Curriculum

Of

Bachelor of Science in Information Technology

BS (IT)

Revised: Fall 2018

Department of Informatics and Systems

School of Systems and Technology

University of Management and Technology, Lahore, Pakistan

Table of Contents

1. Curriculum Review Committee

Following are the members of the curriculum review committee who were involved in the revision of the BS-IT program.

1. Dr. Shaukat Iqbal

Professor and Dean, School of Systems and Technology

2. Dr. Sajid Mahmood

Assistant Professor and Chairperson, Department of Informatics and Systems

3. Dr. Ghulam Mustafa

Assistant Professor, Department of Informatics and Systems

4. Mr. Amjad Hussain Zahid

Assistant Professor and Program Advisor, Department of Informatics and Systems

5. Syed Mohsin Ali

Lecturer, Department of Informatics and Systems

2. Preface

Bachelor of Science in Information Technology (BSIT) program at Department of Informatics and Systems in SST was introduced in 2012. This program was developed keeping in view the emerging and drastically increasing trends of information technology in Pakistan and around the globe. The curriculum had been designed to focus on the application areas of IT in the organizations.

In 2017, Higher Education Commission (HEC) released revised version for guidelines of curriculum for BSIT program. The program here at UMT has now matured enough to consider the changes and incorporate those guidelines. Therefore, in this document, we present a revised version of BSIT curriculum which is intended to be applicable from Fall 2018. The revised curriculum has been designed to get aligned with HEC guidelines and also foster the technology shifts in industry.

3. Acknowledgment

We acknowledge the honorable rector of UMT for his vision and motivation to make our academic programs up-to-date by including state-of-the-art courses, materials and practices. We also acknowledge the support of our faculty members for sharing their experiences and pieces of advice to improve the curriculum.

4. Program Objectives

BSIT program has been deigned to achieve the following objectives:

- Equip the students with fundamental and advanced concepts of computing and mathematics and application of these concepts in diversified domains that require IT solutions.
- Develop the expertise to analyze the systems rigorously and to design the solutions enabling them to plan, select, integrate, deploy, manage and support the required IT resources
- Apply their equipped knowledge and skills to develop a career in an information technology oriented business or industry, or for graduate study in computer science or business fields
- Include emphasis on effective communication, decision-making and leadership skills

We have incorporated the following industrial trends in our course structure and programs focusing on analysis and design:

BSIT Program is being tailored to develop students' skills for Designing, Handling and Maintaining IT Systems in different sectors of life.

5. Program Structure

Course Category	Credit Hours	No. of Courses
Computing Core	39	11
IT Core	24	7
IT Supporting	9	3
IT Electives	15	5
University Electives	12	4
General Education	21	7
Math. and Science Foundation	12	4
Total	132	41

Category-wise course distribution:

5.1. Computing Core Courses (9 Courses + Final Year Project)

1. Programming Fundamentals
2. Object Oriented Programming
3. Discrete Structures
4. Data Structures and Algorithms
5. Operating Systems
6. Database Systems
7. Software Engineering
8. Computer Networks
9. Information Security
10. Final Year Project

5.2. Program Core (7 Courses)

1. Cyber Security
2. Database Administration and Management
3. Information Technology Project Management
4. Information Technology Infrastructure
5. System and Network Administration
6. Virtual Systems and Services
7. Web Technologies

5.3. Supporting Core (4 Courses)

1. Calculus and Analytical Geometry
2. Probability and Statistics
3. Linear Algebra
4. Applied Physics

5.4. General Education Courses (7 Courses)

1. English Composition & Comprehension
2. Composition and Communication
3. Research Paper Writing and Presentation
4. Islamic Studies / Ethics
5. Pakistan Studies
6. Professional Practices
7. Introduction to Information & Communication Technologies

