MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

# Manipal School of Information Sciences

## Manipal Academy of Higher Education, Manipal

*Outcome Based Education (OBE) Framework*

## Two Year full time Postgraduate Program

## Master of Engineering - ME (Blockchain Technology)

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

# TABLE OF CONTENTS

# Contents

# NATURE AND EXTENT OF THE PROGRAM

Blockchain is an emerging technology platform for developing decentralized applications and data storage, over and beyond its role as the technology underlying the cryptocurrencies. The basic principle of this platform is that it allows one to create a distributed and replicated ledger of events, transactions, and data generated through various IT processes with strong cryptographic guarantees of tamper resistance, immutability, and verifiability. Public blockchain platforms allow us to guarantee these properties with overwhelming probabilities even when untrusted users are participants of distributed applications with ability to transact on the platform. Even though, blockchain technology has become popularly known because of its use in the implementation of Cryptocurrencies such as BitCoin, Ethereum, etc., the technology itself holds much more promise in various areas such as time stamping, logging of critical events in a system, recording of transactions, trustworthy e-governance etc. Many researchers are working on many such use cases such as decentralized public key infrastructure, self-sovereign identity management, registry maintenance, health record management, decentralized authentication, decentralized DNS, etc. Also, corporations such as IBM and Microsoft are developing their own applications in diverse fields such as the Internet of Things (IoT), etc., even enabling blockchain platforms on the cloud.

Considering the need to disseminate the emerging concepts for engineering graduates, a program on blockchain technology is offered, so they can fit industry ready and take also take research in the area of Blockchain Technology.

Master of Engineering - ME (Blockchain Technology) postgraduate degree would welcome engineering graduates from any electrical discipline with 50% mark in qualifying exam. Students after successfully completing the program will get career opportunities as an Blockchain Architect, Blockchain application Developer, Blockchain and Blockchain Stack developer.

# PROGRAM EDUCATION OBJECTICE (PEO)

The overall objectives of the Learning Outcomes-based Curriculum Framework (LOCF) for **Master of Engineering - ME (Blockchain Technology) program are as follows.**

| PEO No | Education Objective |
|---|---|
| **PEO 1** | Engineers will possess good fundamentals, computing, and problem-solving ability, and solve the real time problems in the areas of Blockchain Technology. |
| **PEO 2** | Engineers who are innovative, bring out novel ideas in addressing the research issues and challenges and pursue their interest in research /Higher Education |
| **PEO 3** | Engineers who have leadership qualities and inclination, become entrepreneurs |

# GRADUATE ATTRIBUTES

| S No. | Attribute | Description |
|---|---|---|
| 1 | **Scholarship of Knowledge** | Acquire in-depth knowledge of specific discipline or professional area, including wider and global perspective, with an ability to discriminate, evaluate, analyse and synthesise existing and new knowledge, and integration of the same for enhancement of knowledge. |
| 2 | **Critical Thinking** | Analyse complex engineering problems critically, apply independent judgement for synthesising information to make intellectual and/or creative advances for conducting research in a wider theoretical, practical and policy context. |
| 3 | **Problem Solving** | Think laterally and originally, conceptualise and solve engineering problems, evaluate a wide range of potential solutions for those problems and arrive at feasible, optimal solutions after considering public health and safety, cultural, societal and environmental factors in the core areas of expertise. |
| 4 | **Research Skill** | Extract information pertinent to unfamiliar problems through literature survey and experiments, apply appropriate research methodologies, techniques and tools, design, conduct experiments, analyse and interpret data, demonstrate higher order skill and view things in a broader perspective, contribute individually/in group(s) to the development of scientific/technological knowledge in one or more domains of engineering. |
| 5 | **Usage of modern tools** | Create, select, learn and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modelling, to complex engineering activities with an understanding of the limitations. |
| 6 | **Collaborative and Multidisciplinary work** | Possess knowledge and understanding of group dynamics, recognise opportunities and contribute positively to collaborative-multidisciplinary scientific research, demonstrate a capacity for self-management and teamwork, decision-making based on open-mindedness, objectivity and rational analysis in |

| | | |
|---|---|---|
| | | order to achieve common goals and further the learning of themselves as well as others. |
| **7** | **Project Management and Finance** | Demonstrate knowledge and understanding of engineering and management principles and apply the same to one's own work, as a member and leader in a team, manage projects efficiently in respective disciplines and multidisciplinary environments after consideration of economical and financial factors. |
| **8** | **Communication** | Communicate with the engineering community, and with society at large, regarding complex engineering activities confidently and effectively, such as, being able to comprehend and write effective reports and design documentation by adhering to appropriate standards, make effective presentations, and give and receive clear instructions. |
| **9** | **Life-long Learning** | Recognise the need for, and have the preparation and ability to engage in life-long learning independently, with a high level of enthusiasm and commitment to improve knowledge and competence continuously. |
| **10** | **Ethical Practices and Social Responsibility** | Acquire professional and intellectual integrity, professional code of conduct, ethics of research and scholarship, consideration of the impact of research outcomes on professional practices and an understanding of responsibility to contribute to the community for sustainable development of society. |
| **11** | **Independent and Reflective Learning** | Observe and examine critically the outcomes of one's actions and make corrective measures subsequently, and learn from mistakes without depending on external feedback. |

# QUALIFICATIONS DESCRIPTORS

1. Demonstrate

   (i) A systematic, extensive, coherent knowledge and understanding of an academic field of study, as a whole, its applications, and links to related disciplinary areas/subjects of study; including a critical understanding of the established theories, concepts, number of advanced and emerging issues in the field of Blockchain Technology.

   (ii) Procedural knowledge that creates different types of professionals related to the Blockchain Technology, including research and development, teaching, government and public service.

   (iii) Professional skills in the domain of blockchain technology with the knowledge of developing applications deployment, and testing using the modern tools and programming languages.

2. Demonstrate comprehensive knowledge on cryptocurrencies, blockchain applications, network security, cryptographic algorithms, web application development, data structures and operating systems.

3. Demonstrate skills in identifying information needs, collection of relevant quantitative and/or qualitative data drawing on a wide range of sources, analysis and interpretation of data.

4. Methodologies as appropriate to the subject(s) for formulating evidence based solutions and arguments.

5. Use knowledge, understanding and skills for critical assessment of a wide range of ideas and complex problems and issues relating to the chosen field of study.

6. Communicate the results of studies undertaken in an academic field accurately in a range of different contexts using the main concepts, constructs and techniques of the Blockchain studies.

7. Address one's own learning needs relating to current and emerging areas of study, making use of research, development and professional materials as appropriate, including those related to new frontiers of knowledge.

8. Apply one's disciplinary knowledge and transferable skills to new/unfamiliar contexts, to identify, analyse problems, issues, and seek solutions to real-life problems.

# PROGRAM OUTCOMES

**After successful completion of Master of Engineering - ME (Blockchain Technology), Students will be able to:**

| PO No | Attribute | Competency |
|---|---|---|
| **PO 1** | **Scholarship of Knowledge** | Acquire knowledge to build applications for blockchain technology domain of Blockchain Technology. |
| **PO 2** | **Critical Thinking** | Identify, formulate, analyze, and solve real-life problems with Blockchain Technology. |
| **PO 3** | **Problem Solving** | Identify. formulate, review research aspects and analyse issue in the domain of Blockchain Ecosystem using various aspects like Blockchain Application Development, Blockchain Testing and Automation, Blockchain Deployment. |
| **PO 4** | **Research Skill** | To identify problems related to blockchain and able to provide design solutions from literature study. |
| **PO 5** | **Usage of modern tools** | Create, select, and apply appropriate techniques, resources, for building and deploying Blockchain applications and tools. |
| **PO 6** | **Collaborative and Multidisciplinary work** | To identify problems, collaborate with researchers in the field of healthcare, banking, government, gaming and many more and provide solutions using blockchain technology. |
| **PO 7** | **Project Management and Finance** | Demonstrate knowledge and understanding of Blockchain Technology and apply to architect and develop blockchain applications, principles as a member and leader in a team, to manage projects and in multidisciplinary environments. |
| **PO 8** | **Communication** | Communicate effectively with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions. |

| | | |
|---|---|---|
| **PO 9** | **Life-long Learning** | Ability to develop professional skills to prepare them for immediate employment and for lifelong learning in advanced areas of Blockchain and its applications. |
| **PO 10** | **Ethical Practices and Social Responsibility** | Apply ethical principles and commit to professional ethics and responsibilities and norms of the Blockchain Technology and security practices. |
| **PO 11** | **Independent and Reflective Learning** | Observe and examine critically the outcomes of one's actions and make corrective measures subsequently and learn from mistakes without depending on external feedback. |

MANIPAL ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

# COURSE STRUCTURE, COURSEWISE LEARNING OBJECTIVE, AND COURSE OUTCOMES (COS)

## FIRST YEAR: ME (Blockchain Technology)

**Semester: 1** **Semester: 2**

| Subject Code | Subject Title | L | T | P | C | Subject Code | Subject Title | L | T | P | C |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CSE 601 | Data Structures and Algorithms | 3 | - | - | 3 | CDC 607 | DevOps for Cloud | 3 | - | - | 3 |
| BCH 601 | Blockchain Application Development | 3 | - | - | 3 | BCH 602 | Network Security and Analysis | 3 | - | - | 3 |
| BCH 603 | Blockchain Technology | 3 | - | - | 3 | BCH 604 | Advanced Blockchain Application Development | 3 | - | - | 3 |
| BCH 605 | Cryptocurrency and Smart Contracts | 3 | - | - | 3 | BCH 606 | Blockchain Verification and Testing | 3 | - | - | 3 |
| | Elective - 1 | 3 | - | - | 3 | | Elective - 2 | 3 | - | - | 3 |
| CSE 601L | Data Structures and Algorithms Lab | - | - | 3 | 1 | CDC 607L | DevOps for Cloud Lab | - | - | 3 | 1 |
| BCH 601L | Blockchain Application Development Lab | - | - | 3 | 1 | BCH 602L | Network Security and Analysis Lab | - | - | 3 | 1 |
| BCH 603L | Blockchain Technology Lab | - | - | 3 | 1 | BCH 604L | Advanced Blockchain Application Development Lab | - | - | 3 | 1 |
| BCH 605L | Cryptocurrency and Smart Contracts Lab | - | - | 3 | 1 | BCH 606L | Blockchain Verification and Testing Lab | - | - | 3 | 1 |
| | Elective - 1 Lab | - | - | 3 | 1 | | Elective - 2 Lab | - | - | 3 | 1 |
| BCH 695 | Mini Project - 1 | - | - | 4 | - | IOT 696 | Mini Project - 2 | - | - | - | 4 |
| BCH 697 | Seminar - 1 | - | - | 1 | - | IOT 698 | Seminar - 2 | - | - | - | 1 |
| **Total** | | **15** | **-** | **15** | **25** | **Total** | | **15** | **-** | **15** | **25** |

## SECOND YEAR (FINAL YEAR): ME (Blockchain Technology)

| **III and IV Semester** | | |
|---|---|---|
| BCH 799 | Project Work | 25 |
| **Total Number of Credits to Award Degree** | | 75 |

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

## List of Electives(Theory)

| Elective - 1 | | Elective - 2 | |
|---|---|---|---|
| **Code** | **Subject** | **Code** | **Subject** |
| BCH-615 | Distributed Computing and Databases | BCH-616 | Cloud Computing |
| BDA-623 | Architecture of Big Data Systems | BCH-617 | Machine Learning |
| CSE-604 | Database Programming in Java | ENP-601 | Entrepreneurship |

## List of Electives(Lab)

| Elective - 1 | | Elective - 2 | |
|---|---|---|---|
| **Code** | **Subject** | **Code** | **Subject** |
| BCH-615L | Distributed Computing and Databases Lab | BCH-616L | Cloud Computing Lab |
| BDA-623L | Architecture of Big Data Systems Lab | BCH-617L | Machine Learning Lab |
| CSE-604L | Database Programming in Java Lab | ENP-601L | Entrepreneurship Lab |

**Name of the Institution / Department**: Manipal School of Information Sciences (MSIS)

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Data Structures and Algorithms |
| **Course Code:** CSE 601 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 3 | **Prerequisites:** Basic Programming – preferably C |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>1. This course introduces students to elementary data structures and design of algorithms.<br>2. Students learn how to design optimal algorithms with respect to time and space<br>3. Students learn how to implement link list, stack, queues, searching and sorting techniques, sets, trees and graphs.<br>4. Students learn the design of divide and conquer technique, dynamic programming, greedy technique and back tracking. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Specify and analyse algorithms. |
| **CO 2:** | Learn and design programs for implementation of linear and non linear data structure. |
| **CO 3:** | Learn and design programs for sorting and searching. |
| **CO 4:** | Illustrate application of divide and conquer technique, dynamic programming, greedy technique and back tracking. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | | * | | | | | | | |
| CO 2 | * | * | | | | * | | | | | |
| CO 3 | * | | | | | * | | | | | |
| CO 4 | * | * | | | | * | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|

| **Unit 1: Introduction** | |
|---|---|
| Algorithm Specification, Performance Analysis | At the end of the topic student should be able to:<br>1. Define algorithms (C1)<br>2. Analyse algorithms. (C6) |
| **Unit 2: Algorithm Analysis Techniques** | |
| Analysis of Recursive Programs, Solving Recurrence Equations, General Solution for a large class of Recurrences. | 1. Define recursive programs (C2)<br>2. Design simple recursive programs (C6)<br>3. Solve recurrence relations (C6) |
| **Unit 3: Elementary data structures** | |
| Implementation of Lists, Stacks, Queues | 1. Design singly linked list (C6)<br>2. Design doubly linked list(C6)<br>3. Explain the concepts of array-based stacks (C2)<br>4. Explain the concepts of pointer-based stacks (C2)<br>5. Design and implement Queues. (C6) |
| **Unit 4: Sorting & Searching Techniques** | |
| Quick sort, Heap sort, Merge sort, Binary search, linear search, Fibonacci search | 1. Develop algorithm for insertion sort, bubble sort and selection sort. (C6)<br>2. Develop and analyse algorithm for quick sort (C6)<br>3. Develop and analyse algorithm for heap sort (C6)<br>4. Develop and analyse algorithm for merge sort (C6)<br>5. Design and analyse algorithms for binary, linear and Fibonacci search (C6) |

| **Unit 5: Operations on Sets** | |
|---|---|
| Introduction to Sets, A Linked- List implementation of Set, The Dictionary, The Hash Table Data Structure | 1. Develop data structures for sets (C6)<br>2. Design a linked list-based implementation of sets (C6)<br>3. Design a Dictionary (C6)<br>4. Design Data structure for hash table (C6) |
| **Unit 6: Trees** | |
| Basic Terminology, Implementation of Trees, Binary Trees, Binary Search Trees | 1. Examine the concepts of trees. (C3)<br>2. Design and implement general trees (C6)<br>3. Design and implement binary trees (C6)<br>4. Design and implement binary search trees (C6) |
| **Unit 7: Graphs** | |
| Basic definitions, Representation of Graphs, Minimum Cost Spanning Tree, Single Source Shortest Paths,<br>All-Pairs Shortest Path | 1. Define graphs (c6)<br>2. Design data structure for graphs (c6)<br>3. Formulate an algorithm to solve minimum cost spanning tree(c6)<br>4. Formulate an algorithm to solve Single source shortest path (c6)<br>5. Formulate an algorithm to solve All-pair shortest path(c6) |
| **Unit 8: Algorithm Design Techniques** | |
| Divide-and-Conquer Algorithms, Dynamic Programming, Greedy Algorithms, Backtracking | 1. Design of divide and conquer algorithms (C6)<br>2. Solve max min, Strassen's matrix multiplication, multiplication of long integers problem. (C6)<br>3. Design of dynamic programming techniques (C6) |

| | |
|---|---|
| | 4. Solve matrix chain order problem (C6)<br>5. Design of greedy algorithms(C6)<br>6. Solve Knap-sack, job scheduling with deadlines and optimal storage on tapes problems. (C6)<br>7. Design of Back tracking algorithms (C6) |

| **Learning strategies, contact hours and student learning time** | | |
|---|---|---|
| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
| Lecture | 30 | 60 |
| Seminar | - | - |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

| **Assessment Methods:** | |
|---|---|
| **Formative:** | **Summative:** |
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

| **Mapping of assessment with Cos** | | | | |
|---|---|---|---|---|
| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
| Sessional Examination 1 | * | * | | |

| | | | | |
|---|---|---|---|---|
| Sessional Examination 2 | | * | * | * |
| Assignment/Presentation | * | * | * | * |
| End Semester Examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | • End-Semester Feedback |
| **Reference Material** | 1. "Introduction to Algorithms" Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest.<br>2. "Data Structures& Algorithms" Aho, Hopcroft and Ulmann<br>3. "Data structures and algorithm analysis in C" Mark Allen Weiss<br>4. "Computer Algorithms" : Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Blockchain Application Development |
| **Course Code:** BCH 601 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits: 3** | **Prerequisites: -** |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on This Course provides insight on depth practical knowledge in Blockchain technology. The basic understanding of programming required for building blockchain application using Node.js. This course allows students to have a fundamental knowledge on HTML, CSS and JavaScript for developing blockchain or any web applications. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | List and understand the blockchain applications |
| **CO 2:** | To identify the programming languages used for blockchain development |
| **CO 3:** | Plan and prepare the fundamentals of programming required for blockchain development. |
| **CO 4:** | Solve to write application in general or specific to blockchain |
| **CO5:** | Model and Construct the blockchain application |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | * | | * | | | | | | |
| CO 2 | * | * | * | | * | | | | | | |
| CO 3 | | | | | | | * | | | | |
| CO 4 | | | * | * | | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| Introduction to applications – Types of Applications – Application vs | 1. To explain the concept of web applications (C2) |

| | |
|---|---|
| Implementation – Functional vs Non-functional – Programming languages in Blockchain – Blockchain Platforms Introduction to Web Applications – Web Servers – HTML vs HTML 5 – HTML FORMS – CSS – CSS3 | 2. To identify the blockchain platforms for application development (C1)<br>3. To illustrate applications using HTML and CSS (C3) |
| **Unit 2:** | |
| JavaScript – JQUERY – AJAX | 1. To explain the constructs of JAVASCRIPT (C2)<br>2. To illustrate applications with JavaScript , jQuery and AJAX (C3) |
| **Unit 3:** | |
| Node.js – Constructs of Node.js – Classes – Objects - Creating blockchain with node.js | 1. To explain the constructs in Node.js (C2)<br>2. To explain the concepts of OOPS in node.js (C2)<br>3. To illustrate applications with node.js (C3) |
| **Unit 4:** | |
| Introduction to Databases – Database Tools – Databases in Blockchain - ER diagrams - Normalization – SQL – NOSQL | 1. To explain the concepts involved in databases (C2)<br>2. To describe the concepts of ER diagrams (C2)<br>3. To illustrate the working of SQL and NO SQL tools (C3) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Seminar | - | - |
| Quiz | 02 | 04 |

| | | |
|---|---|---|
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | * |
| Assignment/Presentation | * | * | * | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Learn Blockchain Programming with JavaScript: Build your very own Blockchain and decentralized network with JavaScript and Node.js**, Eric Traub, Packt Publishing, 2018.<br>2. **HTML 5 Black Book (Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP, jQuery)**, DT Editorial Services, Dreamtech Press, Second edition,2016.<br>3. **HTML & CSS: The Complete Reference**, Thomas Powell, McGraw Hill Education, Fifth Edition, 2017.<br>4. **Beginning Node.js**, Basarat Ali Syed, Apress, 2014. |

