


Computer Science & Engineering Department, Indus Institute of Engineering & Technology, Indus University

Program - B. Tech (Computer Science & Engineering)

SEMESTER - VI													
Sr. No.		Name of the subject	Credit	Teaching Scheme				Evaluation Scheme					Segment
				(per week)				Theory		Practical		Total	
								CIE	ESE	CIE	ESE	Marks	
				Th.	Tut.	Pr.	Total (hr.)	Th.	Th.	Pr.	Pr.		
1	CS0601	Software Engineering & Project Management	4	3	0	2	5	60	40	60	40	200	Core
2	CS0602	Data Preparation & Analysis	4	3	0	2	5	60	40	60	40	200	Core
3	CE0630	Data Science	4	3	0	2	5	60	40	60	40	200	PE
	CE0631	Information Retrieval											
	CE0632	Web Data Management											
	CS0603	Virtual Reality and Augmented Reality											
4	CE0618	Advanced Java Technology	4	3	0	2	5	60	40	60	40	200	PE
	CE0619	Advance .Net Framework											
	CE0628	Mobile Application Development (Android & iOS)											
5	CE0633	Distributed Systems	3	2	0	2	4	60	40	60	40	200	PE

	CE0622	Internet of Things											
	CE0629	Data Compression											
	CS0604	Quantum Computing											
6	CE0634	Cryptography & Network Security	4	3	0	2	5	60	40	60	40	200	ES
		<i>Minor - 4</i>	3	2	0	2	4	60	40	60	40	200	
		<i>Minor - 5</i>	3	3	0	0	3	60	40	00	00	100	
		TOTAL	29	22	0	14	36	480	320	420	280	1500	

 ज्ञानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Software Engineering and Project Management								
Program: B. Tech Computer Science & Engineering				Subject Code: CS0601				Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
03	00	02	04	40	40	60	60	200

Course Objectives:

1. Be successful professionals and entrepreneur in the field with solid fundamental knowledge of software engineering and Project Management.
2. Utilize and exhibit strong communication and interpersonal skills development, as well as professional and ethical principles when functioning as members and leaders of multi-disciplinary teams.
3. Apply their foundations in software engineering to adapt to readily changing environments using the appropriate theory, principles, and processes.
4. To gain the techniques and skills on how to use modern software testing tools to support software testing projects that lead them to employability in the industry.
5. To familiarize Project Management framework and Tools Contents.

CONTENTS

UNIT-I

Introduction to Software and Software Engineering

[12 hours]

Introduction to Software Engineering-Software, Evolving role of software, characteristic of software, Three “R”-Reuse, Reengineering and Retooling, SDLC, various Software Process Models, waterfall model, spiral model, incremental model, RAD model, Agile Development

UNIT-II

Software Requirement Analysis & Specification (SRS) and basic of testing

[12 hours]

SRS, quality of good SRS, Types of Requirements, Feasibility Study, Requirement Analysis, requirement engineering task, design concepts abstraction, modularity, information hiding, functional independent cohesion, coupling, testing, types of testing, black box, white box testing. Software design concepts and Design principals, Architectural Design, Component Level

Design, Function Oriented Design and Data flow Diagram, drawing rules, Leveling of DFD
Level 0, Level 1, Level 2

UNIT-III

[12 hours]

Software Project Management

Project Management - Project planning, 4 P of management, W5HH Principle, Metrics for process and project- Product metrics, Process metrics, Project metrics, LOC, FUNCTION POINT

Project Estimation techniques- Empirical estimation techniques, Heuristic techniques, COCOMO model, Project scheduling- methods, Work breakdown structure, Task network analysis (ACTIVITY NETWORK), Gantt chart, PERT, CPM

UNIT-IV

[12 hours]

Management

Risk Management – software risk, risk identification, risk projection, risk refinement, risk mitigation and monitoring,

Quality Management-quality control vs quality assurance, software review, types of review, reliability, and availability.

Change management-configuration management, change control, version control.

Course Outcomes:

At the end of this subject, students should be able to:

1. Apply use of knowledge of Software Life Cycle to successfully implement the projects in the corporate world.
2. Implement Project Management Processes to successfully complete project in IT industry.
3. Apply the concept of Functional Oriented and Object-Oriented Approach for Software Design.
4. Recognize how to ensure the quality of software product, different quality standards and software review techniques.
5. Identify the Inputs, Tools, and techniques to get the required Project deliverable and Product deliverable using Knowledge areas of Project Management and Apply various testing techniques and test plans and
6. Will be Able to understand modern Agile Development and Service Oriented Architecture Concept of Industry.

