Shri Vaishnav Vidyapeeth Vishwavidyalaya
Master of Technology (CSE with specialization in Big Data Analytics)
Choice Based Credit System (CBCS)-2019-20
SEMESTER II

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>CATEGORY</th>
<th>COURSE NAME</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTCS201</td>
<td></td>
<td>Advance Computer Architecture &amp; Organization</td>
<td>3</td>
<td>-</td>
<td>-</td>
<td>3</td>
</tr>
</tbody>
</table>

**Legends:** L - Lecture; T - Tutorial/Teacher Guided Student Activity; P – Practical; C - Credit;
*Teacher Assessment* shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

**Course Objective:**

1. Describe current and emerging trends in computer architectures, focusing on performance and the hardware/software interface.
2. Analyzing fundamental issues in architecture design and their impact on application performance.
3. Identify the performance and efficiency in advanced multiple-issue processors
4. Identify and analyzing various memory models.
5. Describe various techniques to enhance a processor's ability.

**Course Outcomes:**

1. Know the classes of computers, and new trends and developments in computer architecture
2. Understand pipelining, instruction set architectures, memory addressing.
3. Understand the performance metrics of microprocessors, memory, networks.
4. Understand the various techniques to enhance a processor's ability to exploit Instruction-level parallelism (ILP), and its challenges.
5. Understand exploiting ILP using dynamic scheduling, multiple issue, and speculation.
6. Understand multithreading by using ILP and supporting thread-level parallelism (TLP).
7. Understand the performance and efficiency in advanced multiple-issue processors.
9. Understand multiprocessor cache coherence using the directory based and snooping class of protocols.
10. Understand the various models to achieve memory consistency.

**UNIT I**

Overview of Parallel Processing and Pipelining Processing, study and comparison of uni-processors and parallel processors, Evolution of parallel processors, Necessity of high performance, Architectural Classification, Applications of parallel processing, Instruction level Parallelism and Thread Level Parallelism.
UNIT II
Principles and implementation of Pipelining, Pipeline Architecture, Study and comparison of processors with and without pipelining, Linear pipeline processor, Nonlinear pipeline processor Instruction pipeline design, Mechanisms for instruction pipelining, pipeline hazards, Dynamic instruction scheduling -score boarding and Tomosulo’s algorithm, Branch handling techniques, Arithmetic Pipeline Design, Static arithmetic pipeline, Multifunctional arithmetic pipelines. Superscaler pipeline design, Super pipeline processor design.

UNIT III
Study and comparison of Vector and array processors, Vector Processing Principles, Vector instruction types, Vector-access memory schemes. Vector supercomputer architecture, SIMD Computer Organization Masking and Data network mechanism, distributed memory model and shared memory model, Parallel Algorithms For Array Processors: Matrix Multiplication. Sorting, SIMD computer organization, Implementation issues of Matrix multiplication and sorting on array processor and their analysis

UNIT IV
Microprocessor Architectures, study and comparison of Loosely and Tightly coupled multiprocessors. Loosely and Tightly coupled multiprocessors, Processor characteristics of multiprocessors, Inter Processor communication network, Time shared bus, Crossbar switch, Multiport Memory Model, Memory contention and arbitration techniques, Cache coherence, Snoopy protocols, Directory based protocols. Message routing schemes in multicomputer network, deadlock and virtual channel.

UNIT V

Text books:
2. J.P.Hayes, “computer Architecture and organization”; MGH.
6. Hwang and Briggs, “Computer Architecture and Parallel Processing”; MGH.

Practical's List:
1. Pi Calculation for implementing parallel programming
2. Implement Array Processing in respect of an array processor
3. Write a program for implementing Client server architecture.
4. Implement Reservation Table program for pipelining.
5. Implementation of multithreading in java.
6. Implement RMI using one web application.
7. Implement Remote Procedure Call on windows.
8. Implement Client – server communication in C/C++/Java.
9. Write a program to calculate access time of each storage device for same file. Case study of VLIW processor, Pentium pro, CRAY Computer systems.
**Course: Advance Database Management System**

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>CATEGORY</th>
<th>COURSE NAME</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CREDITS</th>
<th>THEORY</th>
<th>PRACTICAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTCS202</td>
<td></td>
<td>Advance Database Management System</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>60</td>
<td>20</td>
<td>20</td>
</tr>
</tbody>
</table>