5.5. Program Electives 5 Courses

1. Advanced Networks
2. Advanced Web Technologies
3. Big Data Programming
4. Bioinformatics
5. Blockchain Technology and Application
6. Cloud Computing
7. Cognitive Radio Communication and Networks
8. Competitive Programming
9. Component Based Software Engineering
10. Computer Animations
11. Computer Graphics
12. Computer Vision
13. Cyber Forensics
14. Data Communications
15. Data Mining
16. Data Science & Big Data Analytics
17. Data Science Technologies
18. Data Warehousing
19. Database Administration
20. Deep Learning and Neural Networks
21. Design Pattern and Refactoring
22. Digital Image Processing
23. Distributed Database Systems
24. Distributed Systems
25. Embedded Systems
26. Enterprise Application Development
27. Enterprise Architecture
28. Expert Systems and Knowledge Management
29. Free Space Optics
30. Games Design and Development
31. Information Retrieval
32. Information Systems
33. Internet of Things
34. Iphone Applications Development
35. Machine Learning
36. Mobile Application Development
37. Multi-Agent Systems
38. Multimedia Security
39. Natural Language Processing
40. Network Flows
41. Next Generation Networks

42. Pervasive Smart Environments
43. Secure Software Development
44. System Integration and Architecture
45. Web Technologies

5.6 Supporting Electives 3 Courses

1. Enterprise Systems
2. Modeling and Simulation
3. Formal Methods
4. Operations Research
5. Software Requirements Engineering
6. Analysis of Algorithms

5.7 University Electives 4 Courses

1. Fundamentals of Economics
2. Research Project and Presentation
3. Technical Writing and Presentation Skills
4. Organizational Behavior
5. IT Entrepreneurship
6. Principles of Marketing
7. Introduction to Logic
8. Business Ethics
9. Life, Learning and Leadership
10. Foreign Language
11. Social Impact of Cyber warfare
12. Computing Technologies and Privacy
13. Enterprise Resource Planning
14. Social Networks and society
15. E- Governance
16. Information System Auditing
17. Cyber-Militias and Political Hackers
18. Green Computing
19. History of Computing

Note: Elective courses can be offered from the list as required and decided every semester. The list is also available on SST Website and is not exhaustive.

6. Semester wise Road Map

Total Credit Hours 132

1st Year							
Fall Semester				**Spring Semester**			
Code	**Course Title**	**Cr. Hrs**	**Pre req.**	**Code**	**Course Title**	**Cr. Hrs**	**Pre req.**
IT1091	Introduction to ICT	2	- - - -	CC1022	Object Oriented Programming	3	CC1021, CC1021L
IT1091L	Introduction to ICT Lab	1	- - - -	CC1022L	Object Oriented Programming Lab	1	CC1021, CC1021L
EN111	English Grammar & Comprehension	3	- - - -	CC1041	Discrete Structures	3	- - - -
MA100	Calculus & Analytical Geometry	3	- - - -	EN125	Composition and Communication	3	EN111
CC1021	Programming Fundamentals	3	- - - -		University Elective – I	3	- - - -
CC1021L	Programming Fundamentals Lab	1	- - - -		University Elective – II	3	- - - -
NS125	Applied Physics	2	- - - -				
NS125L	Applied Physics Lab	1	- - - -				
Semester Credit Hours		**16**		**Semester Credit Hours**		**16**	
2nd Year							
Code	**Course Title**	**Cr. Hrs**	**Pre req.**	**Code**	**Course Title**	**Cr. Hrs**	**Pre req.**
CC2042	Data Structures & Algorithms	3	CC1022, CC1022L		IT Supporting Course – II (e.g. SE2102 - SRE)	3	- - - -
CC2042L	Data Structures & Algorithms Lab	1	CC1022, CC1022L	IT2231	IT Infrastructure	3	- - - -
MA150	Probability & Statistics	3	- - - -	CC3071	Computer Networks	3	- - - -
CC2101	Software Engineering	3	- - - -	CC3071L	Computer Networks Lab	1	- - - -
	IT Supporting Course – I (e.g. IT2234 - Enterprise Systems)	3	- - - -		University Elective – III	3	- - - -
MA210	Linear Algebra	3	- - - -	CC2141	Database Systems	3	CC2042, CC2042L
				CC2141L	Database Systems Lab	1	CC2042, CC2042L
Semester Credit Hours		**16**		**Semester Credit Hours**		**17**	