Textbooks:

1. Roger S. Pressman, Software engineering- A practitioner's Approach, McGraw-Hill International Editions.

Reference Books:

1. Managing Information Technology Project, 6edition, by Kathy Schwalbe, Cengage Learning publication.
2. Information Technology Project Management by Jack T Marchewka Wiley India publication.
3. Software Engineering 3rd edition by KK Agrawal, Yogesh Singh, New Age International publication.
4. Software Engineering Project Management by Richard H. Thayer Wiley India Publication.
5. Software Engineering for students: A Programming Approach by Douglas Bell, Pearson publication.


Web Resources:

1. <https://nptel.ac.in/downloads/106105087/>
2. <https://nptel.ac.in/courses/106101061/>

LIST OF EXPERIMENTS

Experi ment. No.	Title	Learning Outcomes
1	Study of different software engineering models. Waterfall model, Prototype model, Incremental model, Spiral model and RAD Models, Scrum Model. Select any application for which software development process can be defined and find out which process model will be more suitable for application.	CO1,CO5
2	Prepare Software Requirement Specification (SRS) document for chosen application	CO1,CO2
3	Draw E-R diagram and class diagram for chosen application	CO3

4	Write Use case Scenario and draw Use Case Diagram for chosen application.	CO3
5	Draw Activity diagram for selected application.	CO3
6	Draw sequence diagram for selected application.	CO3
7	Draw Data Flow Diagram (DFD) for selected application	CO3
8	Apply FP oriented estimation model for selected application.	CO4
9	Study of various software testing methods and design test cases for selected application	CO5
10	Study of any two Open source tools in DevOps for Infrastructure Automation, Configuration Management, Deployment Automation, Performance Management, Log Management and Monitoring	CO6

 ज्ञानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Data Preparation & Analysis								
Program: B. Tech CS				Subject Code: CS0602				Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
3	0	0	4	40	0	60	0	100

Course Objectives:

1. To learn how to gather and analyze large sets of data to gain useful business understanding and how to produce a quantitative analysis report/memo with the necessary information to make decisions.
2. To survey industrial and scientific applications of Data Analytics, with case studies to develop entrepreneurship skill.
3. To prepare data for analytics and perform exploratory data analysis.
4. To develop meaningful data visualizations to empower skills of students in data analytics field.
5. To perform cleaning and reformatting real world data for analysis.
6. To apply summary statistics techniques over datasets.

CONTENTS

UNIT-I

[12 hours]

Introduction to Data Analysis

Defining data analysis problems: Knowing the client, Understanding the questions; Data Gathering and Preparation: Data formats, parsing and transformation, Scalability and real-time issues

UNIT-II

[12 hours]

Exploratory Analysis

Data Cleaning: Consistency Checking, Heterogeneous and Missing data, Data Transformation and Segmentation; Exploratory Analysis: Descriptive and Comparative analysis, Clustering and Association, Hypothesis Generation

UNIT-III

[12 hours]

Visualization

Designing Visualizations, Time Series, Geolocated Data, Correlations and Connections, Hierarchies and Networks, Interactivity

UNIT-IV

[12 hours]

Ethics in the Profession

Cases in Computing, Statistics and Communication, Professional ethics codes: ACM, IEEE, AM Stat. Assoc.

Course Outcomes:

At the end of this subject, students should be able to:


1. Apply clean and format real time data pertaining to real time data science applications
2. Visualize data in multiple dimensions as per the application requirement
3. Draw a comparative analysis of the different format of data
4. Analyze descriptive data with different technique.
5. Obtain results by applying statistic techniques over datasets
6. Students able to understand ethics profession of different codes.

Text Books:

1. Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining, by Glenn J. Myatt

Reference Books:

1. Data Preparation for Data Mining by Dorian Pyle – Morgan Kaufmann Series
2. Data Preparation in the Big Data by Federico Castanedo – O'Reilly
3. Data Analytics by Anil Maheshwari – McGraw Hill

 ज्ञानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Data Science								
Program: B. Tech CE/CSE/IT				Subject Code: CE0630			Semester: VI	
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
3	0	2	4	40	40	60	60	200

Course Outcome:

1. Learn the fundamentals of data analytics and the data science pipeline
2. Learn how to scope the resources required for a data science project
3. Apply principles of Data Science to the analysis of business problems.
4. Skill development in data mining software to solve real-world problems.
5. Increase in employability in cutting edge tools and technologies to analyze Big Data.

