to change his supervisor or topic of the project, he can do so with the approval of the PRC. However, the PRC shall examine whether or not the change of topic/supervisor leads to a major change of his initial plans of project proposal. If yes, his date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.

8.5 A candidate shall submit his project status report in two stages at least with a gap of 3 months between them.

8.6 The work on the project shall be initiated at the beginning of the II year and the duration of the project is two semesters. A candidate is permitted to submit Project Thesis only after successful completion of all theory and practical courses with the approval of PRC not earlier than 40 weeks from the date of registration of the project work. For the approval of PRC the candidate shall submit the draft copy of thesis to the Head of the Department and make an oral presentation before the PRC.

8.7 Three copies of the Project Thesis certified by the supervisor shall be submitted to the College/School/Institute.

8.8 For Project work **Review I** in II Year I Sem. there is an internal marks of 50, the evaluation should be done by the PRC for 25 marks and Supervisor will evaluate for 25 marks. The Supervisor and PRC will examine the Problem Definition, Objectives, Scope of Work, Literature Survey in the same domain. A candidate has to secure a minimum of 50% of marks to be declared successful for Project Work Review I. If he fails to fulfill minimum marks, he has to reappear as per the recommendations of PRC.

8.9 For Project work **Review II** in II Year II Sem. there is an internal marks of 50, the evaluation should be done by the PRC for 25 marks and Supervisor will evaluate for 25 marks. The PRC will examine the overall progress of the Project Work and decide the Project is eligible for final submission or not. A candidate has to secure a minimum of 50% of marks to be declared successful for Project Work Review II. If he fails to fulfill minimum marks, he has to reappear as per the recommendations of PRC.
8.10 For Project Evaluation (Viva Voce) in II Year II Sem. there is an external marks of 150 and the same evaluated by the External examiner appointed by the Institution. The candidate has to secure minimum of 50% marks in Project Evaluation (Viva-Voce) examination.

8.11 If he fails to fulfill as specified in 8.10, he will reappear for the Viva-Voce examination only after three months. In the reappeared examination also, fails to fulfill, he will not be eligible for the award of the degree.

8.12 The thesis shall be adjudicated by one examiner selected by the Institution. For this, Chairmen, BOS of the respective departments shall submit a panel of 3 examiners, who are eminent in that field with the help of the concerned guide and senior faculty of the department.

8.13 If the report of the examiner is not favourable, the candidate shall revise and resubmit the Thesis. If the report of the examiner is unfavourable again, the thesis shall be summarily rejected.

8.14 If the report of the examiner is favourable, Project Viva-Voce examination shall be conducted by a board consisting of the Supervisor, Head of the Department and the external examiner who adjudicated the Thesis.

8.15 The Head of the Department shall coordinate and make arrangements for the conduct of Project Viva-Voce examination.

9. **AWARD OF DEGREE AND CLASS**

9.1 A Student who registers for all the specified Subjects/Courses as listed in the Course Structure, satisfies all the Course Requirements, and passes the examinations prescribed in the entire PG Programme (PGP), and secures the required number of 88 Credits (with CGPA ≥ 6.0), shall be declared to have ‘QUALIFIED’ for the award of the M.Tech. Degree in the chosen Branch of Engineering and Technology with specialization as he admitted.

9.2 **Award of Class**

After a student has satisfied the requirements prescribed for the completion of the programme and is eligible for the award of M. Tech. Degree, he shall be placed in one of the following three classes based on the CGPA:

<table>
<thead>
<tr>
<th>Class Awarded</th>
<th>CGPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Class with Distinction</td>
<td>≥ 7.75</td>
</tr>
<tr>
<td>First Class</td>
<td>6.75 ≤ CGPA &lt; 7.75</td>
</tr>
<tr>
<td>Second Class</td>
<td>6.00 ≤ CGPA &lt; 6.75</td>
</tr>
</tbody>
</table>

9.3 A student with final CGPA (at the end of the PGP) < 6.00 will not be eligible for the Award of Degree.
10. **WITHHOLDING OF RESULTS**

If the student has not paid the dues, if any, to the institution or if any case of indiscipline is pending against him, the result of the student will be withheld and he will not be allowed into the next semester. His degree will be withheld in such cases.

11. **TRANSITORY REGULATIONS**

11.1 If any candidate is detained due to shortage of attendance in one or more subjects, they are eligible for re-registration to maximum of two earlier or equivalent subjects at a time as and when offered.

11.2 The candidate who fails in any subject will be given two chances to pass the same subject; otherwise, he has to identify an equivalent subject as per R15 Academic Regulations.

12. **GENERAL**

12.1 **Credit**: A unit by which the course work is measured. It determines the number of hours of instructions required per week. One credit is equivalent to one hour of teaching (lecture or tutorial) or two hours of practical work/field work per week.

12.2 **Credit Point**: It is the product of grade point and number of credits for a course.

12.3 Wherever the words “he”, “him”, “his”, occur in the regulations, they include “she”, “her”.

12.4 The academic regulation should be read as a whole for the purpose of any interpretation.

12.5 In the case of any doubt or ambiguity in the interpretation of the above rules, the Decision of the Academic Council is final.

12.6 The Academic Council may change or amend the academic regulations or syllabi at any time and the changes or amendments made shall be applicable to all the students with effect from the dates notified by the Academic Council.
## MALPRACTICES RULES
### DISCIPLINARY ACTION FOR IMPROPER CONDUCT IN EXAMINATIONS

<table>
<thead>
<tr>
<th>Nature of Malpractices/Improper conduct</th>
<th>Punishment</th>
</tr>
</thead>
<tbody>
<tr>
<td>If the candidate:</td>
<td></td>
</tr>
<tr>
<td>1. (a) Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, cell phones, pager, palm, computers or any other form of material concerned with or related to the subject of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the candidate which can be used as an aid in the subject of the examination)</td>
<td>Expulsion from the examination hall and cancellation of the performance in that subject only</td>
</tr>
<tr>
<td>(b) Gives assistance or guidance or receives it from any other candidate orally or by any other body language methods or communicates through cell phones with any candidate or persons in or outside the exam hall in respect of any matter.</td>
<td>Expulsion from the examination hall and cancellation of the performance in that subject only of all the candidates involved. In case of an outsider, he will be handed over to the police and a case is registered against him.</td>
</tr>
<tr>
<td>2. Has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the subject of the examination (theory or practical) in which the candidate is appearing.</td>
<td>Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidates has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The hall ticket of the candidate is to be cancelled and sent to the controller of examinations, AGI.</td>
</tr>
<tr>
<td>3. Impersonates any other candidate in connection with the examination.</td>
<td>The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred and forfeits the seat. The performance of the original candidate who has been impersonated, shall be cancelled in all the</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>4.</td>
<td>Smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination.</td>
</tr>
<tr>
<td>5.</td>
<td>Uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks.</td>
</tr>
<tr>
<td>6.</td>
<td>Refuses to obey the orders of the Chief Superintendent/Assistant-Superintendent/any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the officer-in-charge or any person on duty in or outside the examination hall of any injury to his person or to any office relations whether by words, either spoken or written or by signs or by visible representation.</td>
</tr>
</tbody>
</table>
assaults the officer-in-charge, or any person on duty in or outside the examination hall or any of his relations, or indulges in any other act of misconduct or mischief which result in damage to or destruction of property in the examination hall or any part of the college campus or engages in any other act which in the opinion of the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.