**Legends:**
- **L** - Lecture
- **T** - Tutorial/Teacher Guided Student Activity
- **P** - Practical
- **C** - Credit

*Teacher Assessment* shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

**Course Objective:**

1. Describe database management system internals. Understand and describe internal algorithms in detail. Decide on optimization issues given a known database workload, by manipulating indexes, choosing more adequate data types, and modifying queries.
2. Identify opportunities for the use of the object model, and design and code client code to manipulate an object database.
3. Analyze and optimize transactional code, identifying causes of possible anomalies and correct them.
4. Identify and be able to use recent and advanced database techniques (e.g. in concurrency control, buffer management, and recovery).
5. Analyze, describe and use other models than the Relational. Analyze, compare and evaluate alternative database architectures and models in different application contexts. Identify limitations of the standard Relational databases in certain application domains, e.g. for multidimensional data, or unstructured data.

**Course Outcomes:**

1. Be familiar with a commercial relational database system (Oracle) by writing SQL using the system.
2. Be familiar with the relational database theory, and be able to write relational algebra expressions for queries.
3. Be familiar with basic database storage structures and access techniques: file and page organizations, indexing methods including B-tree, and hashing.
4. Master the basics of query evaluation techniques and and query optimization.
5. Be familiar with the basic issues of transaction processing and concurrency control.

**UNIT I**

Introduction to Database Systems: Database System Concepts and Architecture, Data Models, Data Independence, SQL: DDL, DML, DCL, Normalization: 1NF, 2NF, 3NF, BCNF, 4NF, 5NF. Query Processing and Optimization: Query Processing, Syntax Analyzer, Query Decomposition, Query
Optimization, Heuristic Query Optimization, Cost Estimation, Cost Functions for Select, Join, Query Evaluation Plans.

UNIT II
Object Oriented and Object Relational Databases Object Oriented Concepts, Object Oriented Data Model, Object Definition Language, Object Query Language, Object Relational Systems, SQL3, ORDBMS Design.

UNIT III

UNIT IV

UNIT V
Introduction to Data Warehousing and Data Mining: Introduction to OLAP, OLTP, Data Warehouse, Data Marts, Data Mining, Data Mining Process. Distributed Databases: Distributed Database Concepts, Advantages and Disadvantages, Types of Distributed Database Systems, Data Fragmentation, Replication and Allocation Techniques for Distributed Database Design, Five Level Schema Architecture, Query Processing, Concurrency Control and Recovery in Distributed Databases. Commercial Databases: Commercial Database Products, Familiarity with IBM DB2 Universal Database, Oracle, Microsoft SQL Server, MySQL, their features.

References:
1. C. J. Date: An Introduction to Database Systems , Addison-Wesley
2. Avi Silberschatz, Henry F. Korth ,S. Sudarshan ,Data Base System Concepts, TMH
3. Patrick O’Neil & Elizabeth O’Neil, Database Principles, Programming and Performance,
4. Morgan Kaufmann Hardcourt India
5. Gillenson, Fundamental of Data Base Management Sytem, Willey India
6. Ceri & Pelagatti, Distributed Databases Principles & Systems,TMH
7. Paulraj Ponniah, Data Ware Housing Fundamental, Willey India.
8. Jiawei Han, Data Mining Concept & Techniques, Elsevier Pub.

Practical's List:
1. Distributed Database for Bookstore
2. Deadlock Detection Algorithm for distributed database using wait- for graph
3. Object Oriented Database – Extended Entity Relationship (EER)
4. Parallel Database – University Counselling for Engineering colleges
5. Parallel Database – Implementation of Parallel Join & Parallel Sort
6. Active Database – Implementation of Triggers & Assertions for Bank Database
7. Deductive Database – Constructing Knowledge Database for Kinship Domain (Family Relations)
8. Study and Working of WEKA Tool
10. Designing XML Schema for Company Database
UNIT I
Basic Simulation Modeling: The Nature of simulation system, models and simulation, discrete-event simulation, simulation of a single-server queuing, alternative approaches to modeling and coding simulations, network simulation, parallel and distributed simulation, simulation across the internet and web based simulation, steps in a sound simulation study, other types of simulation: continuous simulation, combined discrete-continuous simulation, Monte Carlo simulation, advantages, disadvantages and pitfalls of simulation.