3rd Year							
Fall Semester				**Spring Semester**			
Code	**Course Title**	**Cr. Hrs**	**Pre req.**	**Code**	**Course Title**	**Cr. Hrs**	**Pre req.**
	IT Elective – I	3	- - - -	IT3161	Web Technologies	3	- - - -
ISL101	Islamic Studies	3	- - - -	IT3072	System and Network Administration	3	CC3011, CC3011L
CC3011	Operating Systems	3	CC2042, CC2042L	IT3072L	System and Network Administration Lab	1	CC3011, CC3011L
CC3011L	Operating Systems Lab	1	CC2042, CC2042L	CC3121	Information Security	3	- - - -
IT3142	Database Administration and Management	3	CC2141, CC2141L	EN125	IT Elective – III	3	- - - -
IT3142L	Database Administration and Management Lab	1	CC2141, CC2141L	IT3171	Virtual Systems and Services	3	- - - -
	IT Elective – II	3	- - - -	IT3171L	Virtual Systems and Services Lab	1	- - - -
Semester Credit Hours		**17**		**Semester Credit Hours**		**17**	

4th Year							
Code	**Course Title**	**Cr. Hrs**	**Pre req.**	**Code**	**Course Title**	**Cr. Hrs**	**Pre req.**
CC4181	Final Year Project – I	3	- - - -	CC4182	Final Year Project – II	3	CC4181
EN220	Research Paper Writing and Presentation	3	EN125		University Elective – IV	3	- - - -
IT4122	Cyber Security	3	- - - -		IT Elective – V	3	- - - -
	IT Elective – IV	3	- - - -	HU4092	Professional Practices	3	- - - -
IT4191	IT Project Management	3	- - - -	POL101	Pakistan Studies	3	- - - -
	IT Supporting Course – III (e.g. IT4052-Operations Research)	3	- - - -				
Semester Credit Hours		**18**		**Semester Credit Hours**		**15**	

7. Course outlines

In this section, we present the course outlines of courses for BSIT program as per HEC guidelines.

Computing Core Courses

Programming Fundamentals			
Credit Hours:	4 (3,1)	**Prerequisites:**	Nill

Course Learning Outcomes (CLOs):

1. Understand basic problem solving steps and logic constructs

2. Apply basic programming concepts

3. Design and implement algorithms to **solve** real world problems.

Course Content:

Introduction to problem solving, a brief review of Von-Neumann architecture, Introduction to programming, role of compiler and linker, introduction to algorithms, basic data types and variables, input/output constructs, arithmetic, comparison and logical operators, conditional statements and execution flow for conditional statements, repetitive statements and execution flow for repetitive statements, lists and their memory organization, multi-dimensional lists, introduction to modular programming, function definition and calling, stack rolling and unrolling, string and string operations, pointers/references, static and dynamic memory allocation, File I/O operations

Teaching Methodology:

Lectures, Written Assignments, Practical labs, Semester Project, Presentations

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

1. Starting out with Programming Logic & Designs, 4th Edition, Tony Gaddis, 2015.
2. Object Oriented Programming in C++ by Robert Lafore 2001.
3. C How to Program, 8th Edition by Paul Deitel & Harvey Deitel 2015.
4. Problem Solving and Program Design in C++, 7th Edition by Jeri R. Hanly & Elliot B. Koffman 2000.

Object Oriented Programming			
Credit Hours:	4 (3,1)	**Prerequisites:**	Programming Fundamentals

Course Learning Outcomes (CLOs):

1. Understand principles of object oriented paradigm.

2. Identify the objects & their relationships to build object oriented solution

3. Model a solution for a given problem using object oriented principles

4. Examine an object oriented solution.

Course Content:

Introduction to object oriented design, history and advantages of object oriented design, introduction to object oriented programming concepts, classes, objects, data encapsulation, constructors, destructors, access modifiers, const vs non-const functions, static data members & functions, function overloading, operator overloading, identification of classes and their relationships, composition, aggregation, inheritance, multiple inheritance, polymorphism, abstract classes and interfaces, generic programming concepts, function & class templates, standard template library, object streams, data and object serialization using object streams, exception handling.