<p>| 7. | Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall. | Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidates has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all semester examinations. The continuation of the course by the candidate is subject to the academic regulation in connection with forfeiture of seat. |
| 8. | Posses any lethal weapon or firearm in the examination hall. | Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidates has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat. |
| 9. | If student of the college, who is not a candidate for the particular examination or any person not connected with college indulges in any malpractice or improper conduct mentioned in clause 6 to 8 | Student of the college’s expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidates has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>also debarred and forfeiture the seat. Person(s) who do not belong to the College will be handed over to police and, a police case will be registered against them.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.</td>
<td>Comes in a drunken condition to the examination hall.</td>
<td>Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidates has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year.</td>
</tr>
<tr>
<td>11.</td>
<td>Copying detected on the basis of internal evidence, such as, during valuation or during special scrutiny.</td>
<td>Cancellation of the performance in that subject and all other subjects the candidate has appeared including practical examinations and project work of the semester/year examinations.</td>
</tr>
<tr>
<td>12.</td>
<td>If any malpractice is detected which is not covered in the above clauses 1 to 11 shall be reported to the Malpractices committee, AGI for further action to award suitable punishment.</td>
<td></td>
</tr>
<tr>
<td>Subject Code</td>
<td>Subject Name</td>
<td>Internal marks</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Core Course I</td>
<td>Advanced Data Structures and Algorithms</td>
<td>40</td>
</tr>
<tr>
<td>Core Course II</td>
<td>Web Technologies and Services</td>
<td>40</td>
</tr>
<tr>
<td>Core Course III</td>
<td>Software Requirements and Estimation</td>
<td>40</td>
</tr>
<tr>
<td>Open Elective I</td>
<td>Open Elective I</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>1. Software Development Methodologies</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Social Media Intelligence</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Entrepreneurship</td>
<td></td>
</tr>
<tr>
<td>Core Elective I</td>
<td>Core Elective I</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>1. Secure Software Engineering</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Cyber Security and Cyber Laws</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Information Security and Audit</td>
<td></td>
</tr>
<tr>
<td>Core Elective II</td>
<td>Core Elective II</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>1. Parallel Algorithms</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Advanced Data Mining</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Object Oriented Modeling</td>
<td></td>
</tr>
<tr>
<td>Laboratory I</td>
<td>Web Technologies and Services Lab</td>
<td>40</td>
</tr>
<tr>
<td>Seminar I</td>
<td>Seminar</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td></td>
</tr>
<tr>
<td>Subject Code</td>
<td>Subject Name</td>
<td>Internal Marks</td>
</tr>
<tr>
<td>----------------</td>
<td>--------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Core Course V</td>
<td>Software Architecture and Design Patterns</td>
<td>40</td>
</tr>
<tr>
<td>Core Course VI</td>
<td>Software Process and Project Management</td>
<td>40</td>
</tr>
<tr>
<td>Core Course VII</td>
<td>Software Quality Assurance and Testing</td>
<td>40</td>
</tr>
<tr>
<td>Open Elective II</td>
<td>Open Elective II</td>
<td>40</td>
</tr>
<tr>
<td>Core Elective III</td>
<td>Core Elective –III</td>
<td>40</td>
</tr>
<tr>
<td>Core Elective IV</td>
<td>Core Elective –IV</td>
<td>40</td>
</tr>
<tr>
<td>Laboratory II</td>
<td>Software Testing Lab</td>
<td>40</td>
</tr>
<tr>
<td>Seminar II</td>
<td>Seminar</td>
<td>50</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## II Year I Semester

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject Name</th>
<th>L</th>
<th>T/P/D</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Comprehensive Viva</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Project Work Review I</td>
<td>-</td>
<td>24</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td>-</td>
<td>24</td>
<td>16</td>
</tr>
</tbody>
</table>

## II Year II Semester

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject Name</th>
<th>L</th>
<th>T/P/D</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Project Work Review II</td>
<td></td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Project Evaluation (Viva Voce)</td>
<td>-</td>
<td>16</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td>-</td>
<td>24</td>
<td>16</td>
</tr>
</tbody>
</table>
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – I Sem.  L  P  C
4  0  4

ADVANCED DATA STRUCTURES AND ALGORITHM

Objectives:
The fundamental design, analysis, and implementation of basic data structures.
Basic concepts in the specification and analysis of programs.
- Principles for good program design, especially the uses of data abstraction.
- Significance of algorithms in the computer field
- Various aspects of algorithm development
- Qualities of a good solution

UNIT I
Algorithms, Performance analysis- time complexity and space complexity, Asymptotic
Notation-Big Oh, Omega and Theta notations, Complexity Analysis Examples.
Data structures-Linear and non linear data structures, ADT concept, Linear List ADT,
Array representation, Linked representation, Vector representation, singly linked lists -
insertion, deletion, search operations, doubly linked lists-insertion, deletion operations,
circular lists.
Representation of single, two dimensional arrays, Sparse matrices and their representation.

UNIT II
Stack and Queue ADTs, array and linked list representations, infix to postfix conversion
using stack, implementation of recursion, Circular queue-insertion and deletion, Dequeue
ADT, array and linked list representations, Priority queue ADT, implementation using
Heaps, Insertion into a Max Heap, Deletion from a Max Heap, java.util package-ArrayList,
LinkedList, Vector classes, Stacks and Queues in java.util, Iterators in java.util.

UNIT III
Searching–Linear and binary search methods, Hashing-Hash functions, Collision
Resolution methods-Open Addressing, Chaining, Hashing in java.util-HashMap, HashSet,
Hashtable.
Sorting –Bubble sort, Insertion sort, Quick sort, Merge sort, Heap sort, Radix sort,
comparison of sorting methods.
UNIT IV
Trees- Ordinary and Binary trees terminology, Properties of Binary trees, Binary tree ADT, representations, recursive and non recursive traversals, Java code for traversals, threaded binary trees.

Graphs- Graphs terminology, Graph ADT, representations, graph traversals/search methods-DFS and BFS, Java code for graph traversals, Applications of Graphs-Minimum cost spanning tree using Kruskal’s algorithm, Dijkstra’s algorithm for Single Source Shortest Path Problem.

UNIT V
Search trees- Binary search tree-Binary search tree ADT ,insertion, deletion and searching operations, Balanced search trees, AVL trees-Definition and examples only, Red Black trees – Definition and examples only, B-Trees-definition, insertion and searching operations, Trees in java.util-TreeSet, TreeMap Classes, Tries(examples only),Comparison of Search trees.

Text compression-Huffman coding and decoding, Pattern matching-KMP algorithm.

TEXT BOOKS:
1. Data structures, Algorithms and Applications in Java, S.Sahni, Universities Press.

REFERENCE BOOKS:
1. Java for Programmers, Deitel and Deitel, Pearson education.
3. java: The Complete Reference, 8th edition, Herbert Schildt, TMH.
6. Classic Data structures in Java, T.Budd, Addison-Wesley (Pearson Education).
7. Data structures with Java, Ford and Topp, Pearson Education.
10. Data structures and Software Development in an Object-Oriented Domain,
WEB TECHNOLOGIES AND SERVICES

Objectives:

- The student who has knowledge of programming with java should be able to develop web based solutions using multi-tier architecture. She / He should have good understanding of different technologies on client and server side components as follows:
  - Client Side: HTML, CSS, Javascript, Ajax, JQuery and JSON
  - Server Side: Servlets, JSP
  - Database: MySQL with Hibernate and Connection Pooling
  - Framework: Struts with validation framework, Internationalization (I18N)
  - SOA: Service Oriented Architecture, Web services fundamentals, Axis framework for WS

UNIT I

Client Side Technologies:
Overview of HTML - Common tags for text formatting, Lists, Tables, Images, Forms, Frames etc., XHTML Cascading Style sheets, linking to HTML Pages, Classes in CSS, General CSS statements for Text, Table, List and Page formatting
Introduction to JavaScripts, variables, arrays, methods and string manipulation, BOM/DOM (Browser/Document Object Model), accessing elements by ID, Objects in JavaScript
Dynamic HTML with JavaScript and with CSS, form validation with JavaScript, Handling Timer Events, JQuery

UNIT II: Introduction to Java Servlets:
Introduction to Servlets: Lifecycle of a Servlet, Reading request and initialization parameters, Writing output to response, MIME types in response, Session Tracking: Using Cookies and Sessions Steps involved in Deploying an application Database Access with JDBC and Connection Pooling Introduction to XML, XML Parsing with DOM and SAX Parsers in Java

Ajax - Ajax programming with JSP/Servlets, creating XML Http Object for various browsers, Sending request, Processing response data and displaying

UNIT III: Introduction to JSP:
JSP Application Development: Types of JSP Constructs (Directives, Declarations, Expressions, Code Snippets), Generating Dynamic Content, Exception Handling, Implicit JSP Objects, Conditional Processing, Sharing Data Between JSP pages, Sharing Session
and Application Data, Using user defined classes with jsp:useBean tag, Accessing a Database from a JSP

UNIT IV: Introduction to Struts Framework:
Introduction to MVC architecture, Anatomy of a simple struts application, struts configuration file, Presentation layer with JSP, JSP bean, html and logic tag libraries, Struts Controller class, Using formdata in Actions, Page Forwarding, validation frame work, Internationalization

UNIT V: Service Oriented Architecture and Web Services
Overview of Service Oriented Architecture – SOA concepts, Key Service Characteristics, Technical Benefits of a SOA
Introduction to Web Services– The definition of web services, basic operational model of web services, basic steps of implementing web services.
Core fundamentals of SOAP – SOAP Message Structure, SOAP encoding, SOAP message exchange models Describing Web Services –Web Services life cycle, anatomy of WSDL
Introduction to Axis– Installing axis web service framework, deploying a java web service on axis.
Web Services Interoperability – Creating java and .Net client applications for an Axis Web Service

TEXT BOOKS:
2. The complete Reference Java 7th Edition , Herbert Schidt., TMH.