UNIT II
Modeling Complex Systems: Introduction, list processing in simulation, approaches to stering lists in a computer linked storage allocation

UNIT III
Discrete System Modeling: Classification of simulation models the simulation process, system investigation validation and translation, simulation of complex discrete-event systems with application in industrial and service organization tactical planning and management aspects, Random variable generation and analysis.

UNIT IV
Simulation Software: Comparison of simulation packages with programming languages classification of simulation software, general-purpose simulation packages, object-oriented simulation, building valid, credible and appropriately detailed simulation models, experimental design, sensitivity analysis and optimization simulation of manufacturing systems.

UNIT V
Embedded System Modeling: Embedded systems and system level design, models of computation, specification languages, hardware/software code design, system partitioning, application specific processors and memory, low power design
Real-Time system modeling, Fixed Priority scheduling, Dynamic Priority Scheduling
Data Communication Network modeling, IP network intradomain (e.g. OSPF, RIP) routing simulation.
References:
2. Geoffrey Gordon, "System Simulation", PHI

Practical’s List:
1. Simulate CPU scheduling algorithm using queueing system a) FCFS b) SJF c) Priority.
2. Simulate multiplxer/concentrator using queuing system.
3. Simulate congestion control algorithms.. Simulate Disk scheduling algorithms.. Simulate a Manufacturing shop and write a program in GPSS.
4. Simulate Telephone system model and write a program in SIMSCRIPT
5. Implementation of Link state routing algorithm
6. Implementation of data encryption and decryption
8. Install Network Simulator 2 and study network topologies.
Shri Vaishnav Vidyapeeth Vishwavidyalaya
Master of Technology (CSE with specialization in Big Data Analytics)
Choice Based Credit System (CBCS)-2019-20

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>CATEGORY</th>
<th>COURSE NAME</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CREDITS</th>
<th>TEACHING &amp; EVALUATION SCHEME</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTCS215</td>
<td></td>
<td>Big Data Analytics Programming</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>60</td>
<td>University Exam</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Two Term Exam</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Teachers Assessment*</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>END SEM University Exam</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Teachers Assessment*</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>END SEM University Exam</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Teachers Assessment*</td>
</tr>
</tbody>
</table>

Legends: L - Lecture; T - Tutorial/Teacher Guided Student Activity; P – Practical; C - Credit;
*Teacher Assessment shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

Course Objectives:

1. To provide understanding basic network and distributed programming.
2. Constructing a real world application with data storage and retrieval.
3. To leverage the benefits of reusable components.
4. To analyze basic file modes and operations.
5. To apply MapReduce paradigm to solve problems.

Course Outcomes:

1. Able to understand basic network and distributed programming.
2. Constructing a real world application with data storage and retrieval.
3. Leveraging the benefits of reusable components.
4. Analyzing basic file modes and operations.
5. Applying MapReduce paradigm to solve problems.

UNIT I

UNIT II

UNIT III
The Bean - Writing Process - Using Beans to Build an Application - Bean Property Types – Property Editors - Customizers.

UNIT IV
UNIT V

MapReduce program in Java – Map Reduce API – Progamming Examples- Combiner Functions - Distributed MapReduce Job.

Text books :

References:

Practical’s List :
At least 10 practical based on the above Syllabus (i.e. 2 practical from each unit).
Shri Vaishnav Vidyapeeth Vishwavidyalaya  
Master of Technology (CSE with specialization in Big Data Analytics)  
Choice Based Credit System (CBCS)-2019-20

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>CATEGORY</th>
<th>COURSE NAME</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CRédITS</th>
<th>TEACHING &amp; EVALUATION SCHEME</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>END SEM</td>
<td>University Exam</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Two Term Exam</td>
<td>Teacher’s Assessment</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>END SEM</td>
<td>University Exam</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Teacher’s Assessment</td>
<td>Teachers Assessment</td>
</tr>
</tbody>
</table>

**MTCS211**  
*Expert System*  
2 2 3 60 20 20 - 50  

Legends:  
L - Lecture;  
T - Tutorial/Teacher Guided Student Activity;  
P - Practical;  
C - Credit;  

*Teacher Assessment* shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

**Course Objectives:**  
1. Study of computing and mathematics appropriate to the discipline.  
2. Study to analyze a problem, and identify and define the computing requirements appropriate to its solution.  
3. Study to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.  
4. Study to use current techniques, skills, and tools necessary for computing practice.  
5. To understand the concept of Expert system and intelligent system.