CONTENTS

UNIT-I

[12 Hours]

Introduction to data science:

Defining Data Science, what do data science people do? Data Science in Business, Use Cases for Data Science, Data science and Big data, Data science and Machine learning
 Data Science Process Overview – Defining goals – Retrieving data – Data preparation – Data exploration – Data modeling – Presentation.

UNIT-II

[12 Hours]

Introduction to statistics:

What is statistics, Descriptive Statistics: Introduction, Population and sample, Types of variables, Measures of central tendency, Measures of variability, Coefficient of variance, Skewness and Kurtosis

Inferential Statistics:

Normal distribution, Test hypotheses, Central limit theorem, Confidence interval, T-test, Type I and II errors

UNIT-III

[12 Hours]

Machine Learning Introduction and Concepts:

Machine learning – Modeling Process – Training model – Validating model – Predicting new observations

Important machine learning terminologies, Types of machine learning algorithms, Supervised learning algorithms: Types of supervised learning algorithms, Regression: Linear Regression, Classification algorithms

Unsupervised learning algorithms: Clustering algorithms

UNIT-IV

[12 Hours]

Introduction to data visualization – Data visualization options – Filters – Python libraries for visualization – Matplotlib- seaborn

Data Science Ethics – Doing good data science – Owners of the data - Valuing different aspects of privacy - Getting informed consent - The Five Cs – Diversity – Inclusion – Future Trends.

Course Outcome:

After completion of the course students will be able to:

- 1) Demonstrate knowledge of big data analytics.
- 2) Demonstrate the ability to think critically in making decisions based on data
- 3) Interpret data, extract meaningful information, and assess findings.
- 4) Identify and analyze social, legal, and ethical issues in data science.
- 5) Choose and apply tools and methodologies to solve data science tasks.
- 6) Explore future trends in data.

Text Books:

1. Introducing Data Science, Davy Cielen, Arno D. B. Meysman, Mohamed Ali, Manning Publications Co., 1st edition, 2016
2. An Introduction to Statistical Learning: with Applications in R, Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, Springer, 1st edition, 2013
3. Ethics and Data Science, D J Patil, Hilary Mason, Mike Loukides, O' Reilly, 1st edition, 2018


Reference Books:

1. Machine Learning: A Probabilistic Perspective. Kevin P. Murphy.

LIST OF EXPERIMENTS

Sr. No.	Title	Learning Outcome
1	Getting Started with Skills Network Labs	To know functionality and usage of Skill Network Labs environment
2	Getting Started with Jupyter Notebooks	To know functionality and usage of Jupyter Notebook platform
3	Getting Started with Apache Zeppelin Notebooks	To know functionality and usage of Apache Zeppelin Notebook
4	Getting Started with RStudio IDE	Introduction to Rstudio and its usage in Machine Learning
5	Data Analysis with Python Import data sets Clean and prepare data for analysis Manipulate pandas Data Frame Summarize data Build machine learning models using scikit-learn Build data pipelines	To understand the concept of machine learning, data preparation, pandas and scikit-learn with model building.

6	Data Visualization with Python Introduction to Visualization Tools Basic Visualization Tools Specialized Visualization Tools Creating Maps and Visualizing Geospatial Data	To understand about the field of data visualization and tools used for visualization.
7	Advanced Visualization Tools	Study and understanding about functionalities of advanced visualization tools.

 <div>ज्ञानेन प्रकाशते जगत्</div> <div>INDUS UNIVERSITY</div>				INDUS INSTITUTE OF TECHNOLOGY& ENGINEERING				
Constituent Institute of Indus University								
Subject: Information Retrieval								
Program: B. Tech CE/CSE/IT				Subject Code: CE0631			Semester: VI	
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
3	0	2	4	40	40	60	60	200

Course Objective:

1. To develop a model for Text Indexing and Retrieval
2. To evaluate Information Retrieval Systems
3. To analyze Textual and Semi-Structured data sets
4. To learn Text Similarity Check Measures
5. To Understand Search Engines

CONTENTS

UNIT-I

Introduction to Text Retrieval Systems

[06 hours]

Retrieval System, Types of Retrieval Systems, Boolean Retrieval, Term Vocabulary Lists, Posting Lists, Index Compression