| | |
|---|---|
| | 5. **Node.Js Web Development**, David Herron, Ingram short title; 3rd Revised edition, 2016.<br>6. **Mastering Node.js,** Sandro Pasquali, Kevin Faaborg, Packt Publishing Limited; 2nd Revised edition, 2017<br>7. **Full-Stack JavaScript Development: Develop, Test and Deploy with MongoDB, Express, Angular and Node on AWS**, Eric Bush, Red Sky, 2016.<br>8. **Blockchain Applications: A Hands-On Approach**, Arshdeep Bahga, Vijay Madisetti, VPT, 1 edition, 2018. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Blockchain Technology |
| **Course Code:** BCH 603 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits: 3** | **Prerequisites:** Basic Network Concepts |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on understanding the working of blockchain technology and how blockchain platform works. The course discuss on the nuances involved in blockchain technology and its implementation on the blockchain platform. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Outline the characteristics of the blockchain ecosystem. |
| **CO 2:** | Develop the blockchain ecosystem using Ethereum. |
| **CO 3:** | Evaluate the application based on Ethereum. |
| **CO 4:** | Examine the development process using Hyperledger. |
| **CO 5:** | Demonstrate the blockchain application development process. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | * | | | | | | | |
| CO 2 | * | | | | * | | | | | | |
| CO 3 | * | * | * | | * | | | | | | |
| CO 4 | * | * | | | | | | | | | |
| CO 5 | * | | * | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| Introduction to Blockchain - Potential of Blockchain – Defining Blockchain – | At the end of the topic student should be able to: |

| | |
|---|---|
| Ownership – Understanding Ledger – Ledger Structure – Concepts of Ownership – Centralized vs Decentralized - Components of a Blockchain -Characteristics of Blockchain - The growth of blockchain technology - Distributed systems - The history of blockchain and Bitcoin - Types of blockchain – Consensus - CAP theorem and blockchain - Decentralization using blockchain Methods of decentralization- Routes to decentralization - Blockchain and full ecosystem decentralization- Smart contracts- Decentralized Organizations. | 1. To describe the potential of blockchain and its architecture. (C2)<br>2. To describe the relation between blockchain and smart contracts. (C2)<br>3. To describe the consensus and CAP theorem. (C2) |
| **Unit 2:** | |
| Ethereum and working with Smart Contracts : Understand Ethereum ,Define Smart Contracts,Identify Cryptocurrency used in Ethereum,Describe Transactions in Ethereum,Define Consensus Mechanism in Ethereum,List Development Technologies,Identify Ethereum Clients, Define Platform Functions,Understand Solidity, Describe Solidity Operators and Functions, Setting up Metamask,How to interface with ethereum network,First smart contract,Ethereum accounts and how to receive ether, Structuring a | 1. To describe the working of the smart contracts (C2)<br>2. To illustrate the concepts involved in Ethereum and its development (C2)<br>3. To describe the creation of applications using solidity. (C2) |

| | |
|---|---|
| contract,Declaring a function, Deploying and redeploying of a contract,Comparing Wei & Ether,What is a gas transaction,Remix testing. | |
| **Unit 3:** | |
| Hyperledger : Define Hyperledger Blockchain , Understand Hyperledger Consensus Algorithm ,Explain Hyperledger Iroha ,Identify Hyperledger Components ,Describe Setting up Channels Policies ,Chaincodes List Hyperledger , Explorer Components ,Define Hyperledger Composer,Fabric Under the Hood (Concepts & Terminology),Ledger Implementation,Dev Environment Walkthrough: Peer & CouchDB setup,Ledger Implementation,Peers Nodes : Anchors and Endorsers, Anchor Peers & Endorsing Peers,Clients Node: Endorsement Policies, Client Peer & Endorsing Policies Orderer Nodes,Membership Service Provider & Certification Authority,Dev Environment Walkthrough: Orderer and CA Server,Chaincode Development. | 1. To explain the concepts of Hyperledger (C2)<br>2. To identify different components in Hyperledger (C2)<br>3. To illustrate the examples of Hyperledger (C3) |
| **Unit 4:** | |
| Creating private Blockchain with Multichain : Define Multichain , | 1. To define and describe the multichain blockchain (C2) |

| | |
|---|---|
| Describe MultiChain Streams , Create & deploy private blockchain ,Explain Connecting to a Blockchain ,Identify Multichain Interactive Mode ,List Native assets ,Define Transaction Metadata ,Explain Streams Explain Mining ,Bitcoin to private blockchain,Aim of multichain,Hand-shake process,Multi-chain use cases,Multichain permission,Multichain assets,multichain streams,Basics of retrieving from streams,Consensus model,Multichain flexibility,Deployment options,Speed and scalability of multichain | 2. To explain the mining in multichain process (C3)<br>3. To illustrate the deployment of multichain blockchain and its applications (C2) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Seminar | - | - |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

| **Assessment Methods:** | |
|---|---|
| **Formative:** | **Summative:** |
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | | |
| Sessional Examination 2 | | | * | * | * |
| Assignment/Presentation | * | * | * | * | * |
| End Semester Examination | * | * | * | * | * |
| Laboratory examination | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Blockchain Basics: A Non-Technical Introduction in 25 Steps**, Daniel Drescher, Apress; 1st Edition, 2017.<br>2. **Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions**, Bikramaditya Singhal, Gautam Dhameja , Priyansu Sekhar Panda, Apress; 1st ed. Edition, 2018.<br>3. **Mastering Blockchain**, Imran Bashir, Ingram short title, Second Edition, 2018.<br>4. **Hands-On Blockchain with Hyperledger**, Petr Novotny Venkatraman Ramakrishna Nitin Gaur Anthony O'Dowd Luc Desrosiers, Ingram short title, 2018.<br>5. **Solidity Programming Essentials**, Ritesh Modi, Ingram short title, 2018<br>6. **BlockChain from Concept to Execution**, Debajani Mohanty, BPB; 2nd revised and updated edition, 2018.<br>7. **Mastering Blockchain Programming with Solidity: Write production-ready smart contracts for Ethereum blockchain with Solidity**, Jitendra Chittoda, Packt Publishing Limited, 2019. |

8. **Hands-On Blockchain with Hyperledger**, Petr Novotny Venkatraman Ramakrishna Nitin Gaur Anthony O'Dowd Luc Desrosiers, Ingram short title, 2018
9. **Blockchain for Dummies**, Tiana Laurence, 2nd edition – 2019.
10. **Hands-On Smart Contract Development with Solidity and Ethereum: From Fundamentals to Deployment**, David Hoover, Kevin Solorio, Randall Kanna, Shroff/O'Reilly; First edition, 2019.
11. **Blockchain By Example: A developer's guide to creating decentralized applications using Bitcoin, Ethereum, and Hyperledger**, Bellaj Badr , Richard Horrocks, Xun (Brian) Wu , Packt Publishing Limited, 2018.
12. **Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners**, Chris Dannen, APRESS, 1 edition, 2017.
13. **Ethereum: Blockchains, Digital Assets, Smart Contracts, Decentralized Autonomous Organizations**, Henning Diedrich, CreateSpace Independent Publishing Platform; 1st edition, 2016.

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Cryptocurrency and Smart Contracts |
| **Course Code:** BCH 605 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 3 | **Prerequisites:** Cryptography Basics, Networking Basics, Programming aspects |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>Discuss the implementation of cryptocurrencies.<br>Understand main blockchain concepts like Proof-of-Work, mining, peer-to-peer connections, etc.<br>Build their own blockchain and cryptocurrency.<br>Design, test, and deploy secure Smart Contracts |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | To outline the details cryptography and its basic algorithms. |
| **CO 2:** | To identify and design and program smart contracts. |
| **CO 3:** | Illustrate user-defined operations of arbitrary complexity. |
| **CO 4:** | Design Code Deploy and Execute a Smart Contract. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | | | | | | | | | |
| CO 2 | | * | * | * | * | | | | | | |
| CO 3 | * | * | * | | * | | | | | | |
| CO 4 | | | * | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1: Introduction** | |
| Introduction to Security: Need for Security, CIA triad, Services – Mechanisms and Attacks, Classic Encryption Techniques | At the end of the topic student should be able to: |

| | |
|---|---|
| - Substitution cipher - Transposition cipher, Characteristic of Cryptographic Systems- Modern Encryption Techniques, symmetric key, asymmetric key, PKI and Key Management, block cipher, stream cipher, Hashing | 1. To identify the encryption and decryption techniques (C1)<br>2. To infer key management and Hashing technique in cryptography. (C3) |
| **Unit 2:** | |
| Introduction to Bitcoin - Background on Digital Payment - Bitcoin Protocol specification - Security Transactions in Bitcoin - Applications/Extensions of Bitcoin | 1. To explain the working of the bitcoin (C2)<br>2. To illustrate security transactions, happen in bitcoin. (C3) |
| **Unit 3:** | |
| Identify Bitcoin & How to Get Bitcoins ,Identify Bitcoin Wallets ,Identify Wallet , Define Selling Bitcoins , Compare Bitcoin Blockchain , Transaction & Transaction Script , Describe Various Transaction Forms in Bitcoin,Define Scripts in Bitcoin ,List Nodes in Bitcoin Network. | 1. To identify bitcoin and examine the procedure of selling and purchasing bitcoins. (C1)<br>2. To describe the various Transactions (C2)<br>3. To write scripts for Bitcoin (C3) |
| **Unit 4:** | |
| Understand Economics of Bitcoin , Define Bitcoin , Mining Describe , Fabrication of a Block Header ,Define Mining Identify ,Successful Mining List ,Difficulties in Solo Mining ,Understand Mining ,By pool of Miners,Mining and consensus ,Autonomous verification of mining,Independent verification of mining,Checklist for verification of mining,Combining transactions into blocks,Combination of verified transactions,Combining transactions into blocks,Portrayal of difficulty,Condition of difficulty,Creation of block header,Main chain and Orphan | 1. To describe and discuss on the miners and mining process (C2)<br>2. To test the mining process (C4)<br>3. To describe the types of miners. (C2) |

| | |
|---|---|
| block,Creation of new block,Independent validation of new block,The race for bitcoin mining and hash race,Difficulty rising with the hashing power of the miners,Solo mining and difficulty of solo mining,Benefits of pooled mining,Types of mining pools. | |
| **Unit 5:** | |
| Introduction to Smart Contracts – History - Definition – Advantages of Smart Contracts - Ricardian contracts – Creating Smart Contracts with Ethereum Ecosystem | 1. To describe and create smart contracts (C1)<br>2. Define smart contacts with Ethereum (C1)<br>3. Discuss the advantage and disadvantages of smart contacts(C2) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Seminar | - | - |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |

| | |
|---|---|
| **Mapping of assessment with Cos** | |

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | |
| Assignment/Presentation | | * | * | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | | * | | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Cryptography and Network Security - Principles and Practice, Stallings William**, Pearson Education; Seventh edition, 2017<br>2. **Cryptography And Network Security**, Forouzan, McGraw Hill Education, 2015<br>3. **Cryptography and Network Security**, Atul Kahate, McGraw Hill Education; Third edition, 2017.<br>4. **The Age of Cryptocurrency: How Bitcoin and Digital Money Are Challenging the Global Economic** , Paul Vigna , Michael J. Casey, St. Martin's Press, 2015.<br>5. **Mastering Ethereum: Building Smart Contracts and DApps, Andreas M. Antonopoulos**, Dr. Gavin Wood, , Shroff/O'Reilly; First edition, 2018<br>6. **Mastering Bitcoin: Programming the Open Blockchain**, Andreas M. Antonopoulos, Shroff/O'Reilly; Second edition, 2017.<br>7. **Programming Bitcoin: Learn How to Program Bitcoin from Scratch**, Jimmy Song, O'Reilly Media, 1 edition 2019.<br>8. **Blockchain for Business with Hyper-ledger Fabric**, Nakul Shah, BPB Publications; 1st edition, 2019.<br>9. **Hands-On Cybersecurity with Blockchain**, Rajneesh Gupta , Packt Publishing; 1st edition, 2018. |

| | |
|---|---|
| | 10. **Ethereum Smart Contract Development: Build blockchain-based decentralized applications using solidity**, Mayukh Mukhopadhyay, Packt Publishing Limited, 2018.<br>11. **Building Blockchain Projects**, Narayan Prusty, Packt Publishing Limited, 2017.<br>12. **Hands-On Bitcoin Programming with Python: Build powerful online payment centric applications with Python**, Harish Garg, Packt Publishing Limited, 2018.<br>13. **Bitcoin and Cryptocurrency Technologies – A Comprehensive Introduction Hardcover**, Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller , Steven Goldfeder,Princeton University Press, 2016.<br>14. **Fundamentals of Smart Contract Security**, Richard Ma, Jan Gorzny, Edward Zulkoski, Momentum Press, 2019. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Distributed Computing and Databases |
| **Course Code:** BCH-615 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 3 | **Prerequisites:** Introductory Course in IoT, Networking Basics, Programming aspects, Operating system, Linux |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>The course allows students to have a good understanding on how distributed system allows resource sharing, including software by systems connected to the network. The course also discusses on the distributed databases and how they help in storing data. The related tools and programming using C/JAVA is discussed on the practical aspects of the disturbed computing. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | To identify and introduce concepts related to distributed computing systems. |
| **CO 2:** | Identify the introductory distributed database concepts and its structures. |
| **CO 3:** | Describe terms related to distributed object database design and management. |
| **CO 4:** | Produce the transaction management and query processing techniques in DDBMS. |
| **CO 5:** | Relate the importance and application of emerging database technology. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | * | * | | | | | | |
| CO 2 | * | * | | * | * | | | | | | |
| CO 3 | * | | | | * | | | | | | |
| CO 4 | * | * | * | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |

| | |
|---|---|
| Introduction – Characterization of Distributed Systems – Examples - Resource Sharing and the Web- Challenges – Distributed System Models – Architectural, Fundamental - Interprocess Communication - API for Internet protocols – Message-Oriented Communication – Unicasting - Multicasting and Broadcasting - Client-Server communication – Group Communication. | At the end of the topic student should be able to:<br>1. To outline the characterization of the distributed systems (C1)<br>2. To describe the architecture of the distribute systems(C2)<br>3. To explain the interprocess communications (C5)<br>4. To explain the client-server architecture (C5) |
| **Unit 2:** | |
| Distributed Operating Systems: Issues in Distributed Operating System – Remote Invocation - Communication between distributed objects - Remote procedure calls -Threads in Distributed Systems – Clock Synchronization – Election Algorithms – Distributed Mutual Exclusion – Distributed Transactions – Distributed Deadlock – Agreement Protocols. | 1. To outline the characteristics of operating systems. (C1)<br>2. To examine the threads in distributed systems(C5)<br>3. To explain the exclusion, deadlock in distributed systems (C5) |
| **Unit 3:** | |
| Distributed File Systems: Introduction - File service architecture - File System-Enhancements and further developments - Name Services: Introduction - Name Services and the Domain Name System - Directory Services - Time and Global States - Clocks, events and process states | 1. To review the distributed file systems (C2)<br>2. To explain the domain name system and services (C5)<br>3. To examine the time and clocks involved In Distributed systems. (C5) |

| | |
|---|---|
| Synchronizing physical clocks - Logical time and logical clocks - Distributed debugging - Coordination and Agreement-Introduction - Distributed mutual exclusion – Elections - Multicast communication-Consensus and related problems | |
| **Unit 4:** | |
| Distributed Shared Memory: Introduction - Design and implementation issues - Sequential consistency and Ivy case study - Release consistency and Munin case study - Other consistency models - Introduction to Fault Tolerance - Distributed Commit Protocols – Byzantine Fault Tolerance – Impossibilities in Fault Tolerance. | 1. To explain the shared memory concepts in distributed computing. (C2)<br>2. To describe the consistent models (C2)<br>3. To explain the fault tolerance and its types. (C5) |
| **Unit 5:** | |
| Introduction to Databases –Structured vs Unstructured vs SemiStructured data – RDBMS – Entity Relationships – SQL – NOSQL – Distributed Databases – Graph Databases – Tools | 1. To explain the type of databases (C2)<br>2. To describe the entity relationships (C2)<br>3. To examine the graph databases and its tools (C4) |
| **Unit 6:** | |
| Security In Databases: Security requirements of database systems – Reliability and Integrity in databases – Redundancy – Recovery – Concurrency/ Consistency – Monitors – Sensitive Data – Types of disclosures – | 1. To Describe the security issues in database systems. (C2)<br>2. To describe the security issues in Databases (C2)<br>3. To Describe the process of SQL injection. (C3) |

| | |
|---|---|
| Inference-Finding and Confirming SQL injection | |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Seminar | - | - |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | |
| Assignment/Presentation | | * | * | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |

| | |
|---|---|
| **Reference Material** | 1. **SQL injection Attacks and defense**, Justin Clarke, Syngress, 2nd edition,2012<br>2. **Distributed Systems Concepts and Design**, George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair, Pearson, 5th edition, 2011.<br>3. **Distributed Systems Principles and Paradigms**, A.S.Tanenbaum, M.Van Steen, PHI, 2012.<br>4. **Distributed Computing: Fundamentals, Simulations and Advanced Topics**, Hagit Attiya and Jennifer Welch, Wiley, 2004.<br>5. **Distributed Computing Principles and Applications**, M.L.Liu, Pearson Addison Wesley, 2004.<br>6. **Graph Databases: New Opportunities for Connected Data**, Ian Robinson, Jim Webber, Emil Eifrem, Shroff/O'Reilly, 2nd Edition, 2016.<br>7. **Graph Algorithms: Practical Examples in Apache Spark & Neo4j**, Mark Needham, Amy E. Hodler, Shroff/O'Reilly; First edition, May 2019. |

MANIPAL ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Architecture of Big Data Systems |
| **Course Code:** BDA 623 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 3 | **Prerequisites:** - |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>1. Students learn the concept of big data characteristics, batch and lambda architecture.<br>2. This course introduces students to basics file systems in Big Data<br>3. This course helps the student to understand the concepts of Hadoop framework, Spark framework and their internals.<br>4. This course helps the students to learn Map-reduce programming, Spark programming.<br>5. Students learn the different layers with use cases demonstrations. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Examine the type of data in big data. |
| **CO 2:** | To design applications based with Hadoop framework. |
| **CO 3:** | To design applications based with spark architecture. |
| **CO 4:** | To build applications based on the Big Data Architecture platforms and analyse the results based on the outcome of the applications used. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | * | | | * | | | | | |
| CO 2 | * | * | * | | * | | * | | | * | |
| CO 3 | * | * | * | | * | | * | | | * | |
| CO 4 | * | * | * | | * | * | * | | | * | |