**Course Outcomes:**  
1. An ability to apply knowledge of computing and mathematics appropriate to the discipline.  
2. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.  
3. An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.  
4. An ability to use current techniques, skills, and tools necessary for computing practice.  
5. To understand the concept of Expert system and intelligent system.

**UNIT I**  
Introduction to Expert System, Intelligent agents: reactive, deliberative, goal-driven, utility-driven, and learning agents Artificial Intelligence programming techniques.

**UNIT II**  
Knowledge Representation and Reasoning: ontologies, foundations of knowledge representation and reasoning, representing and reasoning about objects, relations, events, actions, time, and space; predicate logic, situation calculus, description logics, reasoning with defaults, reasoning about knowledge, sample applications.

**UNIT III**  
Machine Learning and Knowledge Acquisition: learning from memorization, examples, explanation, and exploration. learning nearest neighbour, naive Bayes, and decision tree classifiers, Q-learning for
UNIT IV

UNIT V

Textbooks:
1. The Engineering of Knowledge-based Systems, A.J. Gonzalez and D. D. Dankel, Prentice Hall. Donald A. Waterman,

References:
3. Bose, Neural Network fundamental with Graph, Algo.& Appl, TMH
6. Hagen, Neural Network Design, Cengage Learning

Practical’s List:
1. Implement A*, AO* algorithms.
2. Implement Naive Bays Algorithm.
3. Implement Knowledge Representation and reasoning.
4. Implement Fuzzy reasoning.
5. Implement Fuzzy Inference Theorem.
7. Explain different types of Expert System.
8. Explain Intelligent Agents.
### Shri Vaishnav Vidyapeeth Vishwavidyalaya

**Master of Technology (CSE with specialization in Big Data Analytics)**

**Choice Based Credit System (CBCS)-2019-20**

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>CATEGORY</th>
<th>COURSE NAME</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CREDITS</th>
<th>THEORY</th>
<th>PRACTICAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTCS213</td>
<td>Human Computer Interaction</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>60</td>
<td>20</td>
<td>20</td>
<td>50</td>
</tr>
</tbody>
</table>

**Legends:** L - Lecture; T - Tutorial/Teacher Guided Student Activity; P – Practical; C - Credit;

*Teacher Assessment* shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

**Course Objectives:**
1. To facilitate communication between students of psychology, design, and computer science on user interface development projects.
2. To provide the future user interface designer with concepts and strategies for making design decisions.
3. To expose the future user interface designer to tools, techniques, and ideas for interface design.
4. To introduce the student to the literature of human-computer interaction.

**UNIT I**
Introduction: Importance of user Interface, definition, importance of good design. Benefits of good design. A brief history of Screen design, The graphical user interface, popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user Interface popularity, characteristics - Principles of user interface.

**UNIT II**
Design process, Human interaction with computers, importance of human, characteristics human consideration, Human interaction speeds, and understanding business junctions. Screen Designing:- Design goals, Screen planning and purpose, organizing screen elements, ordering of screen data and content, screen navigation and flow, Visually pleasing composition, amount of information, focus and emphasis, presentation information simply and meaningfully, information retrieval on web, statistical graphics, Technological consideration in interface design.

**UNIT III**

**UNIT IV**
Components, text and messages, Icons and increases, Multimedia, colors, uses problems, choosing colors. Software tools, Specification methods, interface, Building Tools.

**UNIT V**
Interaction Devices, Keyboard and function keys, pointing devices, speech recognition digitization.
and generation, image and video displays, drivers.

Text books:
1. The essential guide to user interface design, Wilbert O Galitz, WileyDreamTech.
2. Designing the user interface. 3rd Edition Ben Shneidermann, Pearson Education Asia.

Reference:
2. Interaction Design Prece, Rogers, Sharps. Wiley Dreamtech,

Practical’s List:
1. Orthogonal Corners.
2. Web page Aesthetics.
3. Metabolic Pathways.
4. Graph Algorithms.
5. DAG Map.
7. Shortest Path.
8. Aural Table.
10. Spring Dynamic Graph.
11. Graph Aesthetics.
12. Small multiplies/Animation.
14. UML notation
15. Smalltalk
### Course Objectives:
6. Students have knowledge about the basic functions of embedded systems.
7. Students have knowledge about the basic concepts of embedded systems.
8. Students have knowledge about the basic structure of embedded systems.
9. Students have knowledge about the applications of embedded systems.
10. Students have knowledge about the development of embedded software.