REFERENCE BOOKS:
2. Core SERVLETS ANDJAVA SERVER PAGES VOLUME 1: CORE
3. TECHNOLOGIES , Marty Hall and Larry Brown Pearson
4. Internet and World Wide Web – How to program, Dietel and Nieto PHI/Pearson.
7. Java Server Programming, Ivan Bayross and others, The X Team, SPD
9. Beginning Web Programming, Jon Duckett, WROX.
11. Java Script, D. Flanagan, O’Reilly, SPD.
OBJECTIVES:

- Students will demonstrate knowledge of the distinction between critical and non-critical systems.
- Students will demonstrate the ability to manage a project including planning, scheduling and risk assessment/management.
- Students will author a software requirements document.
- Students will demonstrate an understanding of the proper contents of a software requirements document.
- Students will author a formal specification for a software system.
- Students will demonstrate an understanding of distributed system architectures and application architectures.
- Students will demonstrate an understanding of the differences between real-time and non-real time systems.
- Students will demonstrate proficiency in rapid software development techniques.
- Students will demonstrate proficiency in software development cost estimation.
- Students will author a software testing plan.

UNIT I

Software Requirements: What and Why

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

Software Requirements Engineering

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

UNIT II

Software Requirements Management

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, Dataflow diagram, state transition diagram, class diagrams, Object analysis, Problem Frames

**UNIT III**

**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**

**Effort, Schedule and Cost Estimation**

What is Productivity? Estimation Factors, Approaches to Effort and Schedule Estimation, COCOMO II, Putnam Estimation Model, Algorithmic models, Cost Estimation

**UNIT V**

**Tools for Requirements Management and Estimation**


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

**TEXT BOOK:**


**REFERENCE BOOKS:**

1. Software Requirements by Karl E. Weigers, Microsoft Press.
Objectives:

Your studies will enable you to develop:

- a broad and critical understanding of all the processes for engineering high quality software and the principles, concepts and techniques associated with software development
- an ability to analyze and evaluate problems and draw on the theoretical and technical knowledge to develop solutions and systems
- a range of skills focused on the analysis of requirements, design and implementation of reliable and maintainable software, with strong emphasis on engineering principles applied over the whole development lifecycle
- an awareness of current research in software development, the analytical skills and research techniques for their critical and independent evaluation and their application to new problems.

UNIT I


Process models: The waterfall model, Incremental process models, Evolutionary process models, specialized process models, The Unified process.

UNIT II

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

System models: Context Models, Behavioral models, Data models, Object models, structured methods.

UNIT III

Design Engineering: Design process and Design quality, Design concepts, the design model, pattern based software design.
Creating an architectural design: software architecture, Data design, Architectural styles and patterns, Architectural Design, assessing alternative architectural designs, mapping data flow into a software architecture.

Modeling component-level design: Designing class-based components, conducting component-level design, Object constraint language, designing conventional components.

Performing User interface design: Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation.

UNIT IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging. Product metrics: Software Quality, Frame work for Product metrics, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance. Metrics for Process and Products: Software Measurement, Metrics for software quality.

UNIT V

Risk management: Reactive vs Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan.


Configuration Management: Configuration Management planning, Change management, Version and release management, System building, CASE tools for configuration management.

TEXT BOOKS:

REFERENCE BOOKS:
3 Fundamentals of Software Engineering, Rajib Mall,PHI, 2005


10 Introduction to Software Engineering, R. J. Leach, CRC Press.

11 Software Engineering Fundamentals, Ali Behforooz and Frederick J. Hudson, Oxford University Press, rp2009

UNIT – I

The Beginnings of Social Media Intelligence: What is Social Media monitoring? Anecdotal referencing of Social Media Comments, Text Mining, Some Simple Metrics, Using Social Media as Early Warning System. Fundamental of Opinion Formation: Affecting Opinion versus Biasing Expression, How Do We Form Opinions?, How Do Expectations Affect Opinion?, How Do Expertise and Knowledge Influence How We Form Opinions?, Opinion Formation in a Social Context, Bandwagon behavior and Information Cascades, Implications for Social Media Intelligence

UNIT – II


UNIT – III


UNIT – IV

Managing Social Media Communities for Better Social Media Intelligence: Creating an Inviting Environment, The Benefits of a Well-Managed Opinion Community (and the Costs of Not Managing the Community at All) Quality of Intelligence Depends on the Quality of the Opinion Community, Creating and Manipulating Buzz, Buzz Campaign or Fraud?, Identifying Fraudulent Opinions Cutting Through the Online Chatter : A New Paradigm for Marketing Research, Measure What Matters, Cast a Wide Net, Analyze the Text, Understand the biases, Establish Links to Performance metrics.

UNIT – V

Intelligence Integration : Overview of Marketing Research Methods, Using Social Media for Marketing research, Tracking Brand Health, Understanding Market Structure, Social Shopping, Integration with Data from Other Parts of the Organization, Intelligence

TEXT BOOK:
Course Outcomes:

Student will be able to:

1. Have the knowledge on various concepts of business management and approaches.
2. Understand and analyze the interconnections between the development of key functional areas of business organization and the management thought process.
3. Acquire team management skills and to become a competent leader, who possesses complex and integrated real world skills.
4. Be ethically conscious and socially responsible managers, capable of contributing to the development of the nation and quality of life.


Case 3: From candle seller to CEO (Arya Kumar P.No. 48)


Case : Globalizing Local Talent, (B. Janakiram, M. Rizwana, page 228).


Case 1: Water, Water everywhere: but not a drop to drink, (Richard Blundel , Page 48).


Case 3: Pets.com (Arya Kumar P.No. 88)

Case 4: creativity in start-ups (Arya Kumar P.No. 166)

Case 5: Opportunity – Earthmoving Industry (Arya Kumar P.No. 211)

Case 1: Victoria, Tomlinson; Network. (Richard Blundel, Page 99).
Case 2: Tim Lockett, Knowing your Customers & Suppliers (Richard Blundel Page128).
Case 3: Google (Arya Kumar P.No. 248)
Case 4: Tata Motors – Nano (Arya Kumar P.No. 279)

Unit-V: Strategic perspectives in entrepreneurship- Strategic planning-Strategic actions-strategic positioning-Business stabilization- Building the adaptive firms-Understanding the growth stage-Unique managerial concern of growing ventures.

Case 1:To Lease or Not: A Cash flow Question (David H.Holt, Page 452).
Case 2:-Public Sector - address seed capital (David H.Holt, Page 453).