**Course content and outcomes:**

| *Content* | *Competencies* |
|---|---|
| **Unit 1: Classifying Big Data Characteristics** | |
| Analysis type - real time or batched for later analysis.<br>Processing methodology - predictive, analytical, ad-hoc query, and reporting.<br>Data frequency and size<br>On demand, as with social media data<br>Continuous feed, real-time - weather data, transactional data<br>Time series - time-based data<br>Data type - transactional, historical, master data and metadata.<br>Content formats - structured, unstructured, semi-structured<br>Data sources - Web and social media, humans, machines, transaction data and biometric data. | 1. Identify different types of Data (C1)<br>2. Identify processing methodology (C1) |
| **Unit 2: Big Data processing - the Lambda architecture** | |
| Append-only, immutable data<br>Batch layer<br>Serving layer<br>Speed layer<br>Case study: Druid - A Real-time Analytical Data Store | 1. Understand Lambda architecture to handle Big Data (C2).<br>2. Understand different layers in Lambda Architecture (C2). |
| **Unit 3 Batch layer, Serving layer and Speed layer** | |
| Choosing a storage solution for the batch layer: Distributed file systems, Vertical partitioning. | 1. Develop applications to store data in HDFS (C4). |

| | |
|---|---|
| MapReduce: a paradigm for Big Data computing.<br>Performance metrics for the serving layer<br>Requirements for a serving layer database<br>Computing real time views<br>Storing real time views<br>Challenges of incremental computation<br>Asynchronous versus synchronous updates | 2. Develop applications for batch processing using Map Reduce technique (C4).<br>3. Understand the need of serving layer (C2).<br>4. Design application to store data for processing in serving layer (C4).<br>5. Understand the need of Speed layer for data processing (C2). |
| **Unit 4: Spark: Alternatives to MapReduce** | |
| Spark Architecture<br>Spark Session<br>DataFrame<br>Transformations and Actions<br>Spark SQL<br>Resilient Distributed Datasets (RDDs) | 1. Understand Spark Architecture for data processing (C2).<br>2. Design applications using DataFrames and RDDs (C4). |
| **Unit 5: Stream Processing using Spark** | |
| Advantages and challenges of stream processing<br>Stream Processing Design Points<br>Streaming APIs<br>Structured Stream Processing | 1. Understand different stream processing techniques (C2).<br>2. Design applications for handling real time data using Structured Streaming (C4). |
| **Unit 6: Machine Learning using Spark** | |
| High level M-Lib concepts<br>M-Lib in Action | 1. Understand different libraries and packages for machine learning in Spark (C2).<br>2. Design machine learning model using Spark (C4). |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|

| | | |
|---|---|---|
| Lecture | 30 | 60 |
| Seminar | - | - |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 3 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | * | * | |
| Assignment/Presentation | | | | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. Big Data: Principles and best practices of scalable real-time data systems - Nathan Marz and James Warren. Manning Publisher.<br>2. Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale – Tom White, O'Reilly Publication 4th Edition. |

| | |
|---|---|
| | 3. Spark: The Definitive Guide: Big Data Processing Made Simple – Bill Chambers, Matei Zaharia, O'Reilly Publication 1st Edition.<br>4. http://static.druid.io/docs/druid.pdf, http://druid.io/docs/0.8.0/design/design.html<br>5. Big data architecture and patterns - IBM developerWorks. http://www.ibm.com/developerworks/library/bd-archpatterns1/<br>6. Big Data and Analytics -IBM developerWorks. http://www.ibm.com/developerworks/analytics/<br>7. http://lambda-architecture.net/<br>8. Apache HBase - http://hbase.apache.org/<br>9. Apache Spark Streaming - https://spark.apache.org/streaming/<br>10. Summingbird MapReduce library - https://github.com/twitter/summingbird |

MANIPAL ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Database Programming in Java |
| **Course Code:** CSE-604 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 3 | **Prerequisites:** Basic programming knowledge |

**Synopsis:**

1. To provide fundamental knowledge of various object oriented programming concepts and database concepts.
2. To design and develop database applications using java programming language.

| | |
|---|---|
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| CO 1: | Explain major principles of object oriented programming concepts |
| CO 2: | Discuss the different elements of java programming language |
| CO 3: | Design databases using the conceptual model |
| CO 4: | Develop a java application for various database requirements |

**Mapping of COs to POs**

| *Cos* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* | *PO 12* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | | | | | | | | | |
| CO 2 | * | * | * | | * | | | | | | | |
| CO 3 | | * | | * | | | | | | | | |
| CO 4 | | | * | | * | | | | | | | |

**Course content and outcomes:**

| *Content* | *Competencies* | *No of Hours* |
|---|---|---|
| **Unit 1: Introduction** | | |
| Object Orientation (OO) Concepts | • Illustrate with an example the major principles such as classes, objects, encapsulation, inheritance, polymorphism (C2)<br>• Distinguish between procedure oriented programming and object oriented programming (C2) | |

| | |
|---|---|
| **Unit 2: Introduction to Java** | |
| Data types, Operators, Control Statements. | • Discuss features of java programming language (C2)<br>• Discuss the term platform independence specific to java programming language (C2)<br>• Explain various data types, operators and control statements (C2) |
| **Unit 3: Classes in java** | |
| Class fundamentals, Constructs, Garbage collection, Inner Classes | • Define class structure in java programming language (C1)<br>• Discuss various components of class structure which includes concepts constructors, variables, methods using java (C2)<br>• Explain the mechanism garbage collection (C2)<br>• Illustrate the use of inner classes (C2) |
| **Unit 4: Inheritance** | |
| Introduction to Java Inheritance, Multilevel inheritance, Abstract, final classes | • Define different types of inheritance (C1)<br>• Explain abstract classes (C2)<br>• Discuss final classes (C2)<br>• Apply abstract classes and final classes in applications (C3) |
| **Unit 5: Packages, Interfaces** | |
| Package, access control, Interfaces. | • Illustrate the use of packages in an application (C2)<br>• List various access control mechanism (C1)<br>• Define java interfaces (C1)<br>• Apply interfaces in applications. (C3) |
| **Unit 6: I/O API's** | |

| | |
|---|---|
| Reader, Writer APIs, File Management | • List the types of steam classes available (C1)<br>• Write java program to read data from different types of files (C3)<br>• Discuss file management in java (C2) |
| **Unit 7: Exception Handling** | |
| Using exception handling, Creating user defined exceptions. | • Discuss the types of exception handle (C2)<br>• Explain user define exception class (C2) |
| **Unit 8: Java Applets, Applications** | |
| Java Applets, life cycle, methods, java Application | • Define java applets (C1)<br>• Discuss life cycle of java applets (C2)<br>• Distinguish between java applets and java applications (C2) |
| **Unit 9: Introduction to Swing** | |
| Swing components, Event handling, layout managers | • Distinguish between AWT components and swing components (C2)<br>• Define features of swing components (C1)<br>• Apply different swing components, layout managers in java applications (C3)<br>• Discuss event delegation model (C2) |
| **Unit 10: Introduction to Database concepts** | |
| Primary goal of RDBMS, Purpose of Database System, Characteristics of the Database Approach, Actors on the Scene, Workers behind the scene, Advantages of Using a DBMS, Views of Data | • Define Relational database management (C1)<br>• Discuss the purpose of database system (C2)<br>• Explain characteristics of the database approach (C2)<br>• List actors on the scene and workers behind the scene (C1)<br>• Discuss advantages of using DBMS (C2) |

| **Unit 10: SQL** | |
|---|---|
| Basic Structure, Set Operations, Aggregate Functions, Null Values, Nested Subqueries, Derived Relations, Views, Modification of the Database, Joined Relations, Data-Definition Language | • Explain basic structure of SQL statement (C2)<br>• Discuss set operations (C2)<br>• Explain different types of aggregate functions (C2)<br>• Explain Views, nested queries, joined relations (C2)<br>• Discuss data definition language (C2) |
| **Unit 10: Introduction to JDBC** | |
| JDBC Architecture, Connecting to an ODBC Data Source, JDBC Connection, JDBC Implementation, Resultset Processing, Prepared statement, Other JDBC Classes, Moving the cursor in scrollable Result Sets, Making updates to Updatable Result Sets. | • Explain JDBC architecture (C2)<br>• Explain JDBC connection and its implementation (C2)<br>• Explain different types of jdbc classes which are required for database applications (C2) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Seminar | - | - |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |

| | | |
|---|---|---|
| **TOTAL** | **44** | **74** |
| | | |

| | |
|---|---|
| **Assessment Methods:** | |
| **Formative:** | **Summative:** |
| Theory Assignment | Sessional Examination |
| Lab Assignment | University End Semester Examination |
| Lab Test | Viva |
| Viva | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | |
| Assignment/Presentation | | * | * | * |
| End Semester Examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. Patrick Naughton and Herbert Schildt – "JAVA 2 – The Complete Reference", Tata McGraw Hill.<br>2. George Reese - "Database Programming with JDBC and Java", O'Reilly<br>3. "Database system Concepts",Author: Abraham Silberschatz (Bell Laboratories), Henry F. Korth(Bell Laboratories) and S. Sudarshan (Indian Institute of Technology, Bombay, Publishers: The McGraw-Hill Companies, Inc.<br>4. "Fundamentals of Database systems". Author: Elmasri and Navath |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Data Structures and Algorithms Lab |
| **Course Code:** CSE 601L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits: 1** | **Prerequisites:** C Programming |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>1. This course introduces students to elementary data structures and design of algorithms.<br>2. Students learn how to design optimal algorithms with respect to time and space<br>3. Students learn how to implement link list, stack, queues, searching and sorting techniques, sets, trees and graphs.<br>4. Students learn the design of divide and conquer technique, dynamic programming, greedy technique and back tracking |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Specify and analyse algorithms |
| **CO 2:** | Learn and design programs for implementation of linear and non linear data structure. |
| **CO 3:** | Learn and design programs for sorting and searching. |
| **CO 4:** | Illustrate application of divide and conquer technique, dynamic programming, greedy technique and back tracking. |
| **CO 5:** | Learn to organise the code for scalability and maintainability. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | | * | | | | | | | | | |
| CO 2 | | * | * | | * | | | * | | | |
| CO 3 | | * | * | | * | | | * | | | |
| CO 4 | | * | * | | * | | | * | | | |
| CO 5 | | * | * | | * | | | * | | | |

| **Course content and outcomes:** | |
|---|---|
| ***Content*** | ***Competencies*** |
| **Unit 1: Elementary data structures** | |
| Implementation of Lists, Stacks, Queues | 1. Illustrate and Implement singly linked list. (C3)<br>2. Illustrate and Implement doubly linked list. (C3)<br>3. Illustrate and Implement array-based stack. (C3)<br>4. Illustrate and Implement pointer-based stack. (C3)<br>5. Illustrate and Implement array-based queues. (C3)<br>6. Illustrate and Implement pointer-based queues.(C3) |
| **Unit 2: Sorting & Searching Techniques** | |
| Quick sort, Heap sort, Merge sort, Binary search, linear search, Fibonacci search | 1. Illustrate and implement programs for insertion sort, bubble sort and selection sort. (C3)<br>2. Illustrate and implement programs for quick sort. (C3)<br>3. Illustrate and implement programs for heap sort. (C3)<br>4. Illustrate and implement programs for merge sort. (C3)<br>5. Illustrate and implement programs for binary, linear and Fibonacci search. (C3) |
| **Unit 3: Trees** | |
| Basic Terminology, Implementation of Trees, Binary Trees, Binary Search Trees | 1. Experiment the working of binary trees. (C4)<br>2. Experiment the working of binary search trees. (C4) |

| | |
|---|---|
| | 3. Experiment the working of Tree traversal technique. (C4) |
| **Unit 4: Graphs** | |
| Basic definitions, Representation of Graphs, Minimum Cost Spanning Tree, Single Source Shortest Paths, All-Pairs Shortest Path | 1. Illustrate with a graph using adjacency matrix and adjacency list techniques. (C3)<br>2. Illustrate the implement minimum cost spanning tree. (C3)<br>3. Illustrate the Single source shortest path problem. (C3)<br>4. Illustrate the All- pair shortest path problem. (C3) |
| **Unit 5: Algorithm Design Techniques** | |
| Divide-and-Conquer Algorithms, Dynamic Programming, Greedy Algorithms, Backtracking | 1. Illustrate max min problem. (C3)<br>2. Illustrate Strassen's matrix multiplication problem. (C3)<br>3. Illustrate matrix chain order problem. (C3)<br>4. Illustrate knap-sack, job scheduling with dead line and optima storage on taps problems. (C3)<br>5. Illustrate n queens and graph colouring problems. (C3) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |

| | | |
|---|---|---|
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |
| | | |

| **Assessment Methods:** | |
|---|---|
| **Formative:** | **Summative:** |
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | | | |
| Sessional Examination 2 | | * | * | * | |
| Assignment/Presentation | * | * | * | * | * |
| Laboratory Examination | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. "Introduction to Algorithms" Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest.<br>2. "Data Structures& Algorithms" Aho, Hopcroft and Ulmann<br>3. "Data structures and algorithm analysis in C" Mark Allen Weiss<br>4. "Computer Algorithms" : Ellis Horowitz, SartajSahni, SanguthevarRajasekaran |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Blockchain Application Development LAB |
| **Course Code:** BCH 601L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits: 3** | **Prerequisites:** |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on depth practical knowledge in Blockchain technology. The basic understanding of programming required for building blockchain application using Node.js. This course allows students to have a fundamental knowledge on HTML, CSS and JavaScript for developing blockchain or any web applications. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | To build and develop the web based blockchain applications. |
| **CO 2:** | To identify the programming languages used for blockchain development. |
| **CO 3:** | To illustrate and demonstrate problems required for blockchain development. |
| **CO 4:** | To write application in general or specific to blockchain. |
| **CO5:** | To Model and Construct the blockchain application. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | * | * | | | | | | |
| CO 2 | * | | * | | * | | | | | | |
| CO 3 | * | * | * | | * | | | | | | |
| CO 4 | * | | | * | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| Introduction to applications – Types of Applications – Application vs | 1. To demonstrate the Installation of required platform and its study. (C1) |

| | |
|---|---|
| Implementation – Functional vs Non-functional – Programming languages in Blockchain – Blockchain Platforms<br>Introduction to Web Applications – Web Servers – HTML vs HTML 5 – HTML FORMS – CSS – CSS3 | |
| **Unit 2:** | |
| JavaScript – JQUERY – AJAX | 1. To demonstrate/illustrate constructs of JAVASCRIPT (C2)<br>2. To demonstrate applications with JavaScript , jQuery and AJAX (C3) |
| **Unit 3:** | |
| Node.js – Constructs of Node.js – Classes – Objects - Creating blockchain with node.js | 1. To illustrate the constructs in Node.js (C2)<br>2. To illustrate the concepts of OOPS in node.js (C2)<br>3. To illustrate applications with node.js (C3) |
| **Unit 4:** | |
| Introduction to Databases – Database Tools – Databases in Blockchain - ER diagrams - Normalization – SQL – NOSQL | 1. To illustrate the concepts involved in databases (C2)<br>2. To illustrate the concepts of ER diagrams (C2)<br>3. To illustrate the working of SQL and NO SQL tools (C3) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |

| | | |
|---|---|---|
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | * |
| Assignment/Presentation | * | * | * | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Learn Blockchain Programming with JavaScript: Build your very own Blockchain and decentralized network with JavaScript and Node.js**, Eric Traub, Packt Publishing, 2018.<br>2. **HTML 5 Black Book (Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP, jQuery)**, DT Editorial Services, Dreamtech Press, Second edition,2016. |

| | |
|---|---|
| | 3. **HTML & CSS: The Complete Reference**, Thomas Powell, McGraw Hill Education, Fifth Edition, 2017.<br>4. **Beginning Node.js**, Basarat Ali Syed, Apress, 2014.<br>5. **Node.Js Web Development**, David Herron, Ingram short title; 3rd Revised edition, 2016.<br>6. **Mastering Node.js,** Sandro Pasquali, Kevin Faaborg, Packt Publishing Limited; 2nd Revised edition, 2017<br>7. **Full-Stack JavaScript Development: Develop, Test and Deploy with MongoDB, Express, Angular and Node on AWS**, Eric Bush, Red Sky, 2016.<br>8. **Blockchain Applications: A Hands-On Approach**, Arshdeep Bahga, Vijay Madisetti, VPT, 1 edition, 2018. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Blockchain Technology Lab |
| **Course Code:** BCH 603L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 3 | **Prerequisites:** Basic Network Concepts |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on understanding the working of blockchain technology and how blockchain platform works. The course discuss on the nuances involved in blockchain technology and its implementation on the blockchain platform. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Develop the blockchain ecosystem using Ethereum. |
| **CO 2:** | Evaluate the application based on Ethereum. |
| **CO 3:** | Examine the development process using Hyperledger. |
| **CO 4:** | Demonstrate the blockchain application development process. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | * | | * | | | | | | |
| CO 2 | * | * | * | | * | | | | | | |
| CO 3 | * | * | | | * | | | | | | |
| CO 4 | * | * | | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| Introduction to Blockchain - Potential of Blockchain – Defining Blockchain – Ownership – Understanding Ledger – Ledger Structure – Concepts of Ownership – Centralized vs | At the end of the topic student should be able to:<br>1. Identify the required programming environment and operating systems. (C1) |

| | |
|---|---|
| Decentralized - Components of a Blockchain -Characteristics of Blockchain - The growth of blockchain technology - Distributed systems - The history of blockchain and Bitcoin - Types of blockchain – Consensus - CAP theorem and blockchain - Decentralization using blockchain Methods of decentralization- Routes to decentralization - Blockchain and full ecosystem decentralization- Smart contracts- Decentralized Organizations. | 2. Relate examples and constructs of the chosen programming languages (python/solidity). (C1) |
| **Unit 2:** | |
| Ethereum and working with Smart Contracts : Understand Ethereum ,Define Smart Contracts,Identify Cryptocurrency used in Ethereum,Describe Transactions in Ethereum,Define Consensus Mechanism in Ethereum,List Development Technologies,Identify Ethereum Clients, Define Platform Functions,Understand Solidity, Describe Solidity Operators and Functions, Setting up Metamask,How to interface with ethereum network,First smart contract,Ethereum accounts and how to receive ether, Structuring a contract,Declaring a | 1. To describe the architecture of Ethereum. (C1)<br>2. To illustrate and build blockchain examples with Ethereum platform. (C3) |

| | |
|---|---|
| function, Deploying and redeploying of a contract,Comparing Wei & Ether,What is a gas transaction,Remix testing. | |
| **Unit 3:** | |
| Hyperledger : Define Hyperledger Blockchain , Understand Hyperledger Consensus Algorithm ,Explain Hyperledger Iroha ,Identify Hyperledger Components ,Describe Setting up Channels Policies ,Chaincodes List Hyperledger , Explorer Components ,Define Hyperledger Composer,Fabric Under the Hood (Concepts & Terminology),Ledger Implementation,Dev Environment Walkthrough: Peer & CouchDB setup,Ledger Implementation,Peers Nodes : Anchors and Endorsers, Anchor Peers & Endorsing Peers,Clients Node: Endorsement Policies, Client Peer & Endorsing Policies Orderer Nodes,Membership Service Provider & Certification Authority,Dev Environment Walkthrough: Orderer and CA Server,Chaincode Development. | 1. To identify different components in Hyperledger (C2)<br>2. To illustrate the examples of Hyperledger (C3) |
| **Unit 4:** | |

| | |
|---|---|
| Creating private Blockchain with Multichain : Define Multichain , Describe MultiChain Streams , Create & deploy private blockchain ,Explain Connecting to a Blockchain ,Identify Multichain Interactive Mode ,List Native assets ,Define Transaction Metadata ,Explain Streams Explain Mining ,Bitcoin to private blockchain,Aim of multichain,Hand-shake process,Multi-chain use cases,Multichain permission,Multichain assets,multichain streams,Basics of retrieving from streams,Consensus model,Multichain flexibility,Deployment options,Speed and scalability of multichain | 1. To define and describe the multichain blockchain (C2)<br>2. To explain the mining in multichain process (C3)<br>3. To illustrate the deployment of multichain blockchain and its applications (C2) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |

| | | |
|---|---|---|
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |
| | | |

| **Assessment Methods:** | |
|---|---|
| **Formative:** | **Summative:** |
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | |
| Sessional Examination 2 | | | * | * |
| Assignment/Presentation | * | * | * | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Blockchain Basics: A Non-Technical Introduction in 25 Steps**, Daniel Drescher, Apress; 1st Edition, 2017.<br>2. **Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions**, Bikramaditya Singhal, Gautam Dhameja , Priyansu Sekhar Panda, Apress; 1st ed. Edition, 2018.<br>3. **Mastering Blockchain**, Imran Bashir, Ingram short title, Second Edition, 2018.<br>4. **Hands-On Blockchain with Hyperledger**, Petr Novotny Venkatraman Ramakrishna Nitin Gaur Anthony O'Dowd Luc Desrosiers, Ingram short title, 2018.<br>5. **Solidity Programming Essentials**, Ritesh Modi, Ingram short title, 2018 |

6. **BlockChain from Concept to Execution**, Debajani Mohanty, BPB; 2nd revised and updated edition, 2018.
7. **Mastering Blockchain Programming with Solidity: Write production-ready smart contracts for Ethereum blockchain with Solidity**, Jitendra Chittoda, Packt Publishing Limited, 2019.
8. **Hands-On Blockchain with Hyperledger**, Petr Novotny Venkatraman Ramakrishna Nitin Gaur Anthony O'Dowd Luc Desrosiers, Ingram short title, 2018
9. **Blockchain for Dummies**, Tiana Laurence, 2nd edition – 2019.
10. **Hands-On Smart Contract Development with Solidity and Ethereum: From Fundamentals to Deployment**, David Hoover, Kevin Solorio, Randall Kanna, Shroff/O'Reilly; First edition, 2019.
11. **Blockchain By Example: A developer's guide to creating decentralized applications using Bitcoin, Ethereum, and Hyperledger**, Bellaj Badr , Richard Horrocks, Xun (Brian) Wu , Packt Publishing Limited, 2018.
12. **Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners**, Chris Dannen, APRESS, 1 edition, 2017.
13. **Ethereum: Blockchains, Digital Assets, Smart Contracts, Decentralized Autonomous Organizations**, Henning Diedrich, CreateSpace Independent Publishing Platform; 1st edition, 2016.