### UNIT I
Arm Processor Architecture Architecture, Registers, Interrupts & Vector Table, I/O Ports, ARM Processor family, JTAG, I2C bus

### UNIT II
Arm Programming Instructions Instruction Set: Data processing instructions, Addressing modes, Load Store Instructions, PSR (Program Status Register) Instructions, Conditional Instructions, Interrupt Instructions

### UNIT III
C Programming Integrated Development Environment (IDE) for C/C++ Programming, C/C++ Programs using Function Calls, Pointers, Structures, Integers & Floating Point Arithmetic, Assembly Code using Instruction Scheduling, Register Allocation, Conditional Execution & Loops

### UNIT IV
Interfacing Peripherals Interfacing: ADC & DAC, Sensors, Memory, LCD Display

### UNIT V
Stepper Motor DC Motor, SD-MMC Card, Biometric & RFID, ZIGBEE, GSM Interfaces, Debugging Tools

### References:
3. Trevor Martin, —The Insider’s Guide to The Philips ARM
Practical’s List

2. Study of Interrupt structure in ARM Processors
3. Write ARM Processor program to Flash LED
4. Interfacing of an LCD Display
5. Write a program to interface an ADC
6. Write a program to generate a Ramp waveform using DAC interface
7. Write a program to control a Stepper Motor
8. Write a program to control the speed of DC motor
9. Interface relays and write a program to control them
10. Interface ZIGBEE with ARM to control more external devices
11. Interfacing of Biometric information recorder
12. Interfacing RFID module with ARM Microcontroller
## Course Objective:

The objectives of this course are:

1. To provide fundamental details of the ERP applications.
2. To provide the knowledge of Design Pattern and Database Concepts.
3. To provide the knowledge of Service-Oriented Architecture (SOA).
4. To provide the platforms to develop the ERP applications.
5. To provide the deep knowledge of the advanced web technologies.

## Course Outcome:

1. Fundamental details of the ERP applications.
2. The knowledge of Design Pattern and Database Concepts.
3. The knowledge of Service-Oriented Architecture (SOA).
4. The platforms to develop the ERP applications.
5. The knowledge of the advanced web technologies.

### UNIT I
**Introduction**

### UNIT II
**Design Pattern:** Introduction, Creational Pattern, Structural Pattern, Behavioral Patterns. **Database Concepts:** Database Design, Enterprise Database (Oracle/DB2/MSSQL), Database Connectivity (JDBC/ODBC), Connection Pool.

### UNIT III
**Service-Oriented Architecture:** SOA Concepts and principles, XML/SOAP, Web services.

### UNIT IV
**Platform for Enterprise Solutions:** Java EE5: Java EE Platform Overview, Web Core Technologies: Servlets and JSP. Enterprise Java Bean: Enterprise JavaBean architecture, Developing EJB3.0, Session and message-driven EJBs.

### UNIT V
**Advanced Web Technology:** Web2.0 Introduction and Concepts, Rich Internet Application
Shri Vaishnav Vidyapeeth Vishwavidyalaya
Master of Technology (CSE with specialization in Big Data Analytics)
Choice Based Credit System (CBCS)-2019-20

Development, AJAX, AJAX Frameworks (Prototype Library, DWR Java Ajax Framework).

**Text Books:**

**Reference:**

**Practical’s List**

Develop an ERP application for any organization/institution.
Shri Vaishnav Vidyapeeth Vishwavidyalaya
Master of Technology (CSE with specialization in Big Data Analytics)
Choice Based Credit System (CBCS)-2019-20

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>CATEGORY</th>
<th>COURSE NAME</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CREDITS</th>
<th>THEACHING &amp; EVALUATION SCHEME</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTCS221</td>
<td></td>
<td>Network &amp; Web Security</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>60</td>
<td>Teachers Assessment*</td>
</tr>
</tbody>
</table>

Legends: L - Lecture; T - Tutorial/Teacher Guided Student Activity; P – Practical; C - Credit; *Teacher Assessment shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

Course Objectives:
1. Understand the fundamental principles of access control models and techniques, authentication and secure system design.
2. Apply methods for authentication, access control, intrusion detection and prevention.
3. Identify and mitigate software security vulnerabilities in existing systems.
4. Understand the role of firewalls, IPSec, Virtual Private Networks and identity management, etc.
5. Understand Web Server vulnerabilities and their counter measures.