Teaching Methodology:

Lectures, Written Assignments, Practical labs, Semester Project, Presentations

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

1. C++ How to Program, 10th Edition, Deitel & Deitel. 2016.
2. Object Oriented Programming in C++, 3rd Edition by Robert Lafore 1998.
3. Java: How to Program, 9th Edition by Paul Deitel, 2011.
4. Beginning Java 2, 7th Edition by Ivor Horton, 2011.
5. An Introduction to Object Oriented Programming with Java, 5th Edition by C. Thomas Wu, 2009.
6. Starting Out with C++ from Control Structures to Objects, 9th Edition, Tony Gaddis, 2017.

Discrete Structures			
Credit Hours:	3 (3,0)	**Prerequisites:**	Nill

Course Learning Outcomes (CLOs):

1. **Understand** the key concepts of Discrete Structures such as Sets, Permutations, Relations, Graphs, and Trees etc.

2. **Apply** formal logic proofs and/or informal, but rigorous, logical reasoning to real problems, such as predicting the behavior of software or solving problems such as puzzles.

3. **Apply** discrete structures into other computing problems such as formal specification, verification, databases, artificial intelligence, and cryptography.

4. **Differentiate** various discrete structures and their relevance within the context of computer science, in the areas of data structures and algorithms, in particular.

* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain

Course Content:

Introduction to object oriented design, history and advantages of object oriented design, introduction to object oriented programming concepts, classes, objects, data encapsulation, constructors, destructors, access modifiers, const vs non-const functions, static data members & functions, function overloading, operator overloading, identification of classes and their relationships, composition, aggregation, inheritance, multiple inheritance, polymorphism, abstract classes and interfaces, generic programming concepts, function & class templates, standard template library, object streams, data and object serialization using object streams, exception handling.

Teaching Methodology:

Lectures, Written Assignments, Practical labs, Semester Project, Presentations

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

1. Discrete Mathematics with Applications, 4th Edition by Susanna S. Epp, 2010.
2. Discrete Mathematics, 7th edition by Richard Johnson Baugh, 2007.
3. Discrete Mathematical Structures, 4th edition by Kolman, Busby & Ross, 1999.
4. Discrete and Combinatorial Mathematics: An Applied Introduction by Ralph P. Grimaldi, 2003.
5. Logic and Discrete Mathematics: A Computer Science Perspective by Winifred Grassman, 1995.

6. Discrete Mathematics and Its Applications, 7th edition by Kenneth H. Rosen, 2011.

Data Structures and Algorithms

Credit Hours:	4 (3,1)	**Prerequisites:**	Object Oriented Programming

Course Learning Outcomes (CLOs):

1. Implement various data structures and their algorithms, and **apply** them in implementing simple applications.

2. **Analyze** simple algorithms and determine their complexities.

3. **Apply** the knowledge of data structures to other application domains.

4. **Design** new data structures and algorithms to solve problems.

Course Content:

Abstract data types, complexity analysis, Big Oh notation, Stacks (linked lists and array implementations), Recursion and analyzing recursive algorithms, divide and conquer algorithms, Sorting algorithms (selection, insertion, merge, quick, bubble, heap, shell, radix, bucket), queue, de-queue, priority queues (linked and array implementations of queues), linked list & its various types, sorted linked list, searching an unsorted array, binary search for sorted arrays, hashing and indexing, open addressing and chaining, trees and tree traversals, binary search trees, heaps, M-way tress, balanced trees, graphs, breadth-first and depth-first traversal, topological order, shortest path, adjacency matrix and adjacency list implementations, memory management and garbage collection.

Teaching Methodology:

Lectures, Written Assignments, Practical labs, Semester Project, Presentations

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

1. Data Structures and Algorithms in C++ by Adam Drozdek, 2012.
2. Starting Out with Java: From Control Structures through Data Structures, 4th Edition, Tony Gaddis, Pearson; 4th Edition, 2018
3. Data Structures and Algorithm Analysis in Java by Mark A. Weiss, 2011.
4. Data Structures and Abstractions with Java by Frank M. Carrano & Timothy M. Henry, 2014.
5. Data Structures and Algorithm Analysis in C++ by Mark Allen Weiss, 2005.
6. Java Software Structures: Designing and Using Data Structures by John Lewis and Joseph Chase, 2013.