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Cryptocurrency and Smart Contracts Lab |
| **Course Code:** BCH 605L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 3 | **Prerequisites:** Cryptography Basics, Networking Basics, Programming aspects |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>Create a NodeJS application with real-time WebSocket connections.<br>Build an API with NodeJS and Express<br>Design, test, and deploy secure Smart Contracts |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | To understand how block chain system works |
| **CO 2:** | To design and deploy smart contracts |
| **CO 3:** | Integrate ideas from block chain technology into their own projects |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | | | | | | | | | |
| CO 2 | | * | * | * | * | | | | | | |
| CO 3 | * | * | * | | * | | | | | | |
| CO 4 | | | * | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1: Introduction** | |
| Cryptography and Hashing | At the end of the topic student should be able to:<br>Use open source tools to generate hashes for content. (C3) |
| **Unit 2:** | |
| Block Chain | Identify various popular blockchain applications. Create a list of those applications. (C3) |

| **Unit 3:** | |
|---|---|
| Bitcoin | Build a transaction and then hash it. Generate public and private keys. Digitally sign a transaction. (C3)<br>Explore the bitcoin on blockchain.info for block generation. (C6) |
| **Unit 4:** | |
| Smart Contracts | Create Smart Contracts through open source tools |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | |
| Assignment/Presentation | | * | * | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | | * | | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Cryptography and Network Security - Principles and Practice, Stallings William**, Pearson Education; Seventh edition, 2017<br>2. **Cryptography And Network Security**, Forouzan, McGraw Hill Education, 2015<br>3. **Cryptography and Network Security**, Atul Kahate, McGraw Hill Education; Third edition, 2017.<br>4. **The Age of Cryptocurrency: How Bitcoin and Digital Money Are Challenging the Global Economic** , Paul Vigna , Michael J. Casey, St. Martin's Press, 2015.<br>5. **Mastering Ethereum: Building Smart Contracts and DApps, Andreas M. Antonopoulos**, Dr. Gavin Wood, , Shroff/O'Reilly; First edition, 2018<br>6. **Mastering Bitcoin: Programming the Open Blockchain**, Andreas M. Antonopoulos, Shroff/O'Reilly; Second edition, 2017.<br>7. **Programming Bitcoin: Learn How to Program Bitcoin from Scratch**, Jimmy Song, O'Reilly Media, 1 edition 2019.<br>8. **Blockchain for Business with Hyper-ledger Fabric**, Nakul Shah, BPB Publications; 1st edition, 2019.<br>9. **Hands-On Cybersecurity with Blockchain**, Rajneesh Gupta , Packt Publishing; 1st edition, 2018.<br>10. **Ethereum Smart Contract Development: Build blockchain-based decentralized applications using solidity**, Mayukh Mukhopadhyay, Packt Publishing Limited, 2018. |

| | |
|---|---|
| | 11. **Building Blockchain Projects**, Narayan Prusty, Packt Publishing Limited, 2017.<br>12. **Hands-On Bitcoin Programming with Python: Build powerful online payment centric applications with Python**, Harish Garg, Packt Publishing Limited, 2018.<br>13. **Bitcoin and Cryptocurrency Technologies – A Comprehensive Introduction Hardcover**, Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller , Steven Goldfeder,Princeton University Press, 2016.<br>14. **Fundamentals of Smart Contract Security**, Richard Ma, Jan Gorzny, Edward Zulkoski, Momentum Press, 2019. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Distributed Computing and Databases Lab |
| **Course Code:** BCH-615L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 3 | **Prerequisites:** Introductory Course in IoT, Networking Basics, Programming aspects, Operating system, Linux |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | To identify and introduce concepts related to distributed computing systems. |
| **CO 2:** | Identify the introductory distributed database concepts and its structures. |
| **CO 3:** | Describe terms related to distributed object database design and management. |
| **CO 4:** | Produce the transaction management and query processing techniques in DDBMS. |
| **CO5** | Relate the importance and application of emerging database technology. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | * | * | | | | | | |
| CO 2 | * | * | * | | * | | | | | | |
| CO 3 | * | * | | | * | | | | | | |
| CO 4 | * | * | * | | * | | | | | | |
| CO5 | * | * | * | * | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| Introduction – Characterization of Distributed Systems – Examples - Resource Sharing and the Web-Challenges – Distributed System Models – Architectural, Fundamental - | At the end of the topic student should be able to:<br>1. To identify the programming environment for Distributed computing (C1) |

| | |
|---|---|
| Interprocess Communication - API for Internet protocols – Message-Oriented Communication – Unicasting - Multicasting and Broadcasting - Client-Server communication – Group Communication. | 2. To illustrate small examples with the understanding of constructs required for Interposes communications(C3) |
| **Unit 2:** | |
| Distributed Operating Systems: Issues in Distributed Operating System – Remote Invocation - Communication between distributed objects - Remote procedure calls -Threads in Distributed Systems – Clock Synchronization – Election Algorithms – Distributed Mutual Exclusion – Distributed Transactions – Distributed Deadlock – Agreement Protocols. | 1. To illustrate and solve problems through the concept of Remote procedure calls and threads (C3)<br>2. To illustrate problems with algorithms related to distributed computing. (C3) |
| **Unit 3:** | |
| Distributed File Systems: Introduction - File service architecture - File System-Enhancements and further developments - Name Services: Introduction - Name Services and the Domain Name System - Directory Services - Time and Global States - Clocks, events and process states Synchronizing physical clocks - Logical time and logical clocks - Distributed debugging - Coordination and Agreement-Introduction - Distributed mutual exclusion – Elections - Multicast | 1. To illustrate examples for handling files, clock and time in distributed systems.(C3) |

| | |
|---|---|
| communication-Consensus and related problems | |
| **Unit 4:** | |
| Distributed Shared Memory: Introduction - Design and implementation issues - Sequential consistency and Ivy case study - Release consistency and Munin case study - Other consistency models - Introduction to Fault Tolerance - Distributed Commit Protocols – Byzantine Fault Tolerance – Impossibilities in Fault Tolerance. | 1. To illustrate and build programs for shared memory concepts.(C3) |
| **Unit 5:** | |
| Introduction to Databases –Structured vs Unstructured vs SemiStructured data – RDBMS – Entity Relationships – SQL – NOSQL – Distributed Databases – Graph Databases – Tools | 1. To identify the databases for SQL and NOSQL(C1)<br>2. To illustrate application based on SQL and NOSQL.(C3) |
| **Unit 6:** | |
| Security In Databases: Security requirements of database systems – Reliability and Integrity in databases – Redundancy – Recovery – Concurrency/ Consistency – Monitors – Sensitive Data – Types of disclosures – Inference-Finding and Confirming SQL injection | 1. To illustrate the security issues in databases. (C3)<br>2. To illustrate the steps involved in sql injection and discuss the pros and cons. (C3) |
| **Learning strategies, contact hours and student learning time** | |

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | - |
| | | |

| **Assessment Methods:** | |
|---|---|
| **Formative:** | **Summative:** |
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | |
| Assignment/Presentation | | * | * | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **SQL injection Attacks and defense**, Justin Clarke, Syngress, 2nd edition,2012 |

| | |
|---|---|
| | 2. **Distributed Systems Concepts and Design**, George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair, Pearson, 5th edition, 2011.<br>3. **Distributed Systems Principles and Paradigms**, A.S.Tanenbaum, M.Van Steen, PHI, 2012.<br>4. **Distributed Computing: Fundamentals, Simulations and Advanced Topics**, Hagit Attiya and Jennifer Welch, Wiley, 2004.<br>5. **Distributed Computing Principles and Applications**, M.L.Liu, Pearson Addison Wesley, 2004.<br>6. **Graph Databases: New Opportunities for Connected Data**, Ian Robinson, Jim Webber, Emil Eifrem, Shroff/O'Reilly, 2nd Edition, 2016.<br>7. **Graph Algorithms: Practical Examples in Apache Spark & Neo4j**, Mark Needham, Amy E. Hodler, Shroff/O'Reilly; First edition, May 2019. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Architecture of Big Data Systems Lab |
| **Course Code:** BDA 623L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First year, First semester |
| **No of Credits:** 1 | **Prerequisites**: Programming in Python or Java |

| | |
|---|---|
| **Synopsis:** | 1. This course helps the student to understand the concepts of Hadoop framework, Spark framework and their internals.<br>2. This course helps the students to learn Map-reduce programming, Spark programming.<br>3. This course helps the students to build machine learning model using Spark framework. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Install and develop applications using Hadoop and its ecosystems |
| **CO 2:** | Build applications using Spark frame work |
| **CO 3:** | Build Machine Learning models using Spark |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | * | | * | * | | | * | * | |
| CO 2 | * | * | * | | * | * | | | * | * | |
| CO 3 | * | * | * | | * | * | | | * | * | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1: Hadoop ecosystem** | |
| Installation and configuring Hadoop ecosystem | 1. Configure HDFS and YARN (C2)<br>2. Data handling using Sqoop, Hive, PIG(C2)<br>3. Implementing MapReduce applications (C2) |

| **Unit 2: Spark Framework** | |
|---|---|
| Spark tool chain – RDD, DataFrame, SQL and Streaming | 1. Develop applications using Spark DataFrame and SQL (C4).<br>2. Design real time applications using Spark Streaming (C4). |
| **Unit 3: Machine Learning using Spark** | |
| MLIB | 1. Build machine learning models using Spark(C5) |

| **Learning strategies, contact hours and student learning time** | | |
|---|---|---|
| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | - |
| | | |

| **Assessment Methods:** | |
|---|---|
| **Formative:** | **Summative:** |
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |

| | |
|---|---|
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 |
|---|---|---|---|
| Sessional Examination 1 | * | | |
| Sessional Examination 2 | | * | * |
| Assignment/Presentation | * | * | * |
| End Semester Examination | * | * | * |
| Laboratory Examination | * | * | * |

| | |
|---|---|
| **Feedback Process** | • End-Semester Feedback |
| **Reference Material** | 1. Big Data: Principles and best practices of scalable real-time data systems - Nathan Marz and James Warren. Manning Publisher.<br>2. Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale – Tom White, O'Reilly Publication 4th Edition.<br>3. Spark: The Definitive Guide: Big Data Processing Made Simple – Bill Chambers, Matei Zaharia, O'Reilly Publication 1st Edition. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Database Programming in Java Lab |
| **Course Code:** CSE-604L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 3 | **Prerequisites:** Basic Programming knowledge |

| | |
|---|---|
| **Synopsis:** | 1. To provide fundamental knowledge of various object oriented programming concepts and database concepts.<br>2. To design and develop database applications using java programming language. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| CO 1: | Apply object oriented programming concepts in a java application |
| CO 2: | Practice various types of UI based applications |
| CO 3: | Manipulate database using various SQL Commands |
| CO 4: | Write java applications for various database requirements |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* | *PO 12* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | | * | * | | * | | | | | | | |
| CO 2 | | * | * | | * | | | | | | | |
| CO 3 | | * | * | | * | | | | | | | |
| CO 4 | | * | * | | * | | | | | | | |

**Course content and outcomes:**

| *Content* | *Competencies* |
|---|---|
| **Unit 1: Installation of JDK tools** | |
| Installation of JDK tools, setting environment variables for java application, writing simple java program, practice to compile and run java application | • Use of JDK tools for java application (C3)<br>• Solve the issues related to java setting environment variables (C3)<br>• Analyse simple java application (C4) |
| **Unit 2: Introduction to OOP's concepts** | |
| Implementation of OOP's concepts in java application such as encapsulation, various types of Inheritance, polymorphism. Apart | • Apply OOP's concepts in java application (C3)<br>• Solve the issues such as multiple inheritance, exception handling(C3) |

| | |
|---|---|
| from this other techniques such as exception handling, packages, interfaces, IO streams. | • Write java programs to understand more about file read and write (C3) |
| **Unit 3: Introduction to Window based applications** | |
| Implementation of window based applications using swing components such as forms, menu based applications. Applying event handling mechanism to the applications | • Write UI applications for different look and feel (C3)<br>• Use of swing components and layout managers for UI design (C3)<br>• Test UI applications (C3) |
| **Unit 4: Database applications using JDBC driver** | |
| Installation of JDBC driver, use of it in database applications, creating database, manipulating data through window based applications | • Test various Structured Query Language (SQL) commands (C4)<br>• Write database applications using JDBC driver and mysql database (C3) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Theory Assignment | Sessional Examination |
| Lab Assignment | University End Semester Examination |

| | | | | |
|---|---|---|---|---|
| Lab Test | | | Viva | |
| Viva | | | | |
| **Mapping of assessment with Cos** | | | | |
| Nature of assessment | CO 1 | CO 2 | CO 3 | CO4 |
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | * |
| Assignment/Presentation | * | * | * | * |
| End Semester Examination | | | | |
| Laboratory examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | • Patrick Naughton and Herbert Schildt – "JAVA 2 – The Complete Reference", Tata McGraw Hill.<br>• George Reese - "Database Programming with JDBC and Java", O'Reilly<br>• "Database system Concepts",Author: Abraham Silberschatz (Bell Laboratories), Henry F. Korth(Bell Laboratories) and S. Sudarshan (Indian Institute of Technology, Bombay, Publishers: The McGraw-Hill Companies, Inc.<br>• "Fundamentals of Database systems". Author: Elmasri and Navath |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Mini Project - 1 |
| **Course Code**: BCH 695 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 4 | **Prerequisites:** Any programming language and circuit basics |

| | |
|---|---|
| **Synopsis:** | Students are expected to select a problem in the area of their interest and the area of their specialization that would require an implementation in hardware / software or both in a semester |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Apply the objectives of the project work and provide an adequate background with a detailed literature survey |
| **CO 2:** | Breakdown the project into sub blocks with sufficient details to allow the work to be reproduced by an independent researcher |
| **CO 3:** | Compose hardware/software design, algorithms, flowchart, methodology, and block diagram |
| **CO 4:** | Evaluate the results |
| **CO 5:** | Summarize the work carried out |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | | | | * | | | | | | | |
| CO 2 | | | | | * | | | * | | | |
| CO 3 | | | | | | | * | | | * | |
| CO 4 | | | | | | * | | | | | * |
| CO5: | | | | | | | * | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Phase 1** | |

| | |
|---|---|
| Problem identification, synopsis submission, status submission, mid evaluation. | At the end of the topic student should be able to:<br>1. Identify the problem/specification (C1)<br>2. Discuss the project (C2)<br>3. Prepare the outline (C3)<br>4. Describe the status of the project (C2)<br>5. Prepare a mid-term project presentation report (C3)<br>6. Prepare and present mid-term project presentation slides (C3, C5)<br>7. Develop project implementation in hardware/software or both in chosen platform (C5) |
| **Phase 2** | |
| Status submission, final evaluation. | 1. Prepare the progress report (C3)<br>2. Prepare the final project presentation report (C3)<br>3. Prepare and present final project presentation slides (C3, C5)<br>4. Modify and Develop implementation in hardware/software or both in chosen platform (C3, C5)<br>5. Justify the methods used and obtained results (C6) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | - | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | 48 | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |

| | | |
|---|---|---|
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Practical | - | - |
| Revision | - | - |
| Assessment | 03 | - |
| **TOTAL** | **51** | **09** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Project Problem Selection | Mid-Term Presentation |
| Synopsys review | Second status review |
| First status review | Demo & Final Presentation |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|
| Mid Presentation | * | * | | | |
| Presentation | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | • End-Semester Feedback |
| **Reference Material** | Particular to the chosen project |