Course Outcomes:
Provide students with a high level understanding of how information security functions in an organization. Topics will be both business and technology - centric.

1. To master information security governance, and related legal and regulatory issues,
2. To master understanding external and internal threats to an organization,
3. To be familiarity with information security awareness and a clear understanding of its importance,
4. To be familiar with how threats to an organization are discovered, analyzed, and dealt with,
5. To master fundamentals of secret and public cryptography,
6. To master protocols for security services,
7. To be familiar with network security threats and countermeasures.

UNIT I
Introduction to Security in Networks – Characteristics of Networks – Intrusion – Kinds of security breaches – Plan of attack - Points of vulnerability – Methods of defense – Control measures – Effectiveness of controls

UNIT II
Basic encryption and decryption, Encryption techniques, Characteristics of good encryption systems, Secret key cryptography, Data Encryption Standard, International Data Encryption Algorithm, Advanced Encryption Standard, Hash and MAC algorithms

UNIT III
Shri Vaishnav Vidyapeeth Vishwavidyalaya  
Master of Technology (CSE with specialization in Big Data Analytics)  
Choice Based Credit System (CBCS)-2019-20

Public Key encryptions, Introduction to number theory, RSA algorithm, Diffie-Hellman Digital Signature standard, Elliptic Curve cryptography, Digital signatures and authentication, Trusted intermediaries, Security handshake pitfalls

UNIT IV

UNIT V
User Authentication and session management, Cookies, Secure HTTP, SQL Injection Techniques, Cross Site Scripting, Cross-Site Request Forgery, Session Hijacking and Management, Phishing and Pharming Techniques, Web Services Security.

Text Books:

References:
1. Information security Principles and Practice by Mark Stamp, Wiley publication

Practical’s List
1. Design and implement the RSA cryptosystem.
2. Implement Digital signature scheme using RSA.
3. Simulate the Buffer overflow attack.
4. Simulate the Salami attack.
5. Design and implement a program for adding passwords to a file. The program should be able to filter out weak passwords (based on dictionary words or variants) and store the strong passwords by creating a hash of user ID and password.
6. Study of a packet sniffer like wireshark, or tcpdump. Use this tool to capture and analyze data in packets.
7. Download and install nmap. Use it with different options to scan open ports, perform OS fingerprinting, do a ping scan, tcp port scan, udp port scan, etc
8. Detect ARP spoofing using open source tool ARPWATCH
9. Install an IDS (e.g. SNORT) and study the logs.
10. Use of iptables in linux to create firewalls.
11. Implement a simple SQL injection attack.
### Course Details

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>CATEGORY</th>
<th>COURSE NAME</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CREDITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTCS223</td>
<td></td>
<td>Wireless Sensor Ad-hoc Network</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

**TEACHING & EVALUATION SCHEME**

<table>
<thead>
<tr>
<th>END SEM University Exam</th>
<th>Two Term Exam</th>
<th>Teachers Assessment</th>
<th>END SEM University Exam</th>
<th>Teachers Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teachers Assessment</td>
<td></td>
<td></td>
<td>Teachers Assessment</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>20</td>
<td>20</td>
<td>30</td>
<td>20</td>
</tr>
</tbody>
</table>

**Legends:**
- **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit;

*Teacher Assessment* shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

### Course Objective:

1. To introduce the characteristics, basic concepts and systems issues in mobile and pervasive computing
2. To illustrate architecture and protocols in pervasive computing and to identify the trends and latest development of the technologies in the area
3. To give practical experience in the area through the design and execution of a modest research project
4. To design successful mobile and pervasive computing applications and services
5. To evaluate critical design tradeoffs associated with different mobile technologies, architectures, interfaces and business models and how they impact the usability, security, privacy and commercial viability of mobile and pervasive computing services and applications

### Course Outcome:

1. To discover the characteristics of pervasive computing applications including the major system components and architectures of the systems
2. To analyze the strengths and limitations of the tools and devices for development of pervasive computing systems
3. To explore the characteristics of different types of mobile networks on the performance of a pervasive computing system.
4. To analyze and compare the performance of different data dissemination techniques and algorithms for mobile real-time applications
5. To develop an attitude to propose solutions with comparisons for problems related to pervasive computing system through investigation.