Text Books:
1. D F Kuratko and T V Rao “Entrepreneurship- A South-Asian Perspective “Cengage Learning, 2012. (For PPT, Case Solutions Faculty may visit : login.cengage.com) Cases:

Reference Books:
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)


SECURE SOFTWARE ENGINEERING
(ELECTIVE-I)

Objectives:

• Students will demonstrate knowledge of the distinction between critical and non-critical systems.
• Students will demonstrate the ability to manage a project including planning, scheduling and risk assessment/management.
• Students will author a software requirements document.
• Students will demonstrate an understanding of the proper contents of a software requirements document.
• Students will author a formal specification for a software system.
• Students will demonstrate an understanding of distributed system architectures and application architectures.
• Students will demonstrate an understanding of the differences between real-time and non-real time systems.
• Students will demonstrate proficiency in rapid software development techniques.
• Students will be able to identify specific components of a software design that can be targeted for reuse.
• Students will demonstrate proficiency in software development cost estimation.
• Students will author a software testing plan.

UNIT – I


UNIT – II

Requirements Engineering for secure software: Introduction, the SQUARE process Model, Requirements elicitation and prioritization

UNIT – III

Secure Software Architecture and Design: Introduction, software security practices for architecture and design: architectural risk analysis, software security knowledge for
architecture and design: security principles, security guidelines and attack patterns

**Secure coding and Testing:** Code analysis, Software Security testing, Security testing considerations throughout the SDLC

**UNIT – IV**

**Security and Complexity:** System Assembly Challenges: introduction, security failures, functional and attacker perspectives for security analysis, system complexity drivers and security

**UNIT – V**

**Governance and Managing for More Secure Software:** Governance and security, Adopting an enterprise software security framework, How much security is enough?, Security and project management, Maturity of Practice

**TEXT BOOK:**


**REFERENCE BOOKS:**

1. Developing Secure Software: Jason Grembi, Cengage Learning
Objectives:

- To learn Internet, E-commerce and E-governance with reference to Free Market Economy
- To learn International Efforts relating to Cyberspace laws and Cyber crimes
- To learn Law relating to electronic records and intellectual property rights in India
- To learn Penalties, Compensation and Offences under the Cyberspace and Internet in India
- To learn Miscellaneous provisions of IT Act and Conclusions

UNIT-I
Internet, E-commerce and E-governance with reference to Free Market Economy
Understanding Computers, Internet and Cyber laws, Conceptual Framework of E-commerce: E-governance, the role of Electronic Signatures in E-commerce with Reference to Free Market Economy in India.

UNIT-II
Law relating to electronic records and intellectual property rights in India
Legal aspects of Electronic records / Digital signatures, The roles and regulations of Certifying Authorities in India, Protection of Intellectual Property Rights in Cyberspace in India.

UNIT-III
International Efforts relating to Cyberspace laws and Cyber crimes
International efforts related to Cyber laws, Council of Europe (COE) convention on Cyber Crimes.

UNIT-IV
Penalties, Compensation and Offences under the Cyberspace and Internet in India
Penalties, Compensation and Adjunction of violations of provisions of IT Act and Judicial review, Some important offences under the Cyberspace law and the Internet in India, Other offences under the Information Technology Act in India.

UNIT-V: Miscellaneous provisions of IT Act and Conclusions
The role of Electronic Evidence and miscellaneous provisions of the IT Act.
TEXT BOOK:

REFERENCE BOOKS:

ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – I Sem.  L   P   C
4  0  4

INFORMATION SECURITY AND AUDIT
(ELECTIVE- I)

Objectives:
To introduce the fundamental concepts and techniques in computer and network security, giving students an overview of information security and auditing, and to expose students to the latest trend of computer attack and defense. Other advanced topics on information security such as mobile computing security, security and privacy of cloud computing, as well as secure information system development will also be discussed.

UNIT I
Public key cryptography principles, public key cryptography algorithms (RSA, Diffie-Hellman, ECC), public Key Distribution.

UNIT II
Approaches of Message Authentication, Secure Hash Functions (SHA-512, MD5) and HMAC, Digital Signatures, Kerberos, X.509 Directory Authentication Service,
Email Security: Pretty Good Privacy (PGP)

UNIT III
Web Security: Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).
Firewalls: Firewall Design principles, Trusted Systems, Intrusion Detection Systems

UNIT IV
Auditing For Security:

UNIT V
Auditing For Security:
Approaches to Audits, Technology Based Audits Vulnerability Scanning And Penetration Testing, Resistance to Security Audits, Phase in security audit, Security audit Engagement Costs and other aspects, Budgeting for security audits, Selecting external Security
Consultants, Key Success factors for security audits.

**TEXT BOOKS:**


**REFERENCE BOOKS:**

ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)


PARALLEL ALGORITHMS
(ELECTIVE-II)

Objectives:

- To understand the role of computation models in parallel computation;
- To understand the circuit and comparison network models;
- To design Parallel Matrix Transportation and Multiplication Algorithm
- To understand the PRAM and BSP models and their theoretical foundations;

UNIT-I:
Sequential model, need of alternative model, parallel computational models such as PRAM, LMCC, Hypercube, cube connected cycle, Butterfly, Perfect shuffle Computers, Tree model, Pyramid model, Fully Connected model, PRAM-CREW, EREW models, simulation of one model from another one.

UNIT-II:
Performance Measures of Parallel Algorithms, speed-up and efficiency of PA, Cost Optimality, Example to illustrate Cost-optimal algorithms- such as summation, Min/Max on various models.

UNIT-III:
Parallel Sorting Networks, Parallel Merging Algorithms on CREW/EREW/MCC/, Parallel Sorting Networks on CREW/EREW/MCC/, linear array

UNIT-IV:
Parallel Searching Algorithm, Kth element, Kth element in X+Y on PRAM, Parallel Matrix Transportation and Multiplication Algorithm on PRAM, MCC, Vector-Matrix Multiplication, Solution of Linear Equation, Root finding.

UNIT-V:
Graph Algorithms - Connected Graphs, search and traversal, Combinatorial Algorithms- Permutation, Combinations, Derangements.

TEXT BOOK:
REFERENCE BOOKS:


ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – I Sem. L P C 4 0 4

ADVANCED DATA MINING
(ELECTIVE-II)

Objectives:

- To develop the abilities of critical analysis to data mining systems and applications.
- To implement practical and theoretical understanding of the technologies for data mining
- To understand the strengths and limitations of various data mining models;

UNIT-I

Data mining Overview and Advanced Pattern Mining

Data mining tasks – mining frequent patterns, associations and correlations, classification and regression for predictive analysis, cluster analysis, outlier analysis; advanced pattern mining in multilevel, multidimensional space – mining multilevel associations, mining multidimensional associations, mining quantitative association rules, mining rare patterns and negative patterns.

UNIT-II

Advance Classification

Classification by back propagation, support vector machines, classification using frequent patterns, other classification methods – genetic algorithms, roughest approach, fuzzy set approach;

UNIT-III

Advance Clustering

Density-based methods – DBSCAN, OPTICS, DENCLUE; Grid-Based methods – STING, CLIQUE; Exception – maximization algorithm; clustering High-Dimensional Data; Clustering Graph and Network Data.

UNIT-IV

Web and Text Mining

Introduction, web mining, web content mining, web structure mining, web usage mining, Text mining – unstructured text, episode rule discovery for texts, hierarchy of categories, text clustering.

UNIT-V

Temporal and Spatial Data Mining

Introduction; 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. Data Mining Concepts and Techniques, Jiawei Hang Micheline Kamber, Jian pei, Morgan Kaufmannn.
2. Data Mining Techniques – Arun K pujari, Universities Press.

REFERENCE BOOKS:
1 Introduction to Data Mining – Pang-Ning Tan, Vipin kumar, Michael Steinbach, Pearson.
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – I Sem.  L  P  C
4  0  4

OBJECT ORIENTED MODELING
(ELECTIVE-II)

Objectives:

- Concisely define the following key terms: class, object, state, behavior, object class, class diagram, object diagram, operation, encapsulation, constructor operation, query operation, update operation, scope operation, association, association role, multiplicity, association class, abstract class, concrete class, class-scope attribute, abstract operation, method, polymorphism, overriding, multiple classification, aggregation, and composition.

- To describe the activities in the different phases of the object-oriented development life cycle.

- State the advantages of object-oriented modeling vis-à-vis structured approaches.

- Compare and contrast the object-oriented model with the E-R and EER models.

- Model a real-world application by using a UML class diagram.