UNIT-II

Information Retrieval Models

[16 hours]

Vector Space Model, TF-IDF Weight Model, Evaluation of an Information Retrieval System, Okapi/BM25, Language Models, KL-Divergence, Page Ranking

UNIT-III

Query Expansion and Feedback

[06 hours]

Query Expansion, Relevance Feedback, Pseudo Relevance Feedback, Query Reformation

UNIT-IV

Text Classification and Clustering

[20 hours]

Text Classification Problem, Bayesian Classification, Support Vector Machine, Feature Selection, Flat Clustering, K-Means Clustering

Course Outcomes:

At the end of this subject, students should be able to:

- 1) Recognize Document as Vector and perform Text Analysis over the document
- 2) Perform Evaluation of the Information Retrieval Systems
- 3) Understand Search Engines and Page Ranking Techniques
- 4) Perform Supervised and Unsupervised Learning Techniques
- 5) Understand Relevance Feedback Mechanism in Search Optimization
- 6) Apply Text Classification & Clustering techniques

Text Books:

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

Reference Books:


1. Cheng Xiang Zhai, Statistical Language Models for Information Retrieval (Synthesis Lectures Series on Human Language Technologies), Morgan & Claypool Publishers, 2008.

Web Resources:

- 1) <https://nlp.stanford.edu/IR-book/newslides.html>
- 2) <https://www.tutorialspoint.com/lucene/>

LIST OF EXPERIMENTS

Sr. No.	Title	Learning Outcomes
1	Introduction to Lucene Software	To <i>develop</i> a model for Text Indexing and Retrieval
2	Implementation of Basic Text Processing Techniques on a Text Document	To <i>analyze</i> Textual and Semi-Structured data sets, To <i>learn</i> Text Similarity Check Measures, To <i>evaluate</i> Information Retrieval Systems
3	Query Expansion and Ranking in Lucene	To <i>analyze</i> Textual and Semi-Structured data sets, To <i>learn</i> Text Similarity Check Measures, To <i>evaluate</i> Information Retrieval Systems
4	Implementation of Language Model with Smoothing Techniques	To <i>analyze</i> Textual and Semi-Structured data sets, To <i>learn</i> Text Similarity Check Measures, To <i>evaluate</i> Information Retrieval Systems
5	Introduction to RStudio	To <i>develop</i> a model for Text Indexing and Retrieval
6	Implementation of Naïve Bayes Classifier in RStudio	To <i>apply</i> techniques for Text Classification and Clustering
7	Implementation of K-Means Algorithm in RStudio	To <i>apply</i> techniques for Text Classification and Clustering
8	Case Study: Search Engine	To <i>Understand</i> Search Engines

 इजानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Web Data Management								
Program: B. Tech CE/CSE/IT					Subject Code: CE0632			Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)-Theory	Continuous Internal Evaluation (CIE)-Practical	Total
3	0	2	4	40	40	60	60	200

Course Objective:

1. Understand the basic concept of web data management using XML.
2. To learn Xpath and XQuery to find appropriate data from XML.
3. To learn the current technology stack (URIs, XML, RDF/S, OWL) in web data management.
4. To understand test on an ontology and schema mappings
5. To learn different tools for semantic data management

CONTENTS

UNIT-I

[12 hours]

Data Model

Introduction to Modeling Web Data, Semistructured data, XML, Web Data Management with XML, XML Standards, XML and syntax, XML Data Model, XLink, and XPointer.

XPath and XQuery

Introduction, Basics of XPath and XQuery, XPath: Steps and path expressions, Evaluation of path expressions, Generalities on axes and node tests, Axes, Node tests and abbreviations, Predicates, XPath 2.0; FLWOR expressions in XQuery: Defining variables - the for and let clauses, Filtering - the where clause, The return clause, Advanced features of XQuery; XPath foundations.

UNIT-II

[12 hours]

Typing

Motivating Typing, Automata, Schema Languages for XML, Typing Graph Data: Graph Semi structured Data, Graph Bisimulation, Data guides.

XML Query Evaluation

XML fragmentation, XML identifiers: Region-based identifiers, Dewey-based identifiers, Structural identifiers and updates; XML evaluation techniques: Structural join, Optimizing structural join queries, Holistic twig joins

UNIT-III

[12 hours]

Ontologies, RDF, and OWL

Introduction, Ontologies by example, Web resources, URI, namespaces, RDF, RDFS: RDF Schema, OWL, Ontologies and (Description) Logics.