Operating Systems			
Credit Hours:	4 (3,1)	**Prerequisites:**	Data Structures and Algorithms

Course Learning Outcomes (CLOs):

1. **Understand** the characteristics of different structures of the Operating Systems and identify the core functions of the Operating Systems.

2. **Analyze** and **evaluate** the algorithms of the core functions of the Operating Systems and explain the major performance issues with regard to the core functions.

3. **Demonstrate** the knowledge in applying system software and tools available in modern operating systems.

Course Content:

Operating systems basics, system calls, process concept and scheduling, inter-process communication, multithreaded programming, multithreading models, threading issues, process scheduling algorithms, thread scheduling, multiple-processor scheduling, synchronization, critical section, synchronization hardware, synchronization problems, deadlocks, detecting and recovering from deadlocks, memory management, swapping, contiguous memory allocation, segmentation & paging, virtual memory management, demand paging, thrashing, memory-mapped files, file systems, file concept, directory and disk structure, directory implementation, free space management, disk structure and scheduling, swap space management, system protection, virtual machines, operating system security

Teaching Methodology:

Lectures, Written Assignments, Practical labs, Semester Project, Presentations

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

1. Operating Systems Concepts, 10th edition by Abraham Silberschatz, 2018.
2. Modern Operating Systems, 4th edition by Andrew S. Tanenbaum, 2014.
3. Operating Systems, Internals and Design Principles, 9th edition by William Stallings, 2017.

Database Systems			
Credit Hours:	4 (3,1)	**Prerequisites:**	Data Structures and Algorithms

Course Learning Outcomes (CLOs):

1. **Explain** fundamental database concepts.

2. **Design** conceptual, logical and physical database schemas using different data models.

3. **Identify** functional dependencies and resolve database anomalies by normalizing database tables.

4. **Use** Structured Query Language (SQL) for database definition and manipulation in any DBMS

Course Content:

Basic database concepts, Database approach vs file based system, database architecture, three level schema architecture, data independence, relational data model, attributes, schemas, tuples, domains, relation instances, keys of relations, integrity constraints, relational algebra, selection, projection, Cartesian product, types of joins, normalization, functional dependencies, normal forms, entity relationship model, entity sets, attributes, relationship, entity-relationship diagrams, Structured Query Language (SQL), Joins and sub-queries in SQL, Grouping and aggregation in SQL, concurrency control, database backup and recovery, indexes, NoSQL systems.

Teaching Methodology:

Lectures, Written Assignments, Practical labs, Semester Project, Presentations

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

1. Database Systems: A Practical Approach to Design, Implementation, and Management, 6th Edition by Thomas Connolly and Carolyn Begg, 2014.
2. Database Systems: The Complete Book, 2nd Edition by Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom, 2008.
3. Database System Concepts, 6th Edition by Avi Silberschatz, Henry F. Korth and S. Sudarshan. 2010.
4. Database Management Systems, 3rd Edition by Raghu Ramakrishnan, Johannes Gehrke, 2002

Software Engineering			
Credit Hours:	3 (3,0)	**Prerequisites:**	Nill

Course Learning Outcomes (CLOs):

1. Describe various software engineering processes and activities

2. Apply the system modeling techniques to model a medium size software system

3. Apply software quality assurance and testing principles to medium size software system.

4. Discuss key principles and common methods for software project management such as scheduling, size estimation, cost estimation and risk analysis

Course Content:

Nature of Software, Overview of Software Engineering, Professional software development, Software engineering practice, Software process structure, Software process models, Agile software Development, Agile process models, Agile development techniques, Requirements engineering process, Functional and non-functional requirements, Context models, Interaction models, Structural models, behavioral models, model driven engineering, Architectural design, Design and implementation, UML diagrams, Design patterns, Software testing and quality assurance, Software evolution, Project management and project planning, configuration management, Software Process improvement.