- Provide a snapshot of the detailed state of a system at a point in time using a UML (Unified Modeling Language) object diagram.

- Recognize when to use generalization, aggregation, and composition relationships.

- Specify different types of business rules in a class diagram.

UNIT I

Introduction to UML: The meaning of Object Orientation, object identity, Encapsulation, information hiding, polymorphism, generosity, importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture.


Sequence Diagrams: Terms, concepts, depicting asynchronous messages with/without priority, callback mechanism, broadcast messages.
UNIT II

**Basic Behavioral Modeling:** Use cases, Use case Diagrams, Activity Diagrams.

**Advanced Behavioral Modeling:** Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

**Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams.

UNIT III

**The Unified process:** use case driven, architecture centric, iterative, and incremental **The Four Ps:** people, project, product, and process

**Use case driven process:** why use case, capturing use cases, analysis, design, and implementation to realize the use cases

**Architecture-centric process:** architecture in brief, why we need architecture, use cases and architecture, the steps to architecture, an architecture description.

UNIT IV

**Iterative incremental process:** iterative incremental in brief, why iterative incremental development? The iterative approach is risk driven, the generic iteration.

**The Generic Iteration workflow:** phases are the first division workflow, planning proceeds doing, risks affect project planning, use case prioritization, resource needed, assess the iteration and phases

**Inception phase:** early in the inception phase, the archetypal inception iteration workflow, execute the core workflows, requirements to test.

UNIT V

**Elaboration Phase:** elaboration phase in brief, early in the elaboration phase, the architectural elaboration iteration workflow, execute the core workflows-Requirements to test.

**Construction phase:** early in the construction phase, the archetypal construction iteration workflow, execute the core workflow.

**Transition phase:** early in the transition phase, activities in transition phase

**Case Studies:** Automation of a Library, Software Simulator application (2-floor elevator simulator)

TEXT BOOKS:


2. UML 2 Toolkit by Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado WILEY-Dreamtech India Pvt. Ltd.

3. The Unified Software Development Process by Ivar Jacobson, Grady Booch, James Rumbaugh, Pearson Education
REFERENCES BOOKS:

1. Fundamentals of Object Oriented Design in UML By Meilir Page-Jones, Pearson Education
3. Practical Object-Oriented Design with UML By Mark Priestley, TATA Mc Graw Hill
4. Object Oriented Analysis & Design By Brett D McLaughlin, Gary Pollice and David West, O’REILY.
7. UML and C++,R.C.Lee, and W.M.Tepfenhart, PHI.
Objectives:
1. Write syntactically correct HTTP messages and describe the semantics of common HTTP methods and header fields.
2. Discuss differences between URIs, URNs, and URLs, and demonstrate a detailed understanding of http-scheme URLs, both relative and absolute.
3. Describe the actions, including those related to the cache, performed by a browser in the process of visiting a Web address.
4. Install a web server and perform basic administrative procedures, such as tuning communication parameters, denying access to certain domains, and interpreting an access log.
5. Write a valid standards-conformant HTML document involving a variety of element types, including hyperlinks, images, lists, tables, and forms.
6. Use CSS to implement a variety of presentation effects in HTML and XML documents, including explicit positioning of elements.
7. Demonstrate techniques for improving the accessibility of an HTML document.

List of Sample Problems:

i) Web Technologies
1. Develop static pages (using Only HTML) of an online Book store. The pages should resemble: www.amazon.com the website should consist of the following pages.
- Home page, Registration and user Login
- User Profile Page, Books catalog
- Shopping Cart, Payment By credit card
- Order Conformation
2. Validate the Registration, user login, user profile, and payment by credit card pages using JavaScript.
3. Create and save an XML document at the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.
4. Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using Servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.
5. Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books
catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

6. Implement the “Hello World!” program using JSP Struts Framework.

ii) Additional Assignment Problems

Write an HTML page including any required Javascript that takes a number from one text field in the range of 0 to 999 and shows it in another text field in words. If the number is out of range, it should show “out of range” and if it is not a number, it should show “not a number” message in the result box.

Write a java swing application that takes a text file name as input and counts the characters, words and lines in the file. Words are separated with white space characters and lines are separated with new line character.

Write a simple calculator servlet that takes two numbers and an operator (+, -, /, * and %) from an HTML page and returns the result page with the operation performed on the operands. It should check in a database if the same expression is already computed and if so, just return the value from database. Use MySQL or PostgreSQL.

Write an HTML page that contains a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).

Write a servlet that takes name and age from an HTML page. If the age is less than 18, it should send a page with “Hello <name>, you are not authorized to visit this site” message, where <name> should be replaced with the entered name. Otherwise it should send “Welcome <name> to this site” message.

Write a calculator program in HTML that performs basic arithmetic operations (+, -, /, * and %). Use CSS to change the foreground and background color of the values, buttons and result display area separately. Validate the input strings using JavaScript regular expressions. Handle any special cases like division with zero reasonably. The screen may look similar to the following:

![Calculator Screen]

Write a Java program that creates a calculator GUI, as shown in figure. Extra components may be added for convenience:
The Color Scheme may be Black on White or Blue on Yellow (selectable) and accordingly all components colors must be changed. The values can be either entered or increased or decreased by a step of 10. The operators are +, -, / and * (selectable). Once any change takes place, the result must be automatically computed by the program.

Write a Java Application that will read an XML file that contains personal information (Name, Mobile Number, age and place. It reads the information using SAX parser. After reading the information, it shows two input Text Fields in a window, one for tag name and the other for value. Once these two values are given, it should list all the records in the XML file that match the value of the given field in a text area (result box). For example, if the two text boxes are entered with “name” and “ABCD” then it should show all the records for which name is “ABCD”? An Illustration is given below that takes a mobile number and lists all the records that have the same mobile number.

<table>
<thead>
<tr>
<th>Field</th>
<th>mobile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>9449449449</td>
</tr>
<tr>
<td>Result</td>
<td>abc, 22, Hyd, def, 23, Delhi, xxx, 44, Chennai</td>
</tr>
</tbody>
</table>

Consider the following web application for implementation:

The user is first served a login page which takes user's name and password. After submitting the details the server checks these values against the data from a database and takes the following decisions.

If name and password matches, serves a welcome page with user's full name.

If name matches and password doesn't match, then serves “password mismatch” page

If name is not found in the database, serves a registration page, where users full name, present user name (used to login) and password are collected. Implement this application in:

1. Pure JSP
2. Pure Servlets
3. Struts Framework

Implement a simple arithmetic calculator with +, -, /, *, % and = operations using Struts Framework The number of times the calculator is used should be displayed at the bottom (use session variable).
### ii) Web Technologies and Services Lab - Additional Problems

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a web Service in Java that takes two city names from the user and returns the distance between these two from data available from a table in MySql.</td>
<td>Write a java and a C# client which use the above service</td>
</tr>
<tr>
<td>Write a Java program that takes a file as input and encrypts it using DES encryption. The program should check if the file exists and its size is not zero.</td>
<td>Write a Java program that generates a key pair and encrypts a given file using RSA algorithm.</td>
</tr>
<tr>
<td>Write a Java program that finds digest value of a given string.</td>
<td></td>
</tr>
<tr>
<td>Consider the following xml file for encryption</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;?xml version=&quot;1.0&quot;&gt; &lt;transaction&gt; &lt;from&gt;12345&lt;/from&gt; &lt;to&gt;54321&lt;/to&gt; &lt;amount&gt;10000&lt;/amount&gt; &lt;secretcode&gt;abc123&lt;/secretcode&gt; &lt;checksum&gt;&lt;/checksum&gt; &lt;/transaction&gt;</code></td>
</tr>
<tr>
<td></td>
<td>Replace <code>&lt;from&gt;</code> and <code>&lt;to&gt;</code> values with the RSA encrypted values represented with base64 encoding assuming that the public key is available in a file in local directory “pubkey.dat”. Encrypt <code>&lt;secretcode&gt;</code> with AES algorithm with a password ‘secret’. The checksum of all the field values concatenated with a delimiter character ‘+’ will be inserted in the checksum and the xml file is written to encrypted.xml file.</td>
</tr>
<tr>
<td></td>
<td>Assume that a file ‘config.xml’, which has the following information: <code>&lt;users&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;user&gt; &lt;name&gt;abc&lt;/name&gt; &lt;pwd&gt;pwd123&lt;/pwd&gt; &lt;role&gt;admin&lt;/role&gt; &lt;md5&gt;xxx&lt;/md5&gt; &lt;/user&gt; &lt;user&gt; &lt;name&gt;def&lt;/name&gt; &lt;pwd&gt;pwd123&lt;/pwd&gt; &lt;role&gt;guest&lt;/role&gt; &lt;md5&gt;xxx&lt;/md5&gt; &lt;/user&gt; &lt;/users&gt;</code></td>
</tr>
<tr>
<td></td>
<td>Replace name and role with DES encrypted values and pwd with RSA encrypted values (represent the values with base64 encoding). The public key is available in “public.key” file in current directory. Replace xxx with respective MD5 values of all the fields for each user. Write the resulting file back to config.xml.</td>
</tr>
<tr>
<td></td>
<td>Write an HTML page that gives 3 multiple choice (a,b,c and d) questions from a set of 5 preloaded questions randomly. After each question is answered change the color of the question to either green or blue using CSS. Finally on clicking OK button that is provided, the score</td>
</tr>
</tbody>
</table>
should be displayed as a pop-up window. Use Java Script for dynamic content.