Querying Data through Ontologies.

Introduction, Querying RDF data: notation and semantics, Querying through RDFS ontologies, Answering queries through DL-LITE ontologies

UNIT-IV

[12 hours]

Data Integration

Introduction, Containment of conjunctive queries, Global-as-view mediation, Local-as-view mediation, Ontology-based mediators, Peer-to-Peer Data Management Systems.

Building Web scale applications

Web search, web crawlers, web information retrieval, Web graph mining and hot topics in web search, Distributed systems, failure management, required properties of a distributed system, P2P networks, Hash-based structures, distributed indexing, Distributed computing with MapReduce.

Course Outcomes:

After successful completion of this course, student will be able to

- 1) Learn the overall vision of the Semantic Web
- 2) Analyzed the current technology stack and design for future (URIs, XML, RDF/S, OWL).
- 3) To understand how one could use these technologies for building something useful.
- 4) Design test an ontology and define schema mappings.
- 5) Use different tools for semantic data management
- 6) Understand the concept of MapReduce technique in distributed computing.

Text Books:

- 1) Bhavani Thuraisingham, “Web Data Management and Electronic Commerce”, CRC Press, 2000
- 2) Bhavani Thuraisingham, “XML Databases and the Semantic Web”, CRC Press, 2002

Reference Books:

- 1) Serge Abiteboul, Ioana Manolescu, Philippe Rigaux, Marie-Christine Rousset and Pierre Senellart, “Web Data Management”, Cambridge University Press, 2011
- 2) Athena Vakali and George Pallis, “Web Data Management Practices: Emerging Techniques and Technologies”, IGI Publishing, 2007, ISBN-10: 1599042282; ISBN-13: 978-1599042282


Web Resources

- 1) <https://nptel.ac.in/courses/111104100/16>
- 2) <https://www.coursera.org/learn/data-management-cloud>
- 3) <https://www.classcentral.com/tag/data-management>

LIST OF EXPERIMENTS

Sr. No	Title	Learning Outcome
1	Create an XML file defining an article in newspaper.	Create an XML File
2	Create an XML file containing list of students. Also create stylesheet file to display list in an HTML format.	Create Stylesheet
3	Create an XML file containing list of students. Using XPath display following information <ul style="list-style-type: none">• Information of a student with ID No : 101• All the student in the sorted order according to their CGPA	Using XPath display
4	Create an XForm to collect information from staff member regarding their publications. Details like Year of Publication, National/International, Title, Conference/Journal etc.	Using XForm to collect information
5	From the above gathered information, using XQuery find out the number of publication in a specific year.	Using XQuery
6	Demonstrate the use of AJAX.	Usage of AJAX
7	Study of XMLSPY tool.	Study of XMLSPY

8	Create an RSS for the events occurring in your institute	Create an RSS
9	Write a program to read the articles in RSS created in above practical.	Read article in RSS
10	Study of RDF (Resource Description Framework)	Study of RDF
11	Create a system which can use of Web search, web crawlers and web information retrieval.	To create a system for information retrieval
12	Analyze and implement a system with Web graph mining.	Implement system for Web graph mining
13	Implement and Subscribe RSS News feeds to get latest news in India	Implement and subscribe RSS News Feeds

 इजानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Virtual Reality and Augmented Reality								
Program: B.Tech CSE				Subject Code: CS0603				Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
3	0	2	4	40	40	60	60	200

Prerequisite: NA

Course Objectives:

1. This course is designed to give historical and modern overviews and perspectives on virtual reality.
2. It describes the fundamentals of sensation, perception, technical and engineering aspects of virtual reality systems.
3. Explain the concepts of motion and tracking in VR systems.
4. Describe the importance of interaction and audio in VR systems

Unit 1:

Defining Virtual Reality:

[12 hours]

Introduction to Virtual Reality: Virtual Reality and Virtual Environment: Introduction, Computer graphics, Real time computer graphics, Flight Simulation, Virtual environment requirement, benefits of virtual reality, Historical development of VR, Scientific Landmark.

Computer Graphics and Geometric Modelling

Computer Graphics and Geometric Modelling: Introduction, The Virtual world space, positioning the virtual observer, the perspective projection, human vision, stereo perspective projection, Colour theory, Conversion From 2D to 3D, 3D space curves, 3D boundary representation, Simple 3D modelling, 3D clipping, Illumination models, Reflection models, Shading algorithms. Geometrical Transformations: Introduction, Frames of reference, Modelling

transformations, Instances, Picking, Flying, Scaling the VE, Collision detection.