Teaching Methodology:

Lecturing, Written Assignments, Project, Report Writing

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

1. Software Engineering, Sommerville I., 10th Edition, Pearson Inc., 2014
2. Software Engineering, A Practitioner's Approach, Pressman R. S.& Maxim B. R., 8th Edition, McGraw-Hill, 2015.

Computer Networks			
Credit Hours:	4 (3,1)	**Prerequisites:**	Nill

Course Learning Outcomes (CLOs):

1. **Describe** the key terminologies and technologies of computer networks

2. **Explain** the services and functions provided by each layer in the Internet protocol stack.

3. **Identify** various internetworking devices and protocols, and their functions in a network.

4. **Analyze** working and performance of key technologies, algorithms and protocols.

5. Build Computer Network on various Topologies

Course Content:

Introduction and protocols architecture, basic concepts of networking, network topologies, layered architecture, physical layer functionality, data link layer functionality, multiple access techniques, circuit switching and packet switching, LAN technologies, wireless networks, MAC addressing, networking devices, network layer protocols, IPv4 and IPv6, IP addressing, sub netting, CIDR, routing protocols, transport layer protocols, ports and sockets, connection establishment, flow and congestion control, application layer protocols, latest trends in computer networks.

Teaching Methodology:

Lectures, Written Assignments, Practical labs, Semester Project, Presentations

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

1. Computer Networking: A Top-Down Approach Featuring the Internet, 7th edition by James F. Kurose and Keith W. Ross, 2017.
2. Computer Networks, 5th Edition by Andrew S. Tanenbaum, 2010.
3. Data and Computer Communications, 10th Edition by William Stallings, 2013.
4. Data Communication and Computer Networks, 5th Edition by Behrouz A. Forouzan, 2012.

Information Security			
Credit Hours:	3 (3,0)	**Prerequisites:**	Nill

Course Learning Outcomes (CLOs):

1. **Explain** key concepts of information security such as design principles, cryptography, risk management, and ethics

2. **Discuss** legal, ethical, and professional issues in information security.

3. **Apply** various security and risk management tools for achieving information security and privacy.

4. **Identify** appropriate techniques to tackle and solve problems in the discipline of information security.

Course Content:

Information security foundations, security design principles; security mechanisms, symmetric and asymmetric cryptography, encryption, hash functions, digital signatures, key management, authentication and access control; software security, vulnerabilities and protections, malware, database security; network security, firewalls, intrusion detection; security policies, policy formation and enforcement, risk assessment, cybercrime, law and ethics in information security, privacy and anonymity of data.

Teaching Methodology:

Lectures, Written Assignments, Semester Project, Presentations

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

1. Computer Security: Principles and Practice, 4th edition by William Stallings, 2017.
2. Principles of Information Security, 6th edition by M. Whitman and H. Mattord, 2017.
3. Computer Security, 3rd edition by Dieter Gollmann, 2011.
4. Computer Security Fundamentals, 3rd edition by William Easttom, 2016.
5. Official (ISC)2 Guide to the CISSP CBK, 3rd edition, 2012.

Program Core

Cyber Security			
Credit Hours:	3 (3,0)	**Prerequisites:**	Nill

Course Learning Outcomes (CLOs):

At the end of the course the students will be able to:

1. Evaluate the computer network and information security needs of an organization.
2. Assess cyber security risk management policies in order to adequately protect an organization's critical information and assets.
3. Measure the performance of security systems within an enterprise-level information system.
4. Troubleshoot, maintain and update an enterprise-level information security system.

Course Content:

Basic security concepts, Information security terminology, Malware classifications, Types of malware. Server side web applications attacks. Cross-site scripting, SQL Injection, Cross-site request forgery, Planning and policy, Network protocols and service models. Transport layer security, Network layer security, Wireless security, Cloud & IoT security.