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Seminar - 1 |
| **Course Code**: BCH 697 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 1 |
| **No of Credits:** 1 | **Prerequisites:** Communication Skill |

| | |
|---|---|
| **Synopsis:** | 1. To select, search and learn technical literature.<br>2. To Identify a current and relevant research topic.<br>3. To prepare a topic and deliver a presentation.<br>4. To develop the skill to write a technical report.<br>5. Develop ability to work in groups to review and modify technical content. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Show competence in identifying relevant information, defining and explaining topics under discussion. |
| **CO 2:** | Show competence in working with a methodology, structuring their oral work, and synthesizing information. |
| **CO 3:** | Use appropriate registers and vocabulary, and will demonstrate command of voice modulation, voice projection, and pacing. |
| **CO 4:** | Demonstrate that they have paid close attention to what others say and can respond constructively. |
| **CO 5:** | Develop persuasive speech, present information in a compelling, well-structured, and logical sequence, respond respectfully to opposing ideas, show depth of knowledge of complex subjects, and develop their ability to synthesize, evaluate and reflect on information. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | | | | | | * | * | | * |
| CO 2 | * | | | | | | | * | * | | * |
| CO 3 | * | | | | | | | * | * | | * |
| CO 4 | * | | | | | | | * | * | | * |
| CO5: | * | | | | | | | * | * | | * |

| **Learning strategies, contact hours and student learning time** | | |
|---|---|---|
| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
| Lecture | - | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | 14 | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Practical | - | - |
| Revision | - | - |
| Assessment | - | - |
| **TOTAL** | **14** | - |
| | | |

| **Assessment Methods:** | |
|---|---|
| **Formative:** | **Summative:** |
| Seminar Topic Selection | |
| Synopsys review | |
| PPT Review | |
| | |

| **Mapping of assessment with Cos** | | | | | |
|---|---|---|---|---|---|
| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
| Presentation | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | • End-Semester Feedback |
| **Reference Material** | Particular to the chosen Seminar |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | DevOps for Cloud |
| **Course Code**: CDC-607 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 3 | **Prerequisites:** |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on:<br>1. Devops Product Life Cycles Stage.<br>2. Automation of product lifecycle. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Explain the concept of automation of Product Life Cycle stages. |
| **CO 2:** | Demonstrate Continuous Integration / Continuous Testing / Continuous Deployment of Product. |
| **CO 3:** | Compare and contrast existing Software Methodologies with Devops Life Cycle stages. |
| **CO 4:** | Design and Devops methodologies for Product development and Release |
| **CO 5:** | Explain the concepts of Tools used in each stages of Devops . |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | | | | | | | | |
| CO 2 | | | * | | * | | | | | | |
| CO 3 | | * | * | | | | | | | | |
| CO 4 | * | | | | | | | | | | |
| CO 5 | * | | | | | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1: DevOps Introduction** | |
| Understanding Development - Developement SDLC : WaterFall & Agile - Understanding Operations - | 1. Explain about the Product Life Cycle Software methodologies (C2)<br>2. Describe Devops life cycle for Product Development and Release (C2) |

| | |
|---|---|
| Dev vs Ops - DevOps to the rescue - What is DevOps - DevOps SDLC - Continous Delivery model - DevOps tools for DevOps SDLC - DevOps Roles & Responsiblities. | 3. Explain the stages of Devops (C2)<br>4. Describe about Continuous Integration / Continuous Deployment pipeline. (C2)<br>5. Write the significance of automation in Product life cycle management. (C3)<br>6. Describe different between standard software methodologies and Devops software methodologies. (C2) |
| **Unit 2: Linux** | |
| Linux Introduction, Principles & Linux distro – Booting - Command line utililities & Basic commands - Linux Filesystem - Filters & I/O Redirections - Users & Group administration - File permissions & Ownerships - Sudo - Software Managemen - Useful tools: ssh, telnet, scp, rsync, disk utils, backups etc - Service & Process management - Shell Scripting - Systems and HW stats – Linux Containers (lxc) - Dockers – Kubernetes and Microservices . | 1. Explain the evolution of Linux OS (C2)<br>2. Explain Linux File System (C2)<br>3. Demonstrate Linux Users and Groups (C3)<br>4. Describe OS Level Virtualization techniques like Containers (C3)<br>5. Demonstrate basic Linux Commands (C4) |
| **Unit 3: Networking fundamentals** | |
| Components of computer networks - Classification: LAN, WAN, Peer to Peer network, Server based - Switches - Routers - Network Architecture - Protocols - Port numbers - DNS - DHCP - IP Addresses - Ip Addresses & Subnet Masks - IP Address Ranges - Subnetting - Private Vs Public | 1. Explain Computer network and devices (C2)<br>2. Demonstrate subnetting and its need (C3)<br>3. Explain IPV4 Addressing scheme (C2)<br>4. Demonstrate type of Network Devices like Switches , Hub , Router using Simulator Tools (C4)<br>5. Describe networking Services like DNS , DHCP , NACL , FTP etc ( C4) |

| | |
|---|---|
| networks - High Availaiblity - Firewalls & NACL - Web Application Architecture - Infrastructure - Network layout - Services & Components - Architecture from a DevOps perspective. | |
| **Unit 4: Automation, Orchestration & Config Management** | |
| Version control system with Git : What is VCS & why it is needed - DevOps use cases - Setup your own repo with git - Manage your code base/source code with GIT & GITHUB | 1. Explain need and types of version control software (C1)<br>2. Describe architecture of Distributed version control systems (C2)<br>3. Explain Git and Github as case study (C3) |
| **Unit 5: Continuous Integration with Jenkins** | |
| Introduction to continuous integration. - Build & Release and relation with DevOps - Understanding development and developers - Why Continuous integration Jenkins introduction and setup - Jenkins projects/jobs - Jenkins plugins \| Jenkins administration: Users - Nodes/slaves - Managing plugins - Managing software versions - Introduction - Phases - Java builds - Build and Release job/project setup \| Nexus: Intro & Setup - Software versioning & Hosted repository - Integration with Jenkins - Continuous integration job/project setup \| Complete Jenkins project: Packinging Artifacts - Static code Analysis - Tomcat setup Staging & productions - Artifacts | 1. Describe about Continuous Integration / Continuous Deployment pipeline. (C2)<br>2. Write the significance of automation in Product life cycle management. (C3)<br>3. Describe different between standars software methodologies and Devops software methodologies. (C2)<br>4. Give examples for Automation of stages of Product development using Devops . (C2)<br>5. Write the limitation of a Current Software methodologies for Product Development. (C3)<br>6. Describe the architecture of Continuous Integration server. (C2)<br>7. Apply Devops methodologies for Product Development and Release(C3 |

| | |
|---|---|
| deployments to webservers from Jenkins - Build Pipeline - Jenkins not just CI tool anymore - More DevOps use cases of Jenkins | |
| **Unit 5: Ansible** | |
| Configuration Management & Automation - What is Ansible & its features - Ansible setup on local & cloud - Understanding Ansible architecture & Execution - Inventory \| Ad hoc commands: Automating change Management with Ad Hoc commands - Playbook Introduction - Ansible configuration with ansible.cfg - Ansible documentation - Modules, modules & lots of modules - Writing playbook for webserver & DB server deployments - Tasks - Variables - Templates - Loops - Handlers - Conditions - Register - Debugging - Ansile Roles - Identify server roles - Roles structure - Creating, Managing and executing roles - Ansible Galaxy - Exploring Roles from Galaxy - Download Galaxy roles and integrate with your code - Ansible Advanced Execution - Improving execution time - Limiting and selecting tasks - Troubleshooting and Testing. | 1. Write the steps in Automation of Testing in Web development. (C3)<br>2. Explain the operations Continuous Testing. (C5)<br>3. Write the taxonomy of Continuous Integration / Continuous Delivery / Continuous Deployment (C3)<br>4. Design a Workflow for Automation of Product life cycle using Devops (C5, P3).<br>5. Construct a Continuous Integration / Continuous Deployment pipeline (C5)<br>6. Compare Standard Software methodologies vs Devops methodologies for Product Development. (C6, P2)<br>7. Describe about Containers and Container Orchestration Services. (C2)<br>8. Examine the advantages of using Containers in Web development(C4)<br>9. Describe Container orchestration services architecture(C2)<br>10. Show the function of Container orchestration services(C3)<br>11. Define Configuration Management tools and its need. (C1)<br>12. Describe the features of Configuration Management. (C2) |

| | |
|---|---|
| | 13. Explain the architecture of Configuration Management (C5)<br>14. Design a Configuration Management Codes to administrate infrastructure of organization (C5)<br>15. Explain the need of Continuous Monitoring tools (C5)<br>16. Design an Architecture Continuously Monitor infrastructure. (C4) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | | |
| Sessional Examination 2 | | | * | * | * |
| Assignment/Presentation | * | * | * | * | * |
| End Semester Examination | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | [1]. Eric Foster-Johnson , John C. Welch , Micah Anderson, Beginning Shell Scripting (Programmer to Programmer), Wrox Publications<br>[2]. Randal K. Michael "Mastering Unix Shell Scripting: Bash, Bourne, and Korn Shell Scripting for Programmers, System Administrators, and UNIX Gurus", 2nd Edition, Wiley Publications<br>[3]. Bintu Harwani, "UNIX & Shell Programming", Oxford Publications, 2013<br>[4]. John Ferguson Smart, "Jenkins: The Definitive Guide",O'reilly Publications<br>[5]. Mitesh Soni, "Jenkins Essentials", Packt Publications<br>[6]. Rafal Leszko, "Continuous Delivery with Docker and Jenkins", Packt Publications<br>[7]. Veselin Kantsev, "Implementing DevOps on AWS", Packt Publications<br>[8]. Randall Smith, "Docker Orchestration", Packt Publications<br>[9]. Alan Berg, "Jenkins Continuous Integration Cookbook", Packt Publications<br>[10]. Kumaran S., Senthil, " Practical LXC and LXD Linux Containers for Virtualization and Orchestration", Apress Publications<br>[11]. Konstantin Ivanov, " Containerization with LXC" , Packt Publications |

| | |
|---|---|
| | [12]. Karl Matthias, Sean Kane, "Docker: Up & Running:Shipping Reliable Containers in Production",O'Reilly Media |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Network Security and Analysis |
| **Course Code:** BCH 602 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 3 | **Prerequisites:** Microprocessor architecture , Microcontroller Architecture , Assembly language and Number systems |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>The course allows students to understand the network devices and the data formats for different protocols. As lot of data is populated in the present world an analysis of these packets using the modern tools help to understand the attacks and the nature of the network. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Identify some of the factors driving the need for network security |
| **CO 2:** | Identify and classify particular examples of attacks |
| **CO 3:** | Define the terms vulnerability, threat and attack |
| **CO4:** | Identify physical points of vulnerability in simple networks |
| **CO5:** | Compare and contrast symmetric and asymmetric encryption systems and their vulnerability to attack, and explain the characteristics of hybrid systems. |
| **CO6:** | Analyze the packets through the network tools |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | * | | | | | | | |
| CO 2 | * | * | | | | | | | | | |
| CO 3 | * | * | | | | | | | | | |
| CO 4 | * | * | | | | | | | | | |
| CO 5 | | | | * | | | | | | | |
| CO 6 | | | * | * | | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|

| **Unit 1:** | |
|---|---|
| **INTERNET SECURITY & ENCRYPTION** Encryption of static data, IPSec, AH, ESP, IKE, ISAKMP/Oakley, Tunnel mode, Transport mode, Virtual Private Networks (VPNs), SSH Tunneling, IP6 issues, Cloud Security Issues. | At the end of the topic student should be able to:<br>1. List out the attack scenario in cloud. (C1)<br>2. Explain different types of encryption algorithms to protect the static data. (C2) |
| **Unit 2:** | |
| **FIREWALLS:** Packet Filters, Stateful, Stateless, Bastion Host, Circuit Level, Application gateway, SOCKS, DMZ, Host-Based Firewall, Egress Filtering, Network Address Translation (NAT), Multi-homing, IPTables/NetFilter, implementing NAT. | 1. Examine the need of security for the given network scenario. (C2)<br>2. Infer the design of stateful firewall and stateless firewall. (C2)<br>3. Appraise the importance of DMZ. (C4) |
| **Unit 3:** | |
| **SNIFFERS AND PACKET CRAFTING:** Libpcap, dSniff, Wireshark, tcpdump, Mitigation of Sniffer Attacks, ARP Cache Poisoning, Port Stealing, Switch flooding, DNS and IP Spoofing, Session Hijacking, Sequence Numbers, Ettercap, idle host scanning, Default TTLs, Countermeasures, Packet Crafting using eghping, scapy. | 1. summarize the importance of Scapy Tool. (C2)<br>2. Explain ARP ICMP Protocols. (C2)<br>3. Interpret the packet capture analysis and network flow analysis. (C3) |
| **Unit 4:** | |
| **Metasploit:** Basics of Penetration Testing: The Phase of PTES, Types of | 1. Identify Metasploit Framework. (C2) |

| | |
|---|---|
| Penetration Tests. Metasploit: Introduction, Metasploit Basics: Terminology, Metasploit Interfaces, Metasploit Utilities. Intelligence Gathering: Passive Information Gathering, Active Information Gathering, Target Scanning. Vulnerability Scanning: Basic Vulnerability Scan, Scanning with scanning tools, Using Scan Results for Autopwning. | 2. Explain the steps involved in the penetration Testing. (C2)<br>3. Apply Scanning Tools and gathering information. (C3) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | C4 | C5 | C6 |
|---|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| Sessional Examination 2 | | | * | * | | |
| Assignment/Presentation | * | * | * | * | * | * |
| End Semester Examination | * | * | * | * | * | * |
| Laboratory examination | * | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Cryptography and Network Security - Principles and Practice**, Stallings William, Pearson Education; Seventh edition, 2017.<br>2. **Cryptography And Network Security**, Forouzan, McGraw Hill Education, 2015.<br>3. **Cryptography and Network Security**, Atul Kahate, McGraw Hill Education; Third edition, 2017.<br>4. **Network Security Essentials: Applications and Standards**, William Stallings, Prentice Hall, 4th edition, 2010.<br>5. **Metasploit - The Penetration Tester's Guide by David Kennedy**, Jim O'gorma , Devon Kearns and Mati Aharoni – No Starch Press Publication<br>6. **Internetworking with TCP/IP Vol I : Principles, Protocols and Architecture**, Douglas E Comer, 3rd edition. PHI, 1997.<br>7. **TCP/IP Illustrated, Volume I, The Protocols**, W Richard Stevens, International Student Edition, 1999.<br>8. **RFC's on IPSEC, SSL, TLS, HTTPS, Kerberos** - Internet resources. |

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

| | |
|---|---|
| **Name of the Program:** | M. E. in Blockchain Technology |
| **Course Title:** | Advanced Blockchain Application Development |
| **Course Code:** BCH 604 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 3 | **Prerequisites:** Basics of sensors, Basics of communication |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>The course allows students to have an understanding of the python construct and to build applications for blocking using the web frameworks based on python and JavaScript. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Discuss and Describe Object Oriented Programming concepts with Python |
| **CO 2:** | Develop an blockchain applications using Python and its framework. |
| **CO 3:** | Develop the database application and build it. |
| **CO 4:** | Develop and build application based on REST API. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | * | | * | | | | | | |
| CO 2 | * | * | | | | | | | | | |
| CO 3 | * | * | | | | | | | | | |
| CO 4 | * | * | * | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| Introduction to Application Development – Introduction to procedural vs Object Oriented Language – Introduction to Python - Python datatypes – Constructs - Python Examples with basic constructs | 1. To describe the difference between procedural and object oriented language. (C1)<br>2. To explain the constructs involved in python.(C2) |

| **Unit 2** | |
|---|---|
| Object oriented Concepts – Class – Objects – Encapsulation – Abstraction – Polymorphism – Inheritance - Association – Aggregation – Composition - Object oriented programming with Python - Examples | 1. To describe the OOPS concepts and the way written in Python (C1) |
| **Unit 3:** | |
| Socket Programming with python – database applications - email applications - REST API – Python web frameworks – Buidling Blockchain applications from Scratch – Using Frameworks - Case study | 1. To explain the concept of sockets and the API involved in creating applications (C2)<br>2. To explain the blockchain framework applications development process. (C2) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Seminar | - | - |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | * |
| Assignment/Presentation | * | * | * | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Python: The Complete Reference**, Martin C. Brown , McGraw Hill Education; Forth edition, 2018.<br>2. **Hands-On Python for Finance: A practical guide to implementing financial analysis strategies using Python**, Krish Naik , Packt Publishing Limited , 2019<br>3. **Learning Python: Powerful Object-Oriented Programming**, Mark Lutz, 5th Edition, O'Reilly Media; 5 edition, 2013.<br>4. **Mastering Object-Oriented Python: Build powerful applications with reusable code using OOP design patterns and Python 3.7**, Steven F. Lott, 2nd Edition, 2019<br>5. **Flask Web Development: Developing Web Applications with Python**, Miguel Grinberg, O'Reilly Media; 2 editions, 2018<br>6. **Beginning Web Development with Python: from prototype to production with flask, tornado and nginx**, Andrei Dan, Kindle Edition<br>7. **Hands-On RESTful API Design Patterns and Best Practices: Design, develop, and deploy highly adaptable, scalable, and** |

| | |
|---|---|
| | **secure RESTful web APIs**, Harihara Subramanian , Pethuru Raj, Packt Publishing; 1st edition, 2019.<br>8. **Hands-On Blockchain for Python Developers: Gain blockchain programming skills to build decentralized applications using Python**, Arjuna Sky Kok, Packt Publishing, 1st edition, 2019. |

| | |
|---|---|
| **Name of the Program:** | M. E. in Blockchain Technology |
| **Course Title:** | Blockchain Verification and Testing |
| **Course Code:** BCH 606 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 3 | **Prerequisites:** Basic Programming |
| **Synopsis:** | This Course provides insight on<br>The course allows students to have an understanding of the python construct and to build applications for blocking using the web frameworks based on python and javascript. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Design test cases suitable for a software development for different domains. |
| **CO 2:** | Identify and use the test tools for Blockchain testing and automation. |
| **CO 3:** | Identify suitable tests to be carried out. |
| **CO 4:** | Prepare test planning based on the document. |
| **CO 5:** | Write test plans and test cases designed. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | * | * | * | | | | | | |
| CO 2 | * | * | * | | * | | | | | | |
| CO 3 | * | * | * | | * | | | | | | |
| CO 4 | * | * | * | | * | | | | | | |
| CO 5 | * | * | * | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| INTRODUCTION: Testing as an Engineering Activity – Testing as a Process – Testing axioms – Basic definitions – Software Testing Principles – The Tester's Role in a | 1. To explain the process of testing and its related axioms (C2)<br>2. To describe the concept of defects and it characteristics.(C1) |

| | |
|---|---|
| Software Development Organization – Origins of Defects – Cost of defects – Defect Classes – The Defect Repository and Test Design – Defect Examples – Developer/Tester Support of Developing a Defect Repository – Defect Prevention strategies.<br><br>Testing in Blockchain - Smart Contract Testing - Peer/Node Testing – Introduction to tools for testing blockchain: Ethereum Tester - Truffle - Ganache (formally Testrpc) - Populus - Manticore - Hyperledger Composer - Exonum Testkit - Embark Framework - Corda Testing Tools | 3. To describe the testing in blockchain and smart contacts and its tools.(C1) |
| **Unit 2:** | |
| Overview of the Software - What is Software? - Software Technologies - What is Web Application - Web Application Technologies - Software Development Lifecycle(SDLC): Waterfall Model - Iterative Model - Spiral Model - V-Model - Big Bang Model - Agile Model - RAD Model | 1. To describe the models of SDLC(C1)<br>2. To explain the process involved in each model.(C2) |
| **Unit 3:** | |
| Software Testing Life Cycle - (STLC): Understand Requirement - Create Test Cases - Manual Testing - Automation Testing - Test Report | 1. To write the STLC test cases (C3)<br>2. To differentiate the manual testing and automating testing(C4)<br>3. To prepare a SRS document and create a test plan.(C5) |

| | |
|---|---|
| Software requirements specification - (SRS): What is SRS - Finding gap in SRS - How to Write a Test Plan from SRS Document - How to test software requirements specification (SRS)? - Review SRS Document and Create Test Scenarios<br><br>Functional Testing: Unit Testing - Integration Testing - System Testing Regression Testing - Acceptance Testing<br><br>Non-Functional Testing: Performance Testing - Load Testing - Usability Testing - Security Testing - Portability Testing<br><br>Manual Testing: Writing test scenarios - Test planning - Test case design Test data identification - Reviewing and Execution of Test cases/scripts | |
| **Unit 4:** | |
| Software test automation – skill needed for automation – scope of automation – design and architecture for automation – requirements for a test tool – challenges in automation – Test metrics and measurements – project, progress and productivity metrics. – Need for Automation testing – Tool Selection – | 1. To describe the process of automation (C3)<br>2. To explain the metrics and measurement of test automation. (C2)<br>3. To explain the choice of tools and usage of selenium.(C2) |

| | |
|---|---|
| Frameworks - Test case using Selenium – QTP – and other tools | |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | | |
| Sessional Examination 2 | | | * | * | * |
| Assignment/Presentation | * | * | * | * | * |
| End Semester Examination | * | * | * | * | * |
| Laboratory examination | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |

| | |
|---|---|
| **Reference Material** | 1. **Software Testing – Principles and Practices**, Srinivasan Desikan and Gopalaswamy Ramesh, Pearson Education, 2006.<br>2. **Software Testing**, Ron Patton, Second Edition, Sams Publishing, Pearson Education, 2007<br>3. **Practical Software Testing**, Ilene Burnstein, Springer International Edition, 2003.<br>4. **Software Testing in the Real World – Improving the Process**, Edward Kit, Pearson Education, 1995.<br>5. **Software Testing Techniques**, Boris Beizer, 2nd Edition, Van Nostrand Reinhold, New York, 1990.<br>6. **Foundations of Software Testing _ Fundamental Algorithms and Techniques**, Aditya P. Mathur, Dorling Kindersley (India) Pvt. Ltd., Pearson Education, 2008.<br>7. Ben Laurie, Peter Laurie, "Apache: The Definitive Guide", 3rd Edition, O'Reilly Media,2009.<br>8. Brian Totty, David Gourley, Marjorie Sayer, Anshu Aggarwal, Sailu Reddy, "HTTP: The Definitive Guide", O'Reilly Media, 2009. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Cloud Computing |
| **Course Code**: BCH-616 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 3 | **Prerequisites:** Familiarity in developing application using any high-level language |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>Cloud Computing is the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the Internet ("the cloud") to offer faster innovation, flexible resources, and economies of scale. The students are introduced with the basic understanding of the architecture of cloud computing and its various types. Also the course discusses the characteristics, research issues and application implementation on the cloud. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Describe the need and architecture Distributed Computing paradigms |
| **CO 2:** | Explain the Characteristics and architecture of Cloud Computing |
| **CO 3:** | Compare and contrast service models and deployment models of Cloud |
| **CO 4:** | Explain the concept of Virtualization as a prime Enabling Technology of Cloud Computing |
| **CO 5:** | Explain the concept of Web Services as a prime Enabling Technology of Cloud Computing |
| **CO 6:** | Design an Infrastructure in Cloud for High availability and Fault Tolerant Web applications |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | * | * | | | | | | |
| CO 2 | * | * | | | * | | | | | | |
| CO 3 | * | * | | | * | | | | | | |
| CO 4 | * | * | | | * | | | | | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 5 | * | * | | * | * | | | | | | |
| CO 6 | * | * | | * | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| **Introduction**: Evolution of Cloud Computing, Enabling technologies, Cloud computing infrastructure models, Public, private, and hybrid clouds, Architectural layers of cloud computing, Cloud application programming interfaces, Inside Grid, HPC, Clouds | 1. To explain the architecture of cloud infrastructure (C2)<br>2. To explain the types of the cloud. (C2) |
| **Unit 2:** | |
| **Cloud Architecture:** Models for cloud computing, Types of Clouds and Services, Security, Privacy, and Trust management issues, Cloud Economics and Business Models, Resource management and scheduling, QoS (Quality of Service) and Resource Allocation, Virtual Machines Provisioning and migration services, Support for Market-Aware Cloud Services, Pricing Schemes and Risk Management, SLA (Service Level Agreements) negotiation and management Accounting, Billing and Verification Infrastructure | 1. To explain the virtual machines concepts. (C2)<br>2. To describe the cloud services and it s management. (C2) |
| **Unit 3:** | |

| | |
|---|---|
| **Infrastructure models:** Infrastructure models & its advantages, Private Clouds, Public Clouds, Hybrid Clouds | 1. To explain the types of cloud in detail. (C2) |
| **Unit 4:** | |
| **Important Delivery Mechanisms:** Infrastructure as a Service, Platform as a Service, Software as a Service, Data as a Service, other delivery mechanisms like Globalization as a Service, etc. | 1. To explain the infrastructure, platform, software, and other services in detail. (C2) |
| **Unit 5:** | |
| **Parallelization Concepts:** High Availability, Replication, Load Balancing, Interoperability between Clouds, Internetworking between Clouds (InterClouds) | 1. To explain the concept of load and balancing it. (C2)<br>2. To explain the interoperability of the cloud. (C2) |
| **Unit 6 :** | |
| **Case Study:** Building and Deploying Social Network Applications on Clouds, Portability of applications and data between different cloud providers, Reliability of applications and services running on the cloud, Content Delivery Networks using Storage Clouds, Building and Hosting Internet Service Applications on Cloud, Experience with Building and Using Cloud Infrastructure, Legal issues in Cloud Computing, Business Computing on Clouds | 1. To explain the cloud applications and its characteristics through the case studies discussed. (C2) |
| **Unit 7:** | |

| | |
|---|---|
| **Key Issues:** Recovery, Data Segregation, Underlying Encryption, and the other drawbacks of Cloud Computing | 1. To explain the research issues in cloud computing. (C2) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Seminar | - | - |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 | CO 6 |
|---|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | | | |
| Sessional Examination 2 | | | * | * | | |
| Assignment/Presentation | * | * | * | * | * | * |
| End Semester Examination | * | * | * | * | * | * |
| Laboratory examination | * | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Introduction to Cloud Computing**, Timothy Chou, Active Book Press 2nd Edition<br>2. **Cloud Computing: Principles and Paradigms,** R Buyya, Wiley, 2010.<br>3. **Cloud Computing: Principles, Systems and Applications,** L Gillam, Springer, 2010. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Machine Learning |
| **Course Code**: BCH-617 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 3 | **Prerequisites:** Programming with Python and Data Visualization |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>1. Neurons and biological motivation, activation functions and threshold units, supervised and unsupervised learning, perceptron network models in Artificial Neural Networks.<br>2. Learning from unclassified data using clustering techniques.<br>3. Support Vector Machines for linear and non-linear classification.<br>4. Deep Learning and design of convolutional neural network for Deep Learning.<br>5. Applications and design of Reinforcement Learning algorithms. |
| **Course Outcomes (COs):** | |
| **CO 1:** | Explain concept learning and hypothesis space |
| **CO 2:** | Describe activation functions, weights and threshold units used in artificial neural networks, supervised and unsupervised learning, gradient descent approach, types of perceptron models, overfitting |
| **CO 3:** | Apply Decision Tree, PAC, Bayes and Markov nets, K-NN, SVM, clustering and back propagation models for machine learning |
| **CO 4:** | Analyse different machine learning algorithms |
| **CO 5:** | Design ensemble methods, back propagation neural network, K-means and agglomerative clustering models |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | | | | | | | | | |
| CO 2 | | * | | | | | | | | | |
| CO 3 | | | * | | | | | | | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 4 | | | * | | | | | | | | |
| CO 5 | | | * | | | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| **Introduction:** Definition of learning systems - Goals and applications of machine learning - Aspects of developing a learning system - Training data, concept representation, and function approximation. | 1. Define Machine Learning (C1)<br>2. Describe the applications for which machine learning approaches seem appropriate. (C2)<br>3. Illustrate different types of machine learning techniques (C3) |
| **Unit 2:** | |
| **Inductive Classification:** The concept learning task - Concept learning as search through a hypothesis space - General-to-specific ordering of hypotheses - Finding maximally specific hypotheses - Version spaces and the candidate elimination algorithm - Learning conjunctive concepts. The importance of inductive bias. | 1. Relate concept learning and hypothesis space (C4).<br>2. Apply different algorithms to obtain most general and most specific hypotheses from the training examples. (C3) |
| **Unit 3:** | |
| **Predictive analytics – Supervised learning**<br>**Decision Tree learning:** Representing concepts as decision trees - Recursive induction of decision trees - Picking the best splitting attribute - Entropy and information gain - Searching for simple trees and computational complexity | 1. Construct decision tree machine learning algorithm (C5)<br>2. Explain the method of choosing training examples and target function in the design of a machine learning system (C2)<br>3. Explain different validation technique to find the accuracy in training and testing of data set (C5) |

| | |
|---|---|
| **Ensemble methods (bagging and boosting):** Using committees of multiple hypotheses - Bagging, boosting, and DECORATE - Active learning with ensembles | 4. Choose a suitable method of ensemble learning approach (C3).<br>5. Explain various ensemble techniques (C5) |
| **Unit 4:** | |
| **Computational learning theory:**<br>**Models of learnability:** learning in the limit - Probably approximately correct (PAC) learning - Sample complexity: quantifying the number of examples needed to PAC learn - Computational complexity of training. Sample complexity for finite hypothesis spaces.<br><br>**Bayesian learning:** Probability theory and Bayes rule - Naive Bayes learning algorithm - Parameter smoothing - Generative vs. discriminative training - Logistic regression - Bayes nets and Markov nets for representing dependencies.<br><br>**Instance-based learning:** Constructing explicit generalizations versus comparing to past specific examples - K-Nearest Neighbour algorithm - Case-based learning.<br><br>**Support Vector Machine (SVM):** Maximum margin linear separators - Quadractic programming solution to | 1. Apply K-nearest neighbour, SVM, Logistic Regression and PCA (C3)<br>2. Predict the target value for the new instance using Naïve Bayes classifier. (C3)<br>3. Construct explicit generalizations (C5)<br>4. Discriminate Instances Based and Case-based learning (C4)<br>5. Explain the Kernel trick for learning non-linear functions (C5) |

| | |
|---|---|
| finding maximum margin separators - Kernels for learning non-linear functions. | |
| **Unit 5:** | |
| **Descriptive analytics – unsupervised learning**<br>**Artificial Neural Networks:** Neurons and biological motivation - Linear threshold units -Perceptrons: representational limitation and gradient descent training - Multilayer networks and back propagation - Hidden layers and constructing intermediate, distributed representations – Overfitting. | 1. Relate biological neurons with artificial neurons and the motivation for ANN development. (C1)<br>2. Distinguish Supervised and unsupervised learning (C2).<br>3. Describe about error reduction techniques in used Artificial Neural Networks based learning (C2)<br>4. Write the usability of different activation functions for ANN learning system. (C3)<br>5. Describe the architecture of various perceptron networks. (C2) |
| **Unit 6 :** | |
| **Clustering:** Learning from unclassified data - Clustering. Hierarchical Aglomerative Clustering - Non-Hierarchical Clustering - k-means partitional clustering - Expectation maximization (EM) for soft clustering - Semi-supervised learning with EM using labeled and unlabled data. | 1. Write the different methods of learning from unclassified data (C3).<br>2. Explain the operations of various clustering models in machine learning (C5)<br>3. Describe the methods used for measuring dissimilarity between two clusters. (C2)<br>4. Apply clustering techniques for data analysis. (C3) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |

| | | |
|---|---|---|
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | * | * |
| Sessional Examination 2 | * | * | * | * | * |
| Assignment/Presentation | * | * | * | * | |
| End Semester Examination | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Pattern Recognition and Machine Learning**, Christopher M. Bishop. Springer,1st Edition, 2006.<br>2. **Machine Learning**, Tom Mitchell, McGraw-Hill Education, 1st edition, 1997.<br>3. **An introduction to support vector machines**, Cristianini, N. and J. Shawe-Taylor. Cambridge University Press, Cambridge University Press, 1st edition, 2000.<br>4. **Machine Learning: The Art and Science of Algorithms that Make Sense of Data**, Flach, Peter. Cambridge University Press, 1st edition, 2012.<br>5. **Artificial Intelligence: A Modern Approach,** Russell, Stuart and Peter Norvig, Prentice Hall, 3rd Edition, 2009. |

| | |
|---|---|
| | 6. **Pattern Classification**, Duda, R., P. Hart, and D. Stork. Wiley Publishers, Second Edition,2000.<br>7. **A Course in Machine Learning**, Hal Daumé III . Available online at http://ciml.info/<br>8. **Analytics in a Big Data World**, Bart Baesens. Wiley,1st Edition, 2014.<br>9. **Ensemble Learning,** Thomas G. Dietteri in The Handbook of Brain Theory and Neural Networks, Second edition, (M.A. Arbib, Editor), Cambridge, MA: The MIT Press, 2002.<br>10. **Generative and discriminative classifiers: naïve Bayes and logistic regression.** Available online at http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Entrepreneurship |
| **Course Code**: ENP-601 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 3 | **Prerequisites:** |

| | |
|---|---|
| **Synopsis:** | This course introduces students to the theory of entrepreneurship and its practical implementation. It focuses on different stages related to the entrepreneurial process, including business model innovation, monetization, small business management as well as strategies that improve performance of new business ventures. Centered on a mixture of theoretical exploration as well as case studies of real-world examples and guest lectures, students will develop an understanding of successes, opportunities and risks of entrepreneurship. This course has an interdisciplinary approach and is therefore open to students from other Majors. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to: |
| **CO 1:** | To impart knowledge on the basics of entrepreneurial skills and competencies to provide the participants with necessary inputs for creation of new ventures. |
| **CO 2:** | To familiarize the participants with the concept and overview of entrepreneurship with a view to enhance entrepreneurial talent |
| **CO 3:** | To appraise the entrepreneurial process starting with pre-venture stage |
| **CO 4:** | To Create and exploit innovative business ideas and market opportunities |
| **CO 5:** | To Build a mind-set focusing on developing novel and unique approaches to market opportunities |
| **CO 6:** | To explore new vistas of entrepreneurship in 21st century environment to generate innovative business ideas through case studies. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | | | | | | | | | |
| CO 2 | | | | * | | | | | | | |
| CO 3 | | | * | | | | | | | | |
| CO 4 | | | | | | * | | | | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 5 | | | | | | | | * | | | |
| CO 6 | | | | | | | | | | * | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1: Introduction to Entrepreneurship** | |
| Meaning and Definition of Entrepreneurship-Employment vs Entrepreneurship, Theories of Entrepreneurship, approach to entrepreneurship, Entrepreneurs VS Manager | 1. Explain the meaning of Entrepreneurship (C1)<br>2. Discuss the theories of Entrepreneurship (C1)<br>3. Discuss the approaches to Entrepreneurship (C1) |
| **Unit 2: Entrepreneurial Traits** | |
| Personality of an entrepreneur, Types of Entrepreneurs | 1. Discuss the Personality traits of entrepreneurs. (C2) |
| **Unit 3: Process of Entrepreneurship** | |
| Factors affecting Entrepreneurship process | 1. Identify the fundamentals and responsibilities of entrepreneurship (C2)<br>2. Exemplify one's capabilities in relation to the rigors of successful ventures (C3)<br>3. Identify and differentiates the different characteristics and competencies of an entrepreneurs (C2) |
| **Unit 4: Business Start-up Process** | |
| Idea Generation, Scanning the Environment, Macro and Micro analysis | 1. Explain the Process of Business start up (C1)<br>2. Develop creativity and critical thinking in identifying opportunities (C5)<br>3. Apply innovative approaches in envisioning ones entrepreneurial career (C3) |
| **Unit 5: Business Plan writing** | |
| Points to be considered, Model Business plan | 1. Identify different business models (C3)<br>2. Describe different parts of a business plan(C2) |
| **Unit 6: Case studies** | |

| | |
|---|---|
| Indian and International Entrepreneurship | 1. Perform self-assessment and analyse entrepreneurial personal traits and competencies (C4)<br>2. Evaluate oneself and plan courses of action to help develop one's entrepreneurial characteristics and competencies. (C5) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 30 | 60 |
| Quiz | 02 | 04 |
| Small Group Discussion (SGD) | 02 | 02 |
| Self-directed learning (SDL) | - | 04 |
| Problem Based Learning (PBL) | 02 | 04 |
| Case Based Learning (CBL) | - | - |
| Revision | 02 | - |
| Assessment | 06 | - |
| **TOTAL** | **44** | **74** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 | CO 6 |
|---|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | | | | |
| Sessional Examination 2 | | | * | * | | |
| Assignment/Presentation | | | | | * | * |
| End Semester Examination | * | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | • End-Semester Feedback |
| **Reference Material** | 1. NVR Naidu and T. Krishna Rao, "Management and Entrepreneurship", IK International Publishing House Pvt. Ltd 2008.<br>2. Mohanthy Sangram Keshari, "Fundamentals of Entrepreneurship", PHI Publications, 2005<br>3. Butler, D. (2006). Enterprise planning and development. USA: Elsevier Ltd. Gerber, M.E. (2008) Awakening the entrepreneur within. NY: Harper Collins. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Devops for Cloud Lab |
| **Course Code:** CDC 607L | **Course Instructor:** |
| **Academic Year:** 2020-2021 | **Semester:** Year, Semester |
| **No of Credits:** 1 | **Prerequisites:** Ubuntu OS, Networking and Software Life Cycle |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on:<br>1. Devops Product Life Cycles Stage.<br>2. Automation of product lifecycle. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Explain the concept of automation of Product Life Cycle stages. |
| **CO 2:** | Design an Devops methodologies for Product development and Release |
| **CO 3:** | Demonstrate Continuous Integration / Continuous Testing / Continuous Deployment of Product. |
| **CO4:** | Explain the concepts of Tools used in each stages of Devops. |
| **CO5:** | Demonstrate Continuous Monitoring of Production Environment. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | * | | | | | | | | |
| CO 2 | * | * | * | | | | | | | | |
| CO 3 | * | * | * | | | | | | | | |
| CO 4 | | * | * | | * | | | | | | |
| CO 5 | | * | * | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1: DevOps Introduction** | |

| | |
|---|---|
| Understanding Development - Developement SDLC : WaterFall & Agile - Understanding Operations - Dev vs Ops - DevOps to the rescue - What is DevOps - DevOps SDLC - Continous Delivery model - DevOps tools for DevOps SDLC - DevOps Roles & Responsiblities. | 1. Demonstrate differences between Waterfall and agile software development methodologies (C2) |
| **Unit 2: Linux** | |
| Linux Introduction, Principles & Linux distro – Booting - Command line utililities & Basic commands - Linux Filesystem - Filters & I/O Redirections - Users & Group administration - File permissions & Ownerships - Sudo - Software Managemen - Useful tools: ssh, telnet, scp, rsync, disk utils, backups etc - Service & Process management - Shell Scripting - Systems and HW stats – Linux Containers (lxc) - Dockers – Kubernetes and Microservices | 1. Design Ubuntu based VM using hypervisor to understand booting process , linux file system , linux networking , Users , Groups and Permissions, tools (ssh , scp etc ) ( C3)<br>2. Design a docker environment to containerize web application (C3)<br>3. Design a Kubernetes cluster to deploy containerized application using Kubernetes deployment and service models ( C4) |
| **Unit 3: Networking fundamentals** | |
| Components of computer networks - Classification: LAN, WAN, Peer to Peer network, Server based - Switches - Routers - Network Architecture | 1. Design a College/ University network using packet tracer to understand computer networking devices like Hub , Switches , Routers and Firewalls ( C3) |

| | |
|---|---|
| - Protocols - Port numbers - DNS - DHCP - IP Addresses - Ip Addresses & Subnet Masks - IP Address Ranges - Subnetting - Private Vs Public networks - High Availaiblity - Firewalls & NACL - Web Application Architecture - Infrastructure - Net work layout - Services & Components - Architecture from a DevOps perspective. | 2. Design a Network project using Packet tracer to understand Networking services like DNS , DHCP , FTP etc ( C3) |
| **Unit 4: Automation, Orchestration & Config Management** | |
| Version control system with Git : What is VCS & why it is needed - DevOps use cases - Setup your own repo with git - Manage your code base/source code with GIT & GITHUB | 1. Create Github account and set up repository and use git commands to Clone , Fork and commit files to Github repositories ( C4) |
| **Unit 5: Continuous Integration with Jenkins** | |
| Introduction to continuous integration. - Build & Release and relation with DevOps - Understanding development and developers - Why Continuous integration Jenkins introduction and setup - Jenkins projects/jobs - Jenkins plugins \| Jenkins administration: Users - Nodes/slaves - Managing plugins - Managing software versions - Introduction - Phases - Java builds - Build and Release job/project setup \| Nexus: Intro & Setup | 1. Design a Continuous Integration server using Jenkins in Master Slave architecture ( C3)<br>2. Demonstrate CI/CD for JAVA/PHP/nodejs web application ( C4)<br>3. Design an Eclipse Selenium testing project to automate Web application Testing Process ( C4) |