Unit 2:

Virtual Environment:

[12 hours]

Virtual Environment: Input: Tracker, Sensor, Digital Gloves, Movement Capture, Video-based Input, 3D Menus & 3D Scanner etc. Output: Visual /Auditory / Haptic Devices. Generic VR system: Introduction, Virtual environment, Computer environment, VR technology, Model of interaction, VR Systems. Animating the Virtual Environment: Introduction, The dynamics of numbers, Linear and Nonlinear interpolation, the animation of objects, linear and non-linear translation, shape & object in between, free from deformation, particle system. Physical Simulation: Introduction, Objects falling in a gravitational field, Rotating wheels, Elastic collisions, projectiles, simple pendulum, springs, Flight dynamics of an aircraft

Unit 3:

Visual Perception

[12 hours]

Perception of Depth, Perception of Motion, Perception of Color, Combining Sources of Information Visual Rendering -Ray Tracing and Shading Models, Rasterization, Correcting Optical Distortions, Improving Latency and Frame Rates.

Unit 4:

Augmented Reality

[12 hours]

Augmented Reality: Taxonomy, technology and features of augmented reality, difference between AR and VR, Challenges with AR, AR systems and functionality, Augmented reality methods, visualization techniques for augmented reality, enhancing interactivity in AR environments, evaluating AR systems

Development Tools and Frameworks:

Human factors: Introduction, the eye, the ear, the somatic senses. Hardware: Introduction, sensor hardware, Head-coupled displays, Acoustic hardware, Integrated VR systems. Software: Introduction, Modelling virtual world, Physical simulation, VR toolkits, Introduction to VRML. AR/VR applications: Introduction, Engineering, Entertainment, Science, Training.

Course Outcomes: After completion of this course, student will be able to:

1. To understand fundamental computer vision, computer graphics and human-computer interaction techniques related to VR/AR. (Understand)
2. To understand geometric modeling and Virtual environment. (Understand)
3. To relate and differentiate VR/AR technology. (Analyse)
4. To use various types of Hardware and software in virtual Reality systems (Apply)

5. To implement Virtual/Augmented Reality applications (Apply)

Text Books:


1. Virtual Reality, Steven M. LaValle, Cambridge University Press, 2016
2. Understanding Virtual Reality: Interface, Application and Design, William R Sherman and Alan B Craig, (The Morgan Kaufmann Series in Computer Graphics)”. Morgan Kaufmann Publishers, San Francisco, CA, 2002
3. Developing Virtual Reality Applications: Foundations of Effective Design, Alan B Craig, William R Sherman and Jeffrey D Will, Morgan Kaufmann, 2009.

List of Experiment

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy need to address the average students and inclusive of an element to attract and promote the intelligent students. Use of open-source software is encouraged. Based on the concepts learned. Instructor may also set one assignment or mini-project that is suitable to respective branch beyond the scope of syllabus. Operating System recommended: - 64-bit Windows OS Programming tools recommended: - Unity, C#, Blender, VRTK. VR Devices: HTC Vive, Google Cardboard, Google Daydream and Samsung gear VR.

<u>Experiment No.</u>	<u>Experiment Title</u>	<u>Course Outcome</u>
1	Installation of Unity and Visual Studio, setting up Unity for VR development, understanding documentation of the same.	To understand fundamental computer vision, computer graphics and human-computer interaction techniques related to VR/AR
2	Demonstration of the working of HTC Vive, Google Cardboard, Google Daydream and Samsung gear VR	To understand geometric modeling and Virtual environment
3	Develop a scene in Unity that includes: i. a cube, plane and sphere, apply transformations on the 3 game objects. ii. add a video and audio source.	To relate and differentiate VR/AR technology
4	Develop a scene in Unity that includes a cube, plane and sphere. Create a new material and texture separately for three	To relate and differentiate VR/AR technology