Teaching Methodology:

Lecturing, Written Assignments, Project

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Lab, Presentation, Final Exam

Reference Materials:

1. Security+ Guide to Network Security Fundamentals by Mark Ciampa, 5th Edition, 2018

2. Corporate Computer Society by Randall J.Boyle, 3rd Edition, 2016

Database Administration & Management			
Credit Hours:	4 (3,1)	**Prerequisites:**	Database Systems

Course Learning Outcomes (CLOs):

At the end of the course the students will be able to:

1. Critical analysis: Analyze and model requirements and constraints for the purposes of installing, configuring, and tuning a DBMS, and implementing security, back-up and recovery measures.
2. Problem solving: Design and implement plans for installing, configuring, and tuning a DBMS, and security, back-up and recovery measures, based on requirements analysis/ modeling or a requirements specification.

Course Content:

Introduction to advance data models such as object relational, object oriented. File organizations concepts, Transactional processing and Concurrency control techniques, Recovery techniques, Query processing and optimization, Database Programming, Integrity and security, Database Administration, Physical database design and tuning, Distributed database systems, Emerging research trends in database systems.

Teaching Methodology:

Lecturing, Written Assignments, Project & Research

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Presentation, Final Exam

Reference Materials:

1. Fundamentals of Database Systems, by Ramez Elmasri and Shamkant Navathe, Addison Wesley, 5th Edition, 2016.
2. Database System Concepts by Henry F. Korth and Abraham Silberschatz, 6th edition, McGraw Hill, 2015, ISBN: 0-07-12268-0

Information Technology Project Management			
Credit Hours:	3 (3,0)	**Prerequisites:**	Nill

Course Learning Outcomes (CLOs):

At the end of the course the students will be able to:

1. Understand the importance of project management certification
2. Describe the project management process groups
3. Describe the process of project integration management
4. Understand the importance of project scope management and use various techniques to manage scope
5. Understand the importance of project time management and how to use various techniques to manage time

Course Content:

Introduction to Project Management. The Project Management and Information Technology Context. The Project Management Process Groups. Project Integration Management. Project Scope Management. Project Time Management. Project Cost Management. Project Quality Management. Project Human Resource Management. Project Communications Management. Project Risk Management. Project Procurement Management. Project Management Tools.

Teaching Methodology:

Lecturing, Written Assignments, Presentation, Final Exam

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Report Writing, Presentation, Final Exam

Reference Materials:

1. Information Technology Project Management by Kathy Schwalbe, Course Technology; 7th Edition (July 22, 2014). ISBN-10: 1111221758
2. A Guide to the Project Management Body of Knowledge, 3rd Edition (PMBOK Guides), ISBN-13: 978-1930699458, 2016
3. IT Project Management: On Track from Start to Finish by Joseph Phillips, McGraw-Hill Osborne Media; 3rd Edition (February 25, 2010). ISBN-10: 0071700439
4. Information Technology Project Management by Jack T. Marche, Wiley; 4th Edition (January 6, 2014). ISBN-10: 0470371935

System and Network Administration			
Credit Hours:	4 (3,1)	**Prerequisites:**	Operating Systems

Course Learning Outcomes (CLOs):

At the end of the course the students will be able to:

1. Demonstrate essential IT support skills including installing, configuring, securing and troubleshooting operating systems and hardware.
2. Demonstrate the ability to diagnose and solve operating system and hardware problems.
3. Demonstrate essential networking skills including installing, configuring, securing and troubleshooting the devices, protocols and services within a network infrastructure.
4. Demonstrate the ability to diagnose and solve network problems.
5. Demonstrate the ability to research technology problems, provide technology support, and to learn new technology tools

Course Content:

Introduction To System Administration. SA Components. Server Environment (Microsoft and Linux). Reliable Products, Server Hardware Costing, Maintenance Contracts and Spare Parts, Maintaining Data Integrity, Client Server OS Configuration, Providing Remote Console Access. Comparative Analysis of OS: Important Attributes, Key Features, Pros and Cons. Linux Installation and Verification, Configuring Local Services and Managing Basic System Issues. Administer Users and Groups. Software Management. Managing Network Services and Network Monitoring Tools. Boot Management and Process Management. IP Tables and Filtering. Securing Network Traffic. Advanced File Systems and Logs. Bash Shell Scripting. Configuring Servers (FTP, NFS, Samba, DHCP, DNS and Apache).