| | |
|---|---|
| - Software versioning & Hosted repository - Integration with Jenkins - Continuous integration job/project setup \| Complete Jenkins project: Packinging Artifacts - Static code Analysis - Tomcat setup Staging & productions - Artifacts deployments to webservers from Jenkins - Build Pipeline - Jenkins not just CI tool anymore - More DevOps use cases of Jenkins | |
| **Unit 6: Ansible** | |
| Configuration Management & Automation - What is Ansible & its features - Ansible setup on local & cloud - Understanding Ansible architecture & Execution - Inventory \| Ad hoc commands: Automating change Management with Ad Hoc commands - Playbook Introduction - Ansible configuration with ansible.cfg - Ansible documentation - Modules, modules & lots of modules - Writing playbook for webserver & DB server deployments - Tasks - Variables - Templates - Loops - Handlers - Conditions - Register - Debugging - Ansile Roles - Identify server roles - Roles structure - Creating, Managing and executing roles | 1. Design a Configuration management service using Ansible to administer group of nodes in lab ( C2)<br>2. Demonstrate installation of Software packages like git , Eclipse , Mysql on group of nodes using Ansible ( C4)<br>3. Design a Continuous monitoring server using Nagios to monitor group of servers for different dervices like CPU Utilization , RAM Usage , Network Bandwidth , Apache server logs , Database server logs etc (C5) |

| | |
|---|---|
| - Ansible Galaxy - Exploring Roles from Galaxy - Download Galaxy roles and integrate with your code - Ansible Advanced Execution - Improving execution time - Limiting and selecting tasks - Troubleshooting and Testing | |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|

| | | | | | |
|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | * | * |
| Assignment/Presentation | | | | * | * |
| Laboratory Examination | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. Eric Foster-Johnson , John C. Welch , Micah Anderson, Beginning Shell Scripting (Programmer to Programmer), Wrox Publications<br>2. Randal K. Michael "Mastering Unix Shell Scripting: Bash, Bourne, and Korn Shell Scripting for Programmers, System Administrators, and UNIX Gurus", 2nd Edition, Wiley Publications<br>3. Bintu Harwani, "UNIX & Shell Programming", Oxford Publications, 2013<br>4. John Ferguson Smart, "Jenkins: The Definitive Guide",O'reilly Publications<br>5. Mitesh Soni, "Jenkins Essentials", Packt Publications<br>6. Rafal Leszko, "Continuous Delivery with Docker and Jenkins", Packt Publications<br>7. Veselin Kantsev, "Implementing DevOps on AWS", Packt Publications<br>8. Randall Smith, "Docker Orchestration", Packt Publications<br>9. Alan Berg, "Jenkins Continuous Integration Cookbook", Packt Publications<br>10. Kumaran S., Senthil, " Practical LXC and LXD Linux Containers for Virtualization and Orchestration", Apress Publications<br>11. Konstantin Ivanov, " Containerization with LXC" , Packt Publications |

| | |
|---|---|
| | 12. Karl Matthias, Sean Kane, "Docker: Up & Running:Shipping Reliable Container in Production",O'Reilly Media |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Network Security and Analysis Lab |
| **Course Code:** BCH 602L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 1 | **Prerequisites:** Microprocessor architecture , Microcontroller Architecture , Assembly language and Number systems |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>The course allows students to understand the network devices and the data formats for different protocols. As lot of data is populated in the present world an analysis of these packets using the modern tools help to understand the attacks and the nature of the network. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Identify some of the factors driving the need for network security |
| **CO 2:** | Identify and classify particular examples of attacks |
| **CO 3:** | Define the terms vulnerability, threat and attack |
| **CO4:** | Identify physical points of vulnerability in simple networks |
| **CO5:** | Compare and contrast symmetric and asymmetric encryption systems and their vulnerability to attack, and explain the characteristics of hybrid systems. |
| **CO6:** | Analyze the packets through the network tools |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | * | | | | | | | |
| CO 2 | * | * | | | | | | | | | |
| CO 3 | * | * | | | | | | | | | |
| CO 4 | * | * | | | | | | | | | |
| CO 5 | | | | * | | | | | | | |
| CO 6 | | | * | * | | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |

| | |
|---|---|
| INTERNET SECURITY & ENCRYPTION | At the end of the topic student should be able to : Apply encryption algorithm to protect the attack on confidentiality (C13) |
| **Unit 2:** | |
| FIREWALLS: | Model secure network infrastructure for the given scenario. (C4) |
| **Unit 3:** | |
| SNIFFERS AND PACKET CRAFTING: | Demonstrate network analysis using Wireshark (tshark), tcpdump, Scapy, Snot tools to identify malicious behavior in the network traffic. (C1) |
| **Unit 4:** | |
| Metasploit | Test for security vulnerabilities using Metasploit tools. (C4) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |

| | | |
|---|---|---|
| **Assessment Methods:** | | |
| **Formative:** | | **Summative:** |
| Internal practical Test | | Sessional examination |
| Theory Assignments | | End semester examination |
| Lab Assignment & Viva | | Viva |
| | | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 | CO 6 |
|---|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | | | |
| Sessional Examination 2 | | | | * | * | * |
| Assignment/Presentation | * | * | * | * | * | * |
| Laboratory examination | * | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Cryptography and Network Security - Principles and Practice**, Stallings William, Pearson Education; Seventh edition, 2017.<br>2. **Cryptography And Network Security**, Forouzan, McGraw Hill Education, 2015.<br>3. **Cryptography and Network Security**, Atul Kahate, McGraw Hill Education; Third edition, 2017.<br>4. **Network Security Essentials: Applications and Standards**, William Stallings, Prentice Hall, 4th edition, 2010.<br>5. **Metasploit - The Penetration Tester's Guide by David Kennedy**, Jim O'gorma , Devon Kearns and Mati Aharoni – No Starch Press Publication<br>6. **Internetworking with TCP/IP Vol I : Principles, Protocols and Architecture**, Douglas E Comer, 3rd edition. PHI, 1997.<br>7. **TCP/IP Illustrated, Volume I, The Protocols**, W Richard Stevens, International Student Edition, 1999.<br>8. **RFC's on IPSEC, SSL, TLS, HTTPS, Kerberos** - Internet resources. |

| | |
|---|---|
| **Name of the Program:** | M. E. in Blockchain Technology |
| **Course Title:** | Advanced Blockchain Application Development Lab |
| **Course Code:** BCH 604L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits**: 1 | **Prerequisites:** Basics of sensors, Basics of communication |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>The course allows students to have an understanding of the python construct and to build applications for blocking using the web frameworks based on python and JavaScript. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Discuss and Describe Object Oriented Programming concepts with Python |
| **CO 2:** | Develop an blockchain applications using Python and its framework. |
| **CO 3:** | Develop the database application and build it. |
| **CO 4:** | Develop and build application based on REST API. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | * | | * | | | | | | |
| CO 2 | * | * | * | | * | | | | | | |
| CO 3 | * | * | * | | * | | | | | | |
| CO 4 | * | * | * | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |

| | |
|---|---|
| Introduction to Application Development – Introduction to procedural vs Object Oriented Language – Introduction to Python - Python datatypes – Constructs - Python Examples with basic constructs | 1. To illustrate to solve problems using the constructs involved in python (C2) |
| **Unit 2** | |
| Object oriented Concepts – Class – Objects – Encapsulation – Abstraction – Polymorphism – Inheritance - Association – Aggregation – Composition - Object oriented programming with Python - Examples | 1. To illustrate to solve problems the OOPS concepts and the way written in Python (C2) |
| **Unit 3:** | |
| Socket Programming with python – database applications - email applications - REST API – Python web frameworks – Buidling Blockchain applications from Scratch – Using Frameworks - Case study | 1. To illustrate and to solve problems based on socket programming in python. (C2)<br>2. To illustrate applications based on the blockchain framework. (C2) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |

| | | |
|---|---|---|
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | - |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 |
|---|---|---|---|---|
| Sessional Examination 1 | * | * | | |
| Sessional Examination 2 | | | * | * |
| Assignment/Presentation | * | * | * | * |
| End Semester Examination | * | * | * | * |
| Laboratory examination | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Python: The Complete Reference**, Martin C. Brown , McGraw Hill Education; Forth edition, 2018.<br>2. **Hands-On Python for Finance: A practical guide to implementing financial analysis strategies using Python**, Krish Naik , Packt Publishing Limited , 2019<br>3. **Learning Python: Powerful Object-Oriented Programming**, Mark Lutz, 5th Edition, O'Reilly Media; 5 edition, 2013.<br>4. **Mastering Object-Oriented Python: Build powerful applications with reusable code using OOP design patterns and Python 3.7**, Steven F. Lott, 2nd Edition, 2019 |

| | |
|---|---|
| | 5. **Flask Web Development: Developing Web Applications with Python**, Miguel Grinberg, O'Reilly Media; 2 editions, 2018<br>6. **Beginning Web Development with Python: from prototype to production with flask, tornado and nginx**, Andrei Dan, Kindle Edition<br>7. **Hands-On RESTful API Design Patterns and Best Practices: Design, develop, and deploy highly adaptable, scalable, and secure RESTful web APIs**, Harihara Subramanian , Pethuru Raj, Packt Publishing; 1st edition, 2019.<br>8. **Hands-On Blockchain for Python Developers: Gain blockchain programming skills to build decentralized applications using Python**, Arjuna Sky Kok, Packt Publishing, 1st edition, 2019. |

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Blockchain Verification and Testing LAB |
| **Course Code:** BCH 606L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits: 1** | **Prerequisites:** Basic Programming |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>The course allows students to have an understanding of the python construct and to build applications for blocking using the web frameworks based on python and javascript. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Design test cases suitable for a software development for different domains. |
| **CO 2:** | Identify and use the test tools for Blockchain testing and automation |
| **CO 3:** | Identify suitable tests to be carried out |
| **CO 4:** | Prepare test planning based on the document |
| **CO 5:** | Write test plans and test cases designed |
| **C0 6:** | Use of automatic testing tools |
| **CO 7:** | Develop and validate a test plan |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | * | * | * | | | | | | |
| CO 2 | * | * | * | | * | | | | | | |
| CO 3 | * | * | * | | * | | | | | | |
| CO 4 | * | * | * | | * | | | | | | |
| CO 5 | * | * | * | | * | | | | | | |
| CO 6 | * | * | * | | * | | | | | | |
| CO 7 | * | * | * | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |

| | |
|---|---|
| INTRODUCTION: Testing as an Engineering Activity – Testing as a Process – Testing axioms – Basic definitions – Software Testing Principles – The Tester's Role in a Software Development Organization – Origins of Defects – Cost of defects – Defect Classes – The Defect Repository and Test Design – Defect Examples – Developer/Tester Support of Developing a Defect Repository – Defect Prevention strategies.<br><br>Testing in Blockchain - Smart Contract Testing - Peer/Node Testing – Introduction to tools for testing blockchain: Ethereum Tester - Truffle - Ganache (formally Testrpc) - Populus - Manticore - Hyperledger Composer - Exonum Testkit - Embark Framework - Corda Testing Tools | 1. To identify the required platform and tools for testing of blockchain. (C2)<br>2. To illustrate the testing in blockchain and smart contacts and its tools. (C2) |
| **Unit 2:** | |
| Overview of the Software - What is Software? - Software Technologies - What is Web Application - Web Application Technologies - Software Development Lifecycle(SDLC): Waterfall Model - Iterative Model - Spiral Model - V-Model - Big Bang Model - Agile Model - RAD Model | 1. To illustrate the process involved in model through UML diagrams. (C2) |
| **Unit 3:** | |

| | |
|---|---|
| Software Testing Life Cycle - (STLC): Understand Requirement - Create Test Cases - Manual Testing - Automation Testing - Test Report<br><br>Software requirements specification - (SRS): What is SRS - Finding gap in SRS - How to Write a Test Plan from SRS Document - How to test software requirements specification (SRS)? - Review SRS Document and Create Test Scenarios<br><br>Functional Testing: Unit Testing - Integration Testing - System Testing Regression Testing - Acceptance Testing<br><br>Non-Functional Testing: Performance Testing - Load Testing - Usability Testing - Security Testing - Portability Testing<br><br>Manual Testing: Writing test scenarios - Test planning - Test case design<br>Test data identification - Reviewing and Execution of Test cases/scripts | 1. Compose a SRS document and create a test plan. (C6) |
| **Unit 4:** | |
| Software test automation – skill needed for automation – scope of automation – design and architecture for automation – requirements for a test tool – challenges in automation – Test metrics and | 4. To illustrate the test cases using selenium. (C2) |

| | |
|---|---|
| measurements – project, progress and productivity metrics. – Need for Automation testing – Tool Selection – Frameworks - Test case using Selenium – QTP – and other tools | |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 | CO 6 | C7 |
|---|---|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | | | | |
| Sessional Examination 2 | | | | * | * | * | * |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Assignment/Presentation | * | * | * | * | * | * | * |
| End Semester Examination | * | * | * | * | * | * | * |
| Laboratory examination | * | * | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Software Testing – Principles and Practices**, Srinivasan Desikan and Gopalaswamy Ramesh, Pearson Education, 2006.<br>2. **Software Testing**, Ron Patton, Second Edition, Sams Publishing, Pearson Education, 2007<br>3. **Practical Software Testing**, Ilene Burnstein, Springer International Edition, 2003.<br>4. **Software Testing in the Real World – Improving the Process**, Edward Kit, Pearson Education, 1995.<br>5. **Software Testing Techniques**, Boris Beizer, 2nd Edition, Van Nostrand Reinhold, New York, 1990.<br>6. **Foundations of Software Testing _ Fundamental Algorithms and Techniques**, Aditya P. Mathur, Dorling Kindersley (India) Pvt. Ltd., Pearson Education, 2008.<br>7. Ben Laurie, Peter Laurie, "Apache: The Definitive Guide", 3rd Edition, O'Reilly Media,2009.<br>8. Brian Totty, David Gourley, Marjorie Sayer, Anshu Aggarwal, Sailu Reddy, "HTTP: The Definitive Guide", O'Reilly Media, 2009. |

MANIPAL ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

| | |
|---|---|
| **Name of the Program:** | M. E. in Blockchain Technology |
| **Course Title:** | Cloud Computing Lab |
| **Course Code**: BCH-616L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 1 | **Prerequisites:** Familiarity in developing application using any high-level language |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>Cloud Computing is the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the Internet ("the cloud") to offer faster innovation, flexible resources, and economies of scale. The students are introduced with the basic understanding of the architecture of cloud computing and its various types. Also the course discusses the characteristics, research issues and application implementation on the cloud. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | To identify the type of cloud available in the market |
| **CO 2:** | To setup a private cloud and understand the steps involved |
| **CO 3:** | To illustrate the various steps involved in using the public cloud services |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | | * | | | | | | |
| CO 2 | * | * | | | * | | | | | | |
| CO 3 | * | * | | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| **Introduction**: Evolution of Cloud Computing, Enabling technologies, Cloud computing infrastructure models, Public, private, and hybrid clouds, | 1. To identify the type of cloud platforms available in the marks. (C3) |

| | |
|---|---|
| Architectural layers of cloud computing, Cloud application programming interfaces, Inside Grid, HPC, Clouds | |
| **Unit 2:** | |
| **Cloud Architecture:** Models for cloud computing, Types of Clouds and Services, Security, Privacy, and Trust management issues, Cloud Economics and Business Models, Resource management and scheduling, QoS (Quality of Service) and Resource Allocation, Virtual Machines Provisioning and migration services, Support for Market-Aware Cloud Services, Pricing Schemes and Risk Management, SLA (Service Level Agreements) negotiation and management Accounting, Billing and Verification Infrastructure | 1. To illustrate, install and manage the virtual machines (C2) |
| **Unit 3:** | |
| **Infrastructure models:** Infrastructure models & its advantages, Private Clouds, Public Clouds, Hybrid Clouds | 1. To illustrate the installation steps in a private cloud. (C2)<br>2. To illustrate the usage of public cloud (C2) |
| **Unit 4:** | |
| **Important Delivery Mechanisms:** Infrastructure as a Service, Platform as a Service, Software as a Service, Data as | 1. To explore and illustrate the services of the cloud. (C2) |

| | |
|---|---|
| a Service, other delivery mechanisms like Globalization as a Service, etc. | |
| **Unit 5:** | |
| **Parallelization Concepts:** High Availability, Replication, Load Balancing, Interoperability between Clouds, Internetworking between Clouds (InterClouds) | 1. To illustrate the concept of load balancing. (C2) |
| **Unit 6 :** | |
| **Case Study:** Building and Deploying Social Network Applications on Clouds, Portability of applications and data between different cloud providers, Reliability of applications and services running on the cloud, Content Delivery Networks using Storage Clouds, Building and Hosting Internet Service Applications on Cloud, Experience with Building and Using Cloud Infrastructure, Legal issues in Cloud Computing, Business Computing on Clouds | 1. To illustrate the application on the cloud (private/public/both) |
| **Unit 7:** | |
| **Key Issues:** Recovery, Data Segregation, Underlying Encryption, and the other drawbacks of Cloud Computing | 1. To illustrate the application of cloud security. |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |

| | | |
|---|---|---|
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 |
|---|---|---|---|
| Sessional Examination 1 | * | * | |
| Sessional Examination 2 | | * | * |
| Assignment/Presentation | * | | * |
| Laboratory examination | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Introduction to Cloud Computing**, Timothy Chou, Active Book Press 2nd Edition<br>2. **Cloud Computing: Principles and Paradigms,** R Buyya, Wiley, 2010.<br>3. **Cloud Computing: Principles, Systems and Applications,** L Gillam, Springer, 2010. |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Machine Learning Lab |
| **Course Code**: BCH-617L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits**: 1 | **Prerequisites:** Programming with Python and Data Visualization |

| | |
|---|---|
| **Synopsis:** | This Course provides insight on<br>1. Neurons and biological motivation, activation functions and threshold units, supervised and unsupervised learning, perceptron network models in Artificial Neural Networks.<br>2. Learning from unclassified data using clustering techniques.<br>3. Support Vector Machines for linear and non-linear classification.<br>4. Deep Learning and design of convolutional neural network for Deep Learning.<br>5. Applications and design of Reinforcement Learning algorithms. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Apply activation functions, weights and threshold units used in artificial neural networks, supervised and unsupervised learning, gradient descent approach, types of perceptron models, overfitting |
| **CO 2:** | Demonstrate Decision Tree, PAC, Bayes and Markov nets, K-NN, SVM, clustering and back propagation models |
| **CO 3:** | Analyse different machine learning algorithms |
| **CO 4:** | Design ensemble methods, back propagation neural network, K-means and agglomerative clustering models |
| **CO 5:** | Apply activation functions, weights and threshold units used in artificial neural networks, supervised and unsupervised learning, gradient descent approach, types of perceptron models, overfitting |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | * | | | * | | | | | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 2 | * | * | | | * | | | | | | |
| CO 3 | * | | * | | * | | | | | | |
| CO 4 | * | | | * | * | | | | | | |
| CO 5 | * | | | | * | | | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Unit 1:** | |
| **Introduction:** Definition of learning systems - Goals and applications of machine learning - Aspects of developing a learning system - Training data, concept representation, and function approximation. | 1. Identify programming environments available for the machine learning (C1)<br>2. Classify the pros and cons of various environments for ML coding (C2) |
| **Unit 2:** | |
| **Inductive Classification:** The concept learning task - Concept learning as search through a hypothesis space - General-to-specific ordering of hypotheses - Finding maximally specific hypotheses - Version spaces and the candidate elimination algorithm - Learning conjunctive concepts. The importance of inductive bias. | 1. Design a machine learning model to get a Maximally Specific Hypothesis for the given training examples (C5).<br>2. Construct a machine learning model to obtain most general and most specific hypotheses for the given training examples (C5) |
| **Unit 3:** | |
| **Predictive analytics – Supervised learning**<br>**Decision Tree learning:** Representing concepts as decision trees - Recursive induction of decision trees - Picking the best splitting attribute - Entropy and | 1. Develop a machine learning classifier using decision tree and random forest (C5)<br>2. Develop machine learning models using Ensemble models. (C5) |