### UNIT I

UNIT II

UNIT III

UNIT IV
Single node architecture Hardware components, energy consumption of sensor nodes, Network architecture , Sensor network scenarios, types of sources and sinks, single hop versus multi-hop networks, multiple sinks and sources, design principles, Development of wireless sensor networks. physical layer and transceiver design consideration in wireless sensor networks, Energy usage profile, choice of modulation, Power Management , MAC protocols , fundamentals of wireless MAC protocols, low duty cycle protocols and wakeup concepts, contention-based protocols, Schedule-based protocols SMAC, BMAC, Traffic adaptive medium access protocol (TRAMA), Link Layer protocols fundamentals task and requirements, error control, framing, link management.

UNIT V
Gossiping and agent-based uni-cast forwarding, Energy-efficient unicast, Broadcast and multicast, geographic routing, mobile nodes, Data-centric routing , SPIN, Directed Diffusion, Energy aware routing, Gradient-based routing , COUGAR, ACQUIRE, Hierarchical Routing , LEACH, PEGASIS, Location Based Routing , GAF, GEAR, Data aggregation , Various aggregation techniques. Introduction to TinyOS , NesC, Interfaces, modules, configuration, Programming in TinyOS using

Text Books:
Practical’s List

1. Develop unicast routing protocols using any suitable Network Simulator for (Mobile Ad hoc Networks) MANET to find the best route using the any one of routing protocols from each category from table-driven (e.g., link state or DSDV) on demand (e.g., DSR, AODV, TORA), hybrid (e.g., ZRP, contact-based architectures) and hierarchical (e.g., cluster based.) The efficient path/route should be established for source and destination data transmission using routing protocols. Understand the advantages and disadvantages of each routing protocol types by observing the performance metrics of the routing protocol. In that way the best application/environment suitable routing protocol can be identified in each category.

2. Develop multicast routing protocols using any suitable Network Simulator for MANET in which session nodes are connecting through either tree(MAODV, MCEDAR) or mesh (ODMRP, CAMP, FGMP) structure. Analyze the performance metrics of multicast routing protocols with unicast routing protocols.

3. Develop MAC Protocol using any suitable Network Simulator for MANETs to send the packet without any contention through wireless link using the following MAC protocols; (CSMA/CA (802.11), MACA, MACAW, PAMAS, and SMAC). Analyze its performance with increasing node density and mobility.
Course Objective:
The course is intended to provide basic foundation with fundamental concepts and mechanisms of distributed computing systems. Most of the issues discussed in this course material are the essence of advanced operating systems.

Broad coverage as follows:
3. Introduction to distributed computing systems (DCS)
4. DCS design goals, Transparencies, Fundamental issues
5. Distributed Coordination
6. Process synchronization
7. Inter-process communication

UNIT I

UNIT II
Distributed Objects and Remote Invocation, Introduction, Communication between distributed objects, Remote procedure calls, Events and notifications Case study: Java RMI.
Operating System Support, Introduction, OS layer, Protection, Processes and threads, Communication and invocation OS architecture.

UNIT III

UNIT IV
Time and Global States, Introduction, Clocks, events and process states, Synchronizing physical clocks, Logical time and logical clocks, Global states, Distributed debugging.
Coordination and Agreement, Introduction, Distributed mutual exclusion, Elections, Multicast communication, Consensus and related problems.
UNIT V
Distributed Shared Memory, Introduction, Design and implementation issues, Sequential consistency and Ivy case study Release consistency and Minim case study, other consistency models. CORBA Case Study, Introduction, CORBA RMI, CORBA services.

Text book:

References:
UNIT I
Introduction to circuit elements R, L, C and their characteristics in terms of linearity and time dependence, KCL and KVL analysis, dual networks, analysis of magnetically coupled circuits, Dot convention, coupling co-efficient, Tuned circuits, Series and parallel resonance, voltage and current sources, controlled sources.

UNIT II
Network topology, Concept of Network graph, Tree, tree branches and links, cut set and tie set schedules. Network Theorems – Thevenin, Norton, Superposition, Reciprocity, Compensation, Maximum power transfer and Millmans theorems, problems with controlled sources.