Write an HTML page that has 3 countries on the left side (“USA”, “UK” and “INDIA”) and on the right side of each country, there is a pull-down menu that contains the following entries: (“Select Answer”, “New Delhi”, “Washington” and “London”). The user will match the Countries with their respective capitals by selecting an item from the menu. The user chooses all the three answers (whether right or wrong). Then colors of the countries should be changed either to green or to red depending on the answer. Use CSS for changing color.

Write an HTML Page that can be used for registering the candidates for an entrance test. The fields are: name, age, qualifying examination (diploma or 10+2), stream in qualifying examination. If qualifying examination is “diploma”, the stream can be “Electrical”, “Mechanical” or “Civil”. If the qualifying examination is 10+2, the stream can be “MPC” or “BPC”. Validate the name to accept only characters and spaces.

Write an HTML page that has two selection menus. The first menu contains the states (“AP”, “TN” and “KN”) and depending on the selection the second menu should show the following items: “Hyderabad”, “Vijayawada”, “Kurnool” for AP, “Chennai”, “Salem”, “Madurai” for TN and “Bangalore”, “Bellary”, “Mysore” for KN.

Write an HTML page that has phone buttons 0 to 9 and a text box that shows the dialed number. If 00 is pressed at the beginning, it should be replaced with a + symbol in the text box. If the number is not a valid international number (+ followed by country code and 10 digit phone number) the color of the display should be red and it should turn to green when the number is valid. Consider only “+91, +1 and +44 as valid country codes. Use CSS for defining colors.

Write an HTML page that has a text box for phone number or Name. If a number is entered in the box the name should be displayed next to the number. If 00 is pressed at the beginning, it should be replaced with a + symbol in the text box. If a name is entered in the text box, it should show the number next to the name. If the corresponding value is not found, show it in red and show it in green otherwise. Use CSS for colors. Store at least 5 names and numbers in the script for testing.

A library consists of 10 titles and each title has a given number of books initially. A student can take or return a book by entering his/her HTNo as user ID and a given password. If there are at least two books, the book is issued and the balance is modified accordingly.

(a) Use RDBMS and implement it with JSP.
(b) Use XML File for data and Implement it with JSP

7 Use RDBMS and implement it with Servlets

8 Use XML File for data and Implement it with Servlets
A Bus Reservation System contains the details of a bus seat plan for 40 seats in 2x2 per row arrangement, where the seats are numbered from 1 to 40 from first row to last row. The customer can visit the website and can reserve a ticket of his choice if available by entering his details (Name, Address, Gender and Age). The customer can cancel the ticket by entering the seat number and his name as entered for reservation.

(a) Use RDBMS and implement it with JSP.

(b) Use XML File for data and Implement it with JSP

(c) Use RDBMS and implement it with Servlets

(d) Use XML File for data and Implement it with Servlets

Implement a simple messaging system with the following details:

When a student logs in with his/her HTNO and a given password, they should get all the messages posted to him/her giving the ID of sender and the actual message. Each message may be separated with a ruler. There should be a provision for the user to send a message to any number of users by giving the IDs separated with commas in the “To” text box.

(a) Use RDBMS and implement it with JSP.

(b) Use XML File for data and Implement it with JSP

(c) Use RDBMS and implement it with Servlets

(d) Use XML File for data and Implement it with Servlets.

There is an image of 600x100 size which can be logically divided into 12 button areas with labels (0-9, +, =).

Write a javascript calculator program that uses this image as input virtual keyboard and three text areas for two input numbers and result of sum of these numbers. Add a CSS that can be used to
A web application has the following specifications:

The first page (Login page) should have a login screen where the user gives the login name and password. Both fields must be validated on client side for a minimum length of 4 characters, name should be lower case a-z characters only and password should contain at least one digit. On submitting these values, the server should validate them with a MySQL database and if failed, show the login page along with a message saying “Login Name or Password Mismatch” in Red color below the main heading and above the form. If successful, show a welcome page with the user's full name (taken from database) and and a link to Logout. On logout, a good bye page is displayed with the total time of usage (Logout time – login time). Specify the Schema details of table and web.xml file contents.

Implement it using (a) JSP Pages (b) Servlets (c) Struts

Design a struts based web portal for an international conference with following specifications:

The welcome page should give the details of the conference and a link to login. If login fails, direct them back for re-login and also provide a link for registration. On successful registration/login, the user will be directed to a page where s/he can see the status (accepted/rejected) of their already submitted papers followed by a form for submitting a doc file to the conference. Provide a logout button on all pages including the home page, once the user logs in. Implement validation framework to check that the user name is in the form of CCDDCC and password is in the form of (CCSDDD) (C for character, S for special character (one of @, #, $, %, ^, & and !) and D for digit). Database should be accessed through Connection Pool for MySql for user information. Provide scope for internationalization in future. Assume any missing information and mention it first.
SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

Objectives:
After completing this course, the student should be able to:

- To understand the concept of patterns and the Catalog.
- To discuss the Presentation tier design patterns and their affect on: sessions, client access, validation and consistency.
- To understand the variety of implemented bad practices related to the Business and Integration tiers.
- To highlight the evolution of patterns.
- To learn how to add functionality to designs while minimizing complexity.
- To learn what design patterns really are, and are not.
- To know about specific design patterns.
- To learn how to use design patterns to keep code quality high without overdesign.

UNIT I
Envisioning Architecture

Creating an Architecture
Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture.

UNIT II
Analyzing Architectures
Architecture Evaluation, Architecture design decision making, ATAM, CBAM.

Moving from one system to many
Software Product Lines, Building systems from off the shelf components, Software architecture in future.

UNIT III
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, flyweight.

UNIT IV
Behavioural patterns
Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

UNIT V
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:

2. Design Patterns, Erich Gamma, Pearson Education.

REFERENCE BOOKS:

2. Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR, 2001
5. Design Patterns in Java, Steven John Metsker & William C. Wake, Pearson education, 2006
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – II Sem.

SOFTWARE PROCESS AND PROJECT MANAGEMENT

Objectives:
At the end of the course, the student shall be able to:

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

UNIT I

Process Reference Models Capability Maturity Model (CMM), CMMi, PCMM, PSP, TSP.

UNIT II

Life-Cycle Phases and Process artifacts Engineering and Production stages, inception phase, elaboration phase, construction phase, transition phase, artifact sets, management artifacts, engineering artifacts and pragmatic artifacts, model based software architectures.

UNIT III
Workflows and Checkpoints of process Software process workflows, Iteration workflows, Major milestones, Minor milestones, Periodic status assessments.

Process Planning Work breakdown structures, Planning guidelines, cost and schedule estimating process, iteration planning process, Pragmatic planning.

UNIT IV
Project Organizations Line-of- business organizations, project organizations, evolution of organizations, process automation.