| | |
|---|---|
| information gain - Searching for simple trees and computational complexity<br><br>**Ensemble methods (bagging and boosting):** Using committees of multiple hypotheses - Bagging, boosting, and DECORATE - Active learning with ensembles | |
| **Unit 4:** | |
| **Computational learning theory:**<br><br>**Models of learnability:** learning in the limit - Probably approximately correct (PAC) learning - Sample complexity: quantifying the number of examples needed to PAC learn - Computational complexity of training. Sample complexity for finite hypothesis spaces.<br><br>**Bayesian learning:** Probability theory and Bayes rule - Naive Bayes learning algorithm - Parameter smoothing - Generative vs. discriminative training - Logistic regression - Bayes nets and Markov nets for representing dependencies.<br><br>**Instance-based learning:** Constructing explicit generalizations versus comparing to past specific examples - K-Nearest Neighbour algorithm - Case-based learning. | 1. Design a learning method to determine the sample complexity of training examples (C5)<br>2. Analyse bias-variance trade-off, under-fitting and over-fitting concepts (C4)<br>3. Design a machine learning model using K-NN, SVM, Bayes learning, Bayesian and Markov Networks (C5).<br>4. Develop a machine learning classifier models using different approach (C5) |

| | |
|---|---|
| **Support Vector Machine (SVM):** Maximum margin linear separators - Quadractic programming solution to finding maximum margin separators - Kernels for learning non-linear functions. | |
| **Unit 5:** | |
| **Descriptive analytics – unsupervised learning**<br><br>**Artificial Neural Networks:** Neurons and biological motivation - Linear threshold units -Perceptrons: representational limitation and gradient descent training - Multilayer networks and back propagation - Hidden layers and constructing intermediate, distributed representations – Overfitting. | 1. Demonstrate activation functions, weights and threshold units in artificial neural networks (C3)<br>2. Design of ANN models for classification (C5) |
| **Unit 6 :** | |
| **Clustering:** Learning from unclassified data - Clustering. Hierarchical Aglomerative Clustering - Non-Hierarchical Clustering - k-means partitional clustering - Expectation maximization (EM) for soft clustering - Semi-supervised learning with EM using labeled and unlabled data. | 1. Analyze the performance of clustering techniques on different data (C4)<br>2. Apply clustering techniques for data analysis. (C3) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |

| | | |
|---|---|---|
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | **-** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|
| Sessional Examination 1 | * | * | * | | |
| Sessional Examination 2 | | | | * | * |
| Assignment/Presentation | * | * | * | * | * |
| End Semester Examination | * | * | * | * | * |
| Laboratory examination | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | End-Semester Feedback |
| **Reference Material** | 1. **Pattern Recognition and Machine Learning**, Christopher M. Bishop. Springer,1st Edition, 2006.<br>2. **Machine Learning**, Tom Mitchell, McGraw-Hill Education, 1st edition, 1997. |

3. **An introduction to support vector machines**, Cristianini, N. and J. Shawe-Taylor. Cambridge University Press, Cambridge University Press, 1st edition, 2000.
4. **Machine Learning: The Art and Science of Algorithms that Make Sense of Data**, Flach, Peter. Cambridge University Press, 1st edition, 2012.
5. **Artificial Intelligence: A Modern Approach,** Russell, Stuart and Peter Norvig, Prentice Hall, 3rd Edition, 2009.
6. **Pattern Classification**, Duda, R., P. Hart, and D. Stork. Wiley Publishers, Second Edition,2000.
7. **A Course in Machine Learning**, Hal Daumé III . Available online at http://ciml.info/
8. **Analytics in a Big Data World**, Bart Baesens. Wiley,1st Edition, 2014.
9. **Ensemble Learning,** Thomas G. Dietteri in The Handbook of Brain Theory and Neural Networks, Second edition, (M.A. Arbib, Editor), Cambridge, MA: The MIT Press, 2002.
10. **Generative and discriminative classifiers: naïve Bayes and logistic regression.** Available online at http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | **Entrepreneurship Lab** |
| **Course Code:** ENP-601L | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 1 | **Prerequisites: -** |

| | |
|---|---|
| **Synopsis:** | This course introduces students to the theory of entrepreneurship and its practical implementation. It focuses on different stages related to the entrepreneurial process, including business model innovation, monetization, small business management as well as strategies that improve performance of new business ventures. Cantered on a mixture of theoretical exploration as well as case studies of real-world examples and guest lectures, students will develop an understanding of successes, opportunities and risks of entrepreneurship. This course has an interdisciplinary approach and is therefore open to students from other Majors. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Understand the concept of entrepreneurship |
| **CO 2:** | To appraise the entrepreneurial process starting with pre-venture stage through group discussion |
| **CO 3:** | To Build a mind-set focusing on developing novel and unique approaches to market opportunities by considering case studies and understand the complete flow of entrepreneurship |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | | | | * | | * | | | |
| CO 2 | | | | | | * | | | | | |
| CO 3 | | | | | | | | * | | * | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|

| **Unit 1: Introduction to Entrepreneurship** | |
|---|---|
| Meaning and Definition of Entrepreneurship-Employment vs Entrepreneurship, Theories of Entrepreneurship, approach to entrepreneurship, Entrepreneurs VS Manager | 1. Discuss the theories of Entrepreneurship (C1)<br>2. Discuss the approaches to Entrepreneurship (C1) |
| **Unit 2: Process of Entrepreneurship** | |
| Factors affecting Entrepreneurship process | 1. Exemplify one's capabilities in relation to the rigors of successful ventures (C3)<br>2. Identify and differentiates the different characteristics and competencies of an entrepreneurs (C2) |
| **Unit 3: Business Plan writing** | |
| Points to be considered, Model Business plan | 1. Identify different business models (C3)<br>2. Describe different parts of a business plan(C2) |
| **Unit 4: Case studies** | |
| Indian and International Entrepreneurship | 1. Perform self-assessment and analyse entrepreneurial personal traits and competencies (C4)<br>2. Evaluate oneself and plan courses of action to help develop one's entrepreneurial characteristics and competencies. (C5) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | 12 | - |
| Seminar | - | - |
| Quiz | - | - |

| | | |
|---|---|---|
| Small Group Discussion (SGD) | - | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | 03 | - |
| Clinic | - | - |
| Practical | 24 | - |
| Revision | 03 | - |
| Assessment | 06 | - |
| **TOTAL** | **48** | - |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Internal practical Test | Sessional examination |
| Theory Assignments | End semester examination |
| Lab Assignment & Viva | Viva |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 |
|---|---|---|---|
| Sessional Examination 1 | * | * | |
| Sessional Examination 2 | | | * |
| Assignment/Presentation | | * | * |
| Laboratory Examination | * | * | * |

| | |
|---|---|
| **Feedback Process** | • End-Semester Feedback |
| **Reference Material** | 1. NVR Naidu and T. Krishna Rao, "Management and Entrepreneurship", IK International Publishing House Pvt. Ltd 2008.<br>2. Mohanthy Sangram Keshari, "Fundamentals of Entrepreneurship", PHI Publications, 2005 |

| | |
|---|---|
| | 3. Butler, D. (2006). Enterprise planning and development. USA: Elsevier Ltd. Gerber, M.E. (2008) Awakening the entrepreneur within. NY: Harper Collins. |

| | |
|---|---|
| **Course Title:** | Mini Project - 2 |
| **Course Code**: BCH 696 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |
| **No of Credits:** 4 | **Prerequisites:** Any programming language and circuit basics |

| | |
|---|---|
| **Synopsis:** | Students are expected to select a problem in the area of their interest and the area of their specialization that would require an implementation in hardware / software or both in a semester |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Apply the objectives of the project work and provide an adequate background with a detailed literature survey |
| **CO 2:** | Breakdown the project into sub blocks with sufficient details to allow the work to be reproduced by an independent researcher |
| **CO 3:** | Compose hardware/software design, algorithms, flowchart, methodology, and block diagram |
| **CO 4:** | Evaluate the results |
| **CO 5:** | Summarize the work carried out |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | | | | * | | | | | | | |
| CO 2 | | | | | * | | | * | | | |
| CO 3 | | | | | | | * | | | * | |
| CO 4 | | | | | | * | | | | | * |
| CO5: | | | | | | | * | | | | |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Phase 1** | |
| Problem identification, synopsis submission, status submission, mid evaluation. | At the end of the topic student should be able to:<br>1. Identify the problem/specification (C1)<br>2. Discuss the project (C2) |

| | |
|---|---|
| | 3. Prepare the outline (C3)<br>4. Describe the status of the project (C2)<br>5. Prepare a mid-term project presentation report (C3)<br>6. Prepare and present mid-term project presentation slides (C3, C5)<br>7. Develop project implementation in hardware/software or both in chosen platform (C5) |
| **Phase 2** | |
| Status submission, final evaluation. | 1. Prepare the progress report (C3)<br>2. Prepare the final project presentation report (C3)<br>3. Prepare and present final project presentation slides (C3, C5)<br>4. Modify and Develop implementation in hardware/software or both in chosen platform (C3, C5)<br>5. Justify the methods used and obtained results (C6) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | - | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | 48 | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Practical | - | - |

| | | |
|---|---|---|
| Revision | - | - |
| Assessment | 03 | - |
| **TOTAL** | **51** | **09** |
| | | |

| **Assessment Methods:** | |
|---|---|
| **Formative:** | **Summative:** |
| Project Problem Selection | Mid-Term Presentation |
| Synopsys review | Second status review |
| First status review | Demo & Final Presentation |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|
| Mid Presentation | * | * | | | |
| Presentation | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | • End-Semester Feedback |
| **Reference Material** | Particular to the chosen project |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Seminar - 2 |
| **Course Code**: BCH 698 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** First Year, Semester 2 |

| **No of Credits:** 1 | **Prerequisites:** Communication Skill |
|---|---|
| **Synopsis:** | 1. To select, search and learn technical literature.<br>2. To Identify a current and relevant research topic.<br>3. To prepare a topic and deliver a presentation.<br>4. To develop the skill to write a technical report.<br>5. Develop ability to work in groups to review and modify technical content. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | Show competence in identifying relevant information, defining and explaining topics under discussion. |
| **CO 2:** | Show competence in working with a methodology, structuring their oral work, and synthesizing information. |
| **CO 3:** | Use appropriate registers and vocabulary, and will demonstrate command of voice modulation, voice projection, and pacing. |
| **CO 4:** | Demonstrate that they have paid close attention to what others say and can respond constructively. |
| **CO 5:** | Develop persuasive speech, present information in a compelling, well-structured, and logical sequence, respond respectfully to opposing ideas, show depth of knowledge of complex subjects, and develop their ability to synthesize, evaluate and reflect on information. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | * | | | | | | | * | * | | * |
| CO 2 | * | | | | | | | * | * | | * |
| CO 3 | * | | | | | | | * | * | | * |
| CO 4 | * | | | | | | | * | * | | * |
| CO5: | * | | | | | | | * | * | | * |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|

| | | |
|---|---|---|
| Lecture | - | - |
| Seminar | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | 14 | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Practical | - | - |
| Revision | - | - |
| Assessment | - | - |
| **TOTAL** | **14** | **-** |
| | | |

**Assessment Methods:**

| **Formative:** | **Summative:** |
|---|---|
| Seminar Topic Selection | |
| Synopsys review | |
| PPT Review | |
| | |

**Mapping of assessment with Cos**

| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
|---|---|---|---|---|---|
| Presentation | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | • End-Semester Feedback |
| **Reference Material** | Particular to the chosen Seminar |

| | |
|---|---|
| **Name of the Program:** | Master of Engineering - ME (Blockchain Technology) |
| **Course Title:** | Project Work |
| **Course Code**: BCH 799 | **Course Instructor:** |
| **Academic Year:** 2020 - 2021 | **Semester:** Second Year, Semester 3, 4 |

| | |
|---|---|
| **No of Credits:** 25 | **Prerequisites:** SDLC, Communication Skills, technical skills. |
| **Synopsis:** | The project work aims to challenge analytical, creative ability and to allow students to synthesize, apply the expertise and insight learned in the core discipline.<br>Students build self-confidence, demonstrate independence, and develop professionalism on successfully completion of the project. |
| **Course Outcomes (COs):** | On successful completion of this course, students will be able to |
| **CO 1:** | To be acquainted with working environment and processes that in place at the relevant Industries. |
| **CO 2:** | To familiarize the challenges as relevant professionals. |
| **CO 3:** | Review the literature and develop solutions for real time onboard projects. |
| **CO 4:** | Write technical report and deliver presentation. |
| **CO 5:** | Apply engineering and management principles to achieve project goal. |

**Mapping of COs to POs**

| *COs* | *PO 1* | *PO 2* | *PO 3* | *PO 4* | *PO 5* | *PO 6* | *PO 7* | *PO 8* | *PO 9* | *PO 10* | *PO 11* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO 1 | | | | | | * | * | * | * | * | * |
| CO 2 | | | | | * | | | | | | |
| CO 3 | * | * | * | * | * | | | | | | |
| CO 4 | * | * | * | * | | | | | | | |
| CO5: | | | | | | * | * | * | * | * | * |

**Course content and outcomes:**

| ***Content*** | ***Competencies*** |
|---|---|
| **Phase 1:** | |
| Problem identification, synopsis submission, status submission, mid evaluation. | At the end of the topic student should be able to:<br>1. Identify the problem/specification (C1)<br>2. Discuss the project (C2)<br>3. Prepare the outline (C3) |

| | |
|---|---|
| | 4. Prepare a mid-term project presentation report (C3)<br>5. Prepare and present mid-term project presentation slides (C3, C5)<br>6. Develop project implementation in hardware/software or both in chosen platform (C5) |
| **Phase 2** | |
| Status submission, final evaluation. | 1. Prepare the progress report (C3)<br>2. Prepare the final project presentation report (C3)<br>3. Prepare and present final project presentation slides (C3, C5)<br>4. Modify and Develop implementation in hardware/software or both in chosen platform (C3, C5)<br>5. Justify the methods used and obtained results (C6) |

**Learning strategies, contact hours and student learning time**

| *Learning strategy* | *Contact hours* | *Student learning time (Hrs)* |
|---|---|---|
| Lecture | - | - |
| **Seminar** | - | - |
| Quiz | - | - |
| Small Group Discussion (SGD) | 14 | - |
| Self-directed learning (SDL) | - | - |
| Problem Based Learning (PBL) | - | - |
| Case Based Learning (CBL) | - | - |
| Clinic | - | - |
| Practical | - | - |
| Revision | - | - |
| Assessment | - | - |

| | | |
|---|---|---|
| **TOTAL** | **14** | **-** |
| | | |

| | |
|---|---|
| **Assessment Methods:** | |
| **Formative:** | **Summative:** |
| Project Problem Selection | Mid-Term Presentation |
| Synopsys review | Second status review |
| First status review | Demo & Final Presentation |
| | |

| **Mapping of assessment with Cos** | | | | | |
|---|---|---|---|---|---|
| Nature of assessment | CO 1 | CO 2 | CO 3 | CO 4 | CO 5 |
| Mid Presentation | * | * | | | |
| Presentation | * | * | * | * | * |

| | |
|---|---|
| **Feedback Process** | • End-Semester Feedback |
| **Reference Material** | Particular to the chosen project |

# PROGRAM OUTCOMES (POS) AND COURSE OUTCMES (COS) MAPPING

| Sl.No. | Course Code | Course Name | Credits | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | CSE 601 | Data Structures and Algorithms | 3 | * | * | | * | | * | | | | | |
| 2 | BCH 601 | Blockchain Application Development | 3 | * | * | * | * | * | | * | | | | |
| 3 | BCH 603 | Blockchain Technology | 3 | * | * | * | * | * | | | | | | |
| 4 | BCH 605 | Cryptocurrency and Smart Contracts | 3 | * | * | * | * | * | | | | | | |
| 5 | BCH-615 | Distributed Computing and Databases | 3 | * | * | * | * | * | | | | | | |
| 6 | BDA-623 | Architecture of Big Data Systems | 3 | * | * | * | | * | * | * | | | * | |
| 7 | CSE-604 | Database Programming in Java | 3 | * | * | * | * | * | | | | | | |
| 8 | CSE 601L | Data Structures and Algorithms Lab | 1 | | * | * | | * | | | * | | | |
| 9 | BCH 601L | Blockchain Application Development Lab | 1 | * | * | * | * | * | | | | | | |
| 10 | BCH 603L | Blockchain Technology Lab | 1 | * | * | * | | * | | | | | | |
| 11 | BCH 605L | Cryptocurrency and Smart Contracts Lab | 1 | * | * | * | * | * | | | | | | |
| 12 | BCH-615L | Distributed Computing and Databases Lab | 1 | * | * | * | * | * | | | | | | |
| 13 | BDA-623L | Architecture of Big Data Systems Lab | 1 | * | * | * | | * | * | | | * | * | |
| 14 | CSE-604L | Database Programming in Java Lab | 1 | | * | * | | * | | | | | | |
| 15 | BCH 695 | Mini Project – I | 4 | | | | * | * | * | * | * | | * | * |
| 16 | BCH 697 | Seminar – I | 1 | * | | | | | | | * | * | | * |
| 17 | CDC 607 | DevOps for Cloud | 3 | * | * | * | | * | | | | | | |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 18 | BCH 602 | Network Security and Analysis | 3 | * | * | * | * | | | | | | | |
| 19 | BCH 604 | Advanced Blockchain Application Development | 3 | * | * | * | | * | | | | | | |
| 20 | BCH 606 | Blockchain Verification and Testing | 3 | * | * | * | * | * | | | | | | |
| 21 | BCH-616 | Cloud Computing | 3 | * | * | | * | * | | | | | | |
| 22 | BCH-617 | Machine Learning | 3 | * | * | * | | | | | | | | |
| 23 | ENP-601 | Entrepreneurship | 3 | * | | * | * | | * | | * | | * | |
| 24 | CDC 607L | DevOps for Cloud Lab | 1 | * | * | * | | * | | | | | | |
| 25 | BCH 602L | Network Security and Analysis Lab | 1 | * | * | * | * | | | | | | | |
| 26 | BCH 604L | Advanced Blockchain Application Development Lab | 1 | * | * | * | | * | | | | | | |
| 27 | BCH 606L | Blockchain Verification and Testing Lab | 1 | * | * | * | * | * | | | | | | |
| 28 | BCH-616L | Cloud Computing Lab | 1 | * | * | | | * | | | | | | |
| 29 | BCH-617L | Machine Learning Lab | 1 | * | * | * | * | * | | | | | | |
| 30 | ENP-601L | Entrepreneurship Lab | 1 | * | | | | | * | | * | | * | |
| 23 | BCH 696 | Mini Project – II | 4 | | | | * | * | * | * | * | | * | * |
| 24 | BCH 698 | Seminar – II | 1 | * | | | | | | | * | * | | * |
| 25 | BCH 799 | Project Work | 25 | * | * | * | * | * | * | * | * | * | * | * |