	Game objects. Change the colour, material and texture of each Game object separately in the scene. Write a C# program in visual studio to change the colour and material/texture of the game objects dynamically on button click	
5	Create an immersive environment (living room/ battlefield/ tennis court) with only static game objects. 3D game objects can be created using Blender or use available 3D models.	To use various types of Hardware and software in virtual Reality systems
6	Include animation and interaction in the immersive environment in the pervious experiments.	To use various types of Hardware and software in virtual Reality systems
7	Create a virtual environment for any use case. The application must include at least 4 scenes which can be changed dynamically, a good UI, animation and interaction with game objects. (e.g VR application to visit a zoo)	To implement Virtual/Augmented Reality applications
8	Develop a simple UI(User interface) menu with images, canvas, sprites and button. Write a C# program to interact with UI menu through VR trigger button such that on each successful trigger interaction display a score on scene .	To implement Virtual/Augmented Reality applications

 इजानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Advance Java Technology								
Program: B. Tech CE/CS/IT					Subject Code: CE0618			Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
3	0	2	4	40	40	60	60	200

Course Objectives:

1. To provide and enrich students with knowledge of Enterprise Java standards and architectures.
2. To introduce the concepts and techniques of problem solving through structured Modular approaches.
3. To implement application over the web or network.
4. To cultivate good programming style and discipline. To improve the team work qualities.
5. To understand concept and application of spring and struts framework.
6. By learning the skills of Advance Java Technology students get more employable in the field of Software Development

CONTENTS

UNIT-I

[12 hours]

Introduction to J2EE and Overview of Web Development:

Concept of Java Technology, J2EE Architectures, Java EE Components and Containers, Types of Servers in J2EE Application. Concept of HTTP Protocols and API, Request Processing in Web Application, Web Application Structure, Web Containers, and Web based MVC architecture.

JDBC:

Introduction to JDBC, Architecture of JDBC, JDBC driver types, steps for connecting to JDBC, Types of Statements in JDBC (Statement Interface, Prepared Statement, Callable Statement),

Types of Result Set, Executing SQL Queries, Meta Data, JDBC Exception, Transaction Management.

UNIT-II

[12 hours]

Servlet API:

Introduction to Servlet, Life Cycle of Servlet, HTTP Methods Structure and Deployment descriptor, ServletContext and ServletConfig Object, Request and Response objects, ServletCollaboration, Servlet Annotations, Session Tracking, Filters API, Connecting Servlet API to JDBC.

JSP:

Introduction to JSP, Compare JSP with Servlet, JSP page life cycle, JSP architecture, JSP elements, JSP Implicit Objects, Expression Language, JSP Standard Tag Libraries, JSP Custom Tag, JSP Session Management, JSP Exception Handling, JSTL

UNIT-III

[12 hours]

Network Programming:

Basic of Network Programming, Introduction of Socket, Types of Socket, Socket API, TCP/IP client sockets, java.net package Socket, Datagram's, URL, TCP/IP server sockets, RMI Architecture, Client Server application using RMI.

Java Mail:

Introduction, Protocols used in Java Mail, Architecture of Java Mail, Sending and Receiving Email.

Hibernate Framework:

Introduction, Architecture, Object Relational Mapping in Hibernate, Hibernate annotations, Hibernate Query Language.

UNIT-IV

[12 hours]

Spring Framework:

Introduction, Spring Architecture, Spring MVC Module, Bean Life Cycle, Constructor Injection, Dependency Injection, Inner Beans, Aliases in Bean, Bean Scopes, Spring Annotation. Spring AOP Module, Spring DAO, Database Transaction Management, CRUD Operation using DAO and Spring API

Struts Framework:

Introduction, features, core component, Struts2 Architecture, Action, Struts2 Configuration, Interceptors, Struts2 Validation (Custom & Bundled Validation)

Course Outcome:

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

1. Demonstrate the working of web applications and able to work in web development environment
2. Understanding various java frameworks and its applications, so student can able to decide future way in interested framework
3. Develop both client side and server side programming.
4. Graphical user interface in Java programs and able to work in GUI design requirement in industry.
5. Know MVC Architecture concept in practical .so they able to design secure application.
6. Different classes for database framework and OR Mapping and able to design dynamic application with database

Text Books:

1. Java 6 Programming, Black Book, Dreamtech
2. Java Server Programming, Java EE6 (J2EE 1.6), Black Book, Dreamtech
3. Java Programming Advance Topics, Joe Wigglesworth and Paula McMillan, Cengage Learning.