UNIT III
Transient analysis: Transients in RL, RC and RLC circuits, initial conditions, time constants, networks driven by constant driving sources and their solutions.

UNIT IV

UNIT V

References:
1. M.E. Van Valkenburg: Network Analysis, PHI
Shri Vaishnav Vidyapeeth Vishwavidyalaya  
Master of Technology (CSE with specialization in Big Data Analytics)  
Choice Based Credit System (CBCS)-2019-20

4. Decarlo lin: Linear circuit Analysis, Oxford
5. William D Stanley : Network Analysis with Applications, Pearson Education
6. Roy Choudhary D: Network and systems, New Age Pub

Practical’s List:
1. Reciprocity Theorem
2. Maximum Power Transfer Theorem
3. Star Delta Transformation
4. Study of RC Series AC circuit
5. Study of RL Series AC circuit
6. To Measure Resonance Frequency in RLC Series Circuit
7. To Measure Resonance Frequency in RLC Series Parallel Circuit
8. To Observe Damping in RLC Series Circuit
9. Power Factor Improvement
### Shri Vaishnav Vidyapeeth Vishwavidyalaya
**Master of Technology (CSE with specialization in Big Data Analytics)**
**Choice Based Credit System (CBCS)-2019-20**

<table>
<thead>
<tr>
<th>COURSE CODE</th>
<th>CATEGORY</th>
<th>COURSE NAME</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CREDITS</th>
<th>END SEM University Exam</th>
<th>Term Exam</th>
<th>Teachers Assessment</th>
<th>END SEM University Exam</th>
<th>Teachers Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTCS207</td>
<td></td>
<td>Software Construction Lab</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>50</td>
<td></td>
</tr>
</tbody>
</table>

**Legends:**  
L - Lecture;  
T - Tutorial/Teacher Guided Student Activity;  
P - Practical;  
C - Credit;  
*Teacher Assessment* shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

**Course Objectives:**
1. Apply the fundamentals of software construction as outlines in this course to an actual software development project.
2. Demonstrate by example the key construction life cycle models
3. Interpret key practical construction considerations such as design, languages, coding, testing, quality and reuse
4. Evaluate and provide examples of the key construction technologies in a typical software construction project
5. Explain the application of software construction tools such as GUI builders, unit testing tools, profiling, performance analysis and slicing tools.

**Course Outcomes:**
1. Students will able understand tools used in software construction.
2. Students will able implement software process models & can use CASE tools.
3. Students will able prepare projects plan and implement projects.

**UNIT I**
Course Modules: Software Construction Fundamentals, Minimizing Complexity, Anticipating Change, Constructing for Verification, Reuse Standards in Construction,

**UNIT II**

**UNIT III**
Construction Testing, Construction for Reuse, Construction with Reuse, Construction Quality, Integration

**UNIT IV**
API Design and Use, Object-Oriented Runtime Issues, Parameterization and Generics, Assertions, Design by Contract, and Defensive Programming, Error Handling, Exception Handling, and Fault Tolerance, Executable Models, State-Based and Table-Driven Construction Techniques, Runtime
UNIT V
Constructing Heterogeneous Systems, Performance Analysis and Tuning, Platform Standards, Test-First Programming.
Development Environments, GUI Builders, Unit Testing Tools, Profiling, Performance Analysis, and Slicing Tools

Text Books:
2. D. L. Parnas, On the criteria to be used in decomposing systems into modules, 1972, Parnas72
3. W. Wulf and Mary Shaw, Global variable considered harmful, 1973, WulfShaw84
5. Robert C. Martin, Design principles and design patterns, Martin00.

Practical’s List:
1. Introduction to UML and Course Outlines. Tools Description.
2. Introduction to Rational Rose and Practical Implementation.
3. Introduction to class Diagram.
4. Class Diagram in Detail and Tasks Done by using Rational Rose.
5. Introduction to Use-case Diagram, its Detail and implementation by using Rational Rose.
6. Lab Quiz: 01 (Use-case Diagram)
7. Introduction to Sequence Diagram.
8. Sequence Diagram in Detail and Tasks by using Rational Rose.
10. Introduction to Collaboration Diagram and Task by using Rational Rose.
11. Test cases and Few Scenarios of test-cases in real life.
12. Introduction to TestLog and An implementation on it.
13. Lab Quiz: 02 (Test-cases)
15. Presentations based on Parser Language.