Project Control and process instrumentation The seven core metrics, management indicators, quality indicators, life-cycle expectations, Pragmatic software metrics, metrics automation.
UNIT V

CCPDS-R Case Study and Future Software Project Management Practices

TEXT BOOKS:
2. Software Project Management, Walker Royce, Pearson Education.

REFERENCE BOOKS:
6. Head First PMP, Jennifer Greene & Andrew Stellman, O’Reilly, 2007
SOFTWARE QUALITY ASSURANCE AND TESTING

Objectives:
The student should be able to:

- To understand software testing and quality assurance as a fundamental component of software life cycle
- To define the scope of SW T&QA projects
- To efficiently perform T&QA activities using modern software tools
- To estimate cost of a T&QA project and manage budgets
- To prepare test plans and schedules for a T&QA project
- To develop T&QA project staffing requirements
- To effectively manage a T&QA project

UNIT I
Software Quality Assurance and Standards

Quality Standards: ISO 9000 and Companion ISO Standards, CMM, CMMI, PCMM, Malcom Balridge, 3 Sigma, 6 Sigma and other latest quality standards (Refer Internet and R11, R12, R13).

UNIT II

Building Software Testing Process: Software Testing Guidelines, workbench concept, Customizing the Software Testing Process, Process Preparation checklist - (Chapters: 2,3) of T1

Software Testing Techniques: Dynamic Testing – Black Box testing techniques, White Box testing techniques, Static testing, Validation Activities, Regression testing -(Chapters: 4, 5, 6, 7, 8) of T2
UNIT III
Software Testing Tools
Selecting and Installing Software Testing tools – (Chapter 4) of T1. Automation and Testing Tools - (Chapter 15) of T2
Load Runner, Win runner and Rational Testing Tools, Silk test, Java Testing Tools, JMetra, JUNIT and Cactus. (Refer Internet and R9, R10)

UNIT IV
Testing Process
Seven Step Testing Process – I: Overview of the Software Testing Process, Organizing of Testing, Developing the Test Plan, Verification Testing, Validation Testing. (Chapters 6, 7, 8, 9, 10) of T1

UNIT V
Seven Step Testing Process – II: Analyzing and Reporting Test results, Acceptance and Operational Testing, Post-Implementation Analysis

TEXT BOOKS:

REFERENCE BOOKS:


ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – II Sem.

COMPONENT BASED SOFTWARE ENGINEERING
(OPEN ELECTIVE-II)

Objectives:
- To understand the essentials of component-based software engineering
- To know the main characteristics of components and component models
- To be aware of software development processes for component-based systems
- To be aware of the mutual relations between software architecture and component models

UNIT I
The Case for Components- The Business Case for components, COTS Myths and Other Lessons Learned in Component-Based Software Development.

UNIT II

UNIT III

UNIT IV
UNIT V
Component Technologies - Overview of the CORBA Component Model, Overview of COM+, Overview of the EJB Component Model, Bonobo and Free Software GNOME Components, Choosing between COM+, EJB, and CCM, Software Agents as Next Generation Software Components.

TEXT BOOKS:

REFERENCE BOOKS:
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – II Sem.                                L    P    C
                                                      4    0    4

COMPUTER FORENSICS
(OPEN ELECTIVE - II)

Objectives:
1. To understand the cyberspace
2. To understand the forensics fundamentals
3. To understand the evidence capturing process.
4. To understand the preservation of digital evidence.

UNIT I

UNIT II

UNIT III

UNIT IV
Computer Evidence, Tools Needed for Intrusion Response to the Destruction of Data, Incident Reporting and Contact Forms.

UNIT V

TEXT BOOKS:

REFERENCES:
4. Dan Farmer & Wietse Venema, "Forensic Discovery", Addison Wesley, 2005
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – II Sem.

L  P  C
4  0  4

E – COMMERCE
(OPEN ELECTIVE - II)

Objectives:

- Identify the major categories and trends of e-commerce applications.
- Identify the essential processes of an e-commerce system.
- Identify several factors and web store requirements needed to succeed in e-commerce.
- Discuss the benefits and trade-offs of various e-commerce clicks and bricks alternatives.
- Understand the main technologies behind e-commerce systems and how these technologies interact.
- Discuss the various marketing strategies for an online business. Define various electronic payment types and associated security risks and the ways to protect against them.

UNIT - I

UNIT – II
Electronic payment systems - Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems. Inter Organizational Commerce - EDI, EDI Implementation, Value added networks.

UNIT – III

UNIT- IV
Advertising and Marketing - Information based marketing, Advertising on Internet, on-line marketing process, market research. Consumer Search and Resource Discovery - Information search and Retrieval, Commerce Catalogues, Information Filtering.

UNIT - V
Multimedia - key multimedia concepts, Digital Video and electronic Commerce, Desktop video processing, Desktop video conferencing.
TEXT BOOK:

REFERENCES BOOKS:
1. E-Commerce fundamentals and applications Hendry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang, John Wiley.
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – II Sem.

L  P  C
4  0  4

ADVANCED DATA MINING
(ELECTIVE -III)

Objectives:

- To develop the abilities of critical analysis to data mining systems and applications.
- To implement practical and theoretical understanding of the technologies for data mining
- To understand the strengths and limitations of various data mining models;

UNIT-I
Data mining Overview and Advanced Pattern Mining Data mining tasks - mining frequent patterns, associations and correlations, classification and regression for predictive analysis, cluster analysis, outlier analysis; advanced pattern mining in multilevel, multidimensional space - mining multilevel associations, mining multidimensional associations, mining quantitative association rules, mining rare patterns and negative patterns.

UNIT-II
Advance Classification
Classification by back propagation, support vector machines, classification using frequent patterns, other classification methods - genetic algorithms, roughest approach, fuzzyset approach;

UNIT-III
Advance Clustering
Density - based methods -DBSCAN, OPTICS, DENCLUE; Grid-Based methods - STING, CLIQUE; Exception - maximization algorithm; clustering High-Dimensional Data; Clustering Graph and Network Data.

UNIT-IV
Web and Text Mining
Introduction, web mining, web content mining, web structure mining, web usage mining, Text mining -unstructured text, episode rule discovery for texts, hierarchy of categories, text clustering.
UNIT-V

Temporal and Spatial Data Mining

Introduction; 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. Data Mining Concepts and Techniques, Jiawei Hang Micheline Kamber, Jian pei, Morgan Kaufmann.
2. Data Mining Techniques - Arun K pujari, Universities Press.

REFERENCE BOOKS:
1. Introduction to Data Mining - Pang-Ning Tan, Vipin kumar, Michael Steinbach, Pearson.
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – II Sem.       L    P    C
                                      4   0   4

SCRIPTING LANGUAGES
(ELECTIVE-III)

Objectives:

• The course demonstrates an in depth understanding of the tools and the scripting languages necessary for design and development of applications dealing with Bio-information/ Bio-data. The instructor is advised to discuss examples in the context of Bio-data/ Bio-information application development.

UNIT I Introduction to PERL and Scripting

Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Web Scripting, and the universe of Scripting Languages, PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines, advance perl - finer points of looping, pack and unpack, filesystem, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

UNIT II PHP Basics

PHP Basics- Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Datatypes, Variables, Constants, expressions, string interpolation, control structures, Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions.

UNIT III Advanced PHP Programming

Php and Web Forms, Files, PHP Authentication and Methodologies -Hard Coded, File Based, Database Based, IP Based, Login Administration, Uploading Files with PHP, Sending Email using PHP, PHP Encryption Functions, the Mcrypt package, Building Web sites for the World – Translating Websites- Updating Web sites Scripts, Creating the Localization Repository, Translating Files, text, Generate Binary Files, Set the desired language within your scripts, Localizing Dates, Numbers and Times.

UNIT IV TCL – Tk

TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures , strings , patterns, files, Advance TCL- eval, source, exec and up level commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C
Interface. Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding, Perl-Tk.