Teaching Methodology:

Lecturing, Written Assignments, Presentation, Final Exam

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Report Writing, Presentation, Final Exam

Reference Materials:

1. The Practice of System and Network Administration, Second Edition by Thomas Limoncelli, Christina Hogan and Strata Chalup, Addison-Wesley Professional; 3rd Edition (2017). ISBN-10: 0321492668
2. Red Hat Enterprise Linux 6 Bible: Administering Enterprise Linux Systems by William vonHagen, 2014
3. Studyguide for Practice of System and Network Administration by Thomas A. Limoncelli, Cram101; 3rd Edition (2016). ISBN-10: 1428851755
4. Networking Systems Design and Development by Lee Chao, CRC Press; 2nd Edition (December 21, 2016). ISBN-10: 142009159X (TB2)

Virtual Systems and Services			
Credit Hours:	4 (3,1)	**Prerequisites:**	Nill

Course Learning Outcomes (CLOs):

At the end of the course the students will be able to:

1. Gain the knowledge and skills to successfully install, configure, manage, and deploy virtual servers and workstations in an organization
2. Learn how to choose the proper virtual machine product for environment
3. Learn how to partition servers to isolate applications, improve portability and migration, and create entire testing labs within a single PC.

Course Content:

This course will investigate the current state of virtualization in computing systems. Virtualization at both the hardware and software levels will be examined, with emphasis on the hypervisor configurations of systems such as Vmware, Zen and Hyper-V. The features and limitations of virtual environments will be considered, along with several case studies used to demonstrate the configuration and management of such systems. Para-virtualized software components will be analyzed and their pros and cons discussed. Processor and peripheral support for virtualization will also be examined, with a focus on emerging hardware features and the future of virtualization.

Teaching Methodology:

Lectures, Written Assignments, Semester Project, Lab Assignments, Presentations

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Project, Presentations, Final Exam

Reference Materials:

Handbook of Virtual Environments: Design, Implementation, and Applications (Human Factors and Ergonomics), Edited by Kay M Stanney, Lawrence Erlbaum Associates
Virtual Reality Technology by GRIGORE, 2015

Web Technologies			
Credit Hours:	3 (3,0)	**Prerequisites:**	Nill

Course Learning Outcomes (CLOs):

At the end of the course the students will be able to:

1. History and development of the World Wide Web and associated technologies.
2. The client-server architecture of the World Wide Web and its communication protocol HTTP/HTTPS.
3. Formats and languages used in modern web-pages: HTML, XHTML, CSS, XML, XSLT, Javascript, DOM
4. Programming web pages with Javascript/DOM (client)

Course Content:

Introduction to Web Applications, TCP/IP Application Services. Web Servers: Basic Operation, Virtual hosting, Chunked transfers, Caching support, Extensibility. SGML, HTML5, CSS3. XML Languages and Applications: Core XML, XHTML, XHTM MP. Web Service: SOAP, REST, WML, XSL. Web Services: Operations, Processing HTTP Requests, Processing HTTP Responses, Cookie Coordination, Privacy and P3P, Complex HTTP Interactions, Dynamic Content Delivery. Server Configuration. Server Security. Web Browsers Architecture and Processes. Active Browser Pages: JavaScript, DHTML, AJAX. JSON, Approaches to Web Application Development. Programing in any Scripting language. Search Technologies. Search Engine Optimization. XML Query Language, Semantic Web, Future Web Application Framework.

Teaching Methodology:

Lecturing, Written Assignments, Presentation, Final Exam

Course Assessment:

Sessional Exam, Home Assignments, Quizzes, Report Writing, Presentation, Final Exam

Reference Materials:

1. Web Application Architecture: Principles, protocols and practices by Leon Shklar and Richard Rosen, Wiley; 3rd Edition (May 5, 2018). ISBN-10:047051860X
2. Web Technologies: A Computer Science Perspective by Jeffrey C. Jackson, Prentice Hall; 2nd Edition (August 27, 2016). ISBN-10:0131856030