UNIT V Python


TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP, J.Lee and B.Ware (Addison Wesley) Pearson Education.
2. Programming Python, M.Lutz, SPD.
4. PHP 5.1, I.Bayross and S.Shah, The X Team, SPD.
5. Core Python Programming, Chun, Pearson Education.
7. Perl by Example, E.Quigley, Pearson Education.
8. Programming Perl, Larry Wall, T.Christiansen and J.Orwant, O’Reilly, SPD.
9. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.
10. PHP and MySQL by Example, E.Quigley, Prentice Hall(Pearson).
12. PHP Programming solutions, V.Vaswani, TMH.
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)
M. Tech – I Year – II Sem. L P C
4 0 4
INFORMATION RETRIEVAL SYSTEMS
(ELECTIVE III)

Objectives:
On completion of this course you should have gained a good understanding of the foundation concepts of information retrieval techniques and be able to apply these concepts into practice. Specifically, you should be able to:

- To use different information retrieval techniques in various application areas
- To apply IR principles to locate relevant information large collections of data
- To analyse performance of retrieval systems when dealing with unmanaged data sources
- To implement retrieval systems for web search tasks.

UNIT I

UNIT II
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

UNIT IV
Support vector machines and machine learning on documents, flat clustering, Hierarchical clustering, Matrix decompositions and latent semantic indexing.

UNIT V
Web search basics, Web crawling and indexes, Link analysis.
TEXT BOOKS:

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

REFERENCE BOOKS:

5. Information Storage & Retrieval, Robert Korfhage, John Wiley & Sons.
ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – II Sem.

SEMANTIC WEB AND SOCIAL NETWORKS
(ELECTIVE-III)

Objectives:

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

Unit –I: Web Intelligence

Unit -II: Knowledge Representation for the Semantic Web

Unit-III: Ontology Engineering
Ontology Engineering, Constructing Ontology, Ontology Development Tools, Ontology Methods, Ontology Sharing and Merging, Ontology Libraries and Ontology Mapping, Logic, Rule and Inference Engines.

Unit-IV: 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-V: 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:


REFERENCE BOOKS:

ANURAG GROUP OF INSTITUTIONS
(AUTONOMOUS)

M. Tech – I Year – II Sem.  L  P  C
4  0  4

CLOUD COMPUTING
(ELECTIVE-IV)

Objectives:

Prerequisite: Computer Networks and Operating Systems

Course Description:
Cloud computing has evolved as a very important computing model, which enables information, software, and shared resources to be provisioned over the network as services in an on-demand manner. This course provides an insight into what is cloud computing and the various services cloud is capable.

UNIT I
Systems Modeling, Clustering and Virtualization
Distributed System Models and Enabling Technologies, Computer Clusters for Scalable Parallel Computing, Virtual Machines and Virtualization of Clusters and Data centers.

UNIT II Foundations

UNIT III
Infrastructure as a Service (IAAS) & Platform and Software as a Service (PAAS / SAAS)

UNIT IV
Monitoring, Management and Applications
UNIT V

Governance and Case Studies


TEXT BOOKS:


REFERENCE BOOKS:

Objectives:
By the end of the course, you will know:
- History and Structure of databases
- How to design a database
- How to convert the design into the appropriate tables
- Handling Keys appropriately
- Enforcing Integrity Constraints to keep the database consistent
- Normalizing the tables to eliminate redundancies
- Querying relational data
- and processing the queries
- Storage Optimizing Strategies for easy retrieval of data through index
- Triggers, Procedures and Cursors, Transaction Management
- Distributed databases management system concepts and Implementation

UNIT I
Database System Applications, Purpose of Database Systems, View of Data – Data Abstraction, Instances and Schemas, Data Models – the ER Model, Relational Model, Other Models – Database Languages – DDL, DML, Database Access from Applications Programs, Transaction Management, Data Storage and Querying, Database Architecture, Database Users and Administrators, ER diagrams, Relational Model: Introduction to the Relational Model – Integrity Constraints Over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views – Altering Tables and Views, Relational Algebra, Basic SQL Queries, Nested Queries, Complex Integrity Constraints in SQL, Triggers

UNIT II
UNIT III
Transaction Management: The ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions – Lock Based Concurrency Control, Deadlocks – Performance of Locking – Transaction Support in SQL.
Concurrency Control: Serializability, and recoverability – Introduction to Lock Management – Lock Conversions, Dealing with Deadlocks, Specialized Locking Techniques – Concurrency Control without Locking.
Crash recovery: Introduction to Crash recovery, Introduction to ARIES, the Log, and Other Recovery related Structures, the Write-Ahead Log Protocol, Check pointing, recovering from a System Crash, Media recovery

UNIT IV
Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing – Clustered Indexes, Primary and Secondary Indexes, Index data Structures – Hash Based Indexing, Tree based Indexing
Hash Based Indexing: Static Hashing, Extendable hashing, Linear Hashing, Extendable Vs Linear Hashing.

UNIT V
Distributed databases: Introduction to distributed databases, Distributed DBMS architectures, Storing data in a distributed DBMS, Distributed catalog management, Distributed query processing
Updating distributed data, Distributed transactions, Distributed concurrency control, Distributed recovery

TEXT BOOKS:

REFERENCE BOOKS:

1. Introduction to Database Systems, C.J.Date, Pearson Education.
2. Database Management System Oracle SQL and PL/SQL, P.K.Das Gupta, PHI.
9. Distributed Databases Principles & Systems, Stefano Ceri, Giuseppe Pelagatti, TMH.
Objectives:

- To recognize the role of business processes within an Infinity based application
- To understand the importance of parameter sets to a business process
- To learn common patterns and best practices for formatting and restricting the output from a business process
- To understand the difference between a business process and a business process instance
- To learn how data processing occurs within a business process
- To list the Infinity SDK software developer responsibilities for building and supporting the functionality required for a business process
- To describe the database tables used by Blackbaud Enterprise CRM to manage business processes
- To describe how a QueryViewSpec can be used to define the output format for a business process
- To describe how a selection can be used to limit the rows processed by a business process

UNIT I

UNDERSTANDING BPM - I:
How can we demystify business process management?
What is business process management?
Why is it important to improve business process before automating them?
When should you do BPM – what are the main drivers and triggers?
Who should be involved in BPM?
UNIT II

UNDERSTANDING BPM - II:
Why are organizational strategy and process architecture important in BPM implementation? How do you sell BPM technology to the organization?
What are the critical success factors in a BPM project?
What are the critical implementation aspects for a BPM solution?

UNIT III
FRAMEK - I:
Framework overview, Guidelines on how to use the framework, Organization strategy phase, Process architecture phase, Launch pad phase, Understand phase, Innovate phase.

UNIT IV
FRAMEK – II:
People phase, Develop phase, Implement phase, Realize value phase, Sustainable performance phase, Essentials introduction, Project management, People change management, Leadership.

UNIT V
BPM AND THE ORGANIZATION:
BPM maturity, Embedding BPM within the organization.

TEXT BOOKS:


REFERENCE BOOK:

SOFTWARE TESTING LAB

Objectives:
The student should be able to:

- To understand software testing and quality assurance as a fundamental component of software life cycle
- To define the scope of SW T&QA projects
- To efficiently perform T&QA activities using modern software tools
- To estimate cost of a T&QA project and manage budgets
- To prepare test plans and schedules for a T&QA project
- To develop T&QA project staffing requirements
- To effectively manage a T&QA project

Software Testing Objectives:
To learn to use the following (or Similar) automated testing tools to automate testing:

a. Win Runner/QTP for functional testing.
b. LoadRunner for Load/Stress testing.
c. Test Director for test management.
d. JUnit, HTMLUnit, CPPUnit.

Sample problems 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* and final test report document.

**Additional problems on testing:**

1. Test the following using JUnit and CPPUnit:
   i) Sorting problems  
   ii) Searching problems
   iii) Finding gcd of two integers  
   iv) Finding factorial of a number.
10. Test web based forms using HTMLUnit.
11. Test database stored procedures using SQLUnit.
(Use sufficient number of test cases in solving above Problems)

*Note: To create the various testing related documents refer to the text “Effective Software Testing Methodologies by William E. Perry”

**REFERENCE BOOKS:**


Publishers and Distributors, 2012