Reference Books:

1. Professional Java Server Programming, a Apress
2. Core Java, Volume II – Advanced Features, Eight Edition, Pearson
3. Unleashed Java 2 Platform, Sams Techmedia
4. The Complete Reference J2EE, Keogh, McGrawHill
5. Java EE 5 for beginners, Bayross and Shah, SPD
6. JDBC 3 Java Database Connectivity, Bernand Van Haecke, Wiley-dreamtech
7. Java Server Pages for Beginners, Bayross and Shah, SPD
8. Java Servlet Programming, Jason Hunter, SPD (O'Reilly)


Web Resources:

1. <https://www.javatpoint.com/what-is-framework-in-java>
2. <https://www.javatpoint.com/java-swing>
3. <https://www.tutorialspoint.com/spring/index.htm>
4. <https://www.javatpoint.com/spring-tutorial>

LIST OF EXPERIMENTS

Sr. No.	Class Activity	List of Practical	Learning Outcomes
01	Lab 1	Web server, Introduction to JSP (print hello word), Dynamic table printing using JSP (use color change)	Students know Basic Concept of JSP.
02	Lab 2	Data forwarding and manual URL building, Learning JSP page directive (any one)	Student know concept of URL building through JSP.
03	Lab 3	CRUD operation using Cookie, Cookie management tool (Cookie table with delete and update button)	Students can able store data in cookies in Java application.
04	Lab 4	Demonstrating five methods of session, Shopping cart using session	Student can able to understand practical concept of session.
05	Lab 5	Login using session (static username & password), Create Registration form using type-1 database connectivity	Student can understand login concept of any web application.
06	Lab 6	List all the records of a table using type-2 db connectivity driver, Search engine using type-2 database connectivity	Student can understand database connectivity with java application
07	Lab 7	Introduction to Servlet (folder structure-generic servlet-http servlet-lifecycle-deployment descriptor file)	Student can understand servlet life cycle by practical
08	Lab 8	Create a browser specific page hit counter using servlet, Using single HTTP Servlet create a form and insert a record into database..	Student can create servlet application.

09	Lab 9	Parameter passing in servlet, Create an application specific page hit counter in which the counter value should remain as it is even though we restart the server	Student can able to create servlet application
10	Lab 10	Demonstrate web base MVC architecture using JSP, Servlet and JavaBean, Introduction to swing (blank frame-FlowLayout-button)	Student understand concept MVC architecture
11	Lab 11	Pluggable look and feel using JButton, Swing + Database	Student can understand concept of swing and able to design GUI Application
12	Lab 12	Learning TCP connection using java, Learning UDP connection using JAVA	Student can understand concept of network programming
13	Lab 13	Demonstrating the use and architecture of RMI app	Understand concept of RMI
14	Lab 14	Implementing internationalization in JAVA, Learning hibernate configuration and architecture	Understand concept of hibernate.
15	Lab 15	Insert, delete, update n select records using hibernate, Hibernate annotations	Understand database connectivity in hibernate.

 ज्ञानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Advanced .NET Framework								
Program: B. Tech CE/CS/IT					Subject Code: CE0619			Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
3	0	2	4	40	40	60	60	200

Course Objective:

1. Understand Basic architecture of ASP.NET and how to provide the connectivity.
2. Ideas for basic namespaces, Object oriented Concepts and database connection.
3. Understand concept of master pages and themes for any application.
4. Learn to use of web service for web application.
5. Identify debug issues related to the development of concurrent programs.
6. Learn the skills of web development which makes them industry ready.

CONTENTS

UNIT-I

[12 hours]

C# .Net : Namespaces, Constructor and Destructor- Using parameterized constructor in class, creating copy constructor, calling a destructor in class, Function Overloading and inheritance- Function overloading, Inheritance, Modifiers, Properties and indexers-creating read and write property, creating read only property, creating a static property, Attributes, Reflection API- Events and Delegates.

ASP.Net: Introduction to ASP.NET Framework, Difference between ASP and ASP .Net, lifecycle of an Asp.Net Web page- difference stages of asp.net page, ASP.Net page life cycle, Working with Asp.net pages, Exploring the concepts of code behind page model, Explain View state, Working with post back, Working with Controls, Working with validation control, Using Rich server control, Working in user control in asp.net.

UNIT-II

[12 hours]

Managing State and Master Pages and Themes

Managing State: Preserving state in web application, Page level state, using cookies to preserve state, ASP.Net session state, Storing objects in session state, configuring session state, storing session state in sql server, using cookie less session IDs, Application state.

Master Pages and Themes: Master Pages-Simple and Nested Master pages, Working with content Placeholder and Nested master Page-Creating simple master page, Creating nested master page.Asp.Net Themes: Working with CSS and Skin files.

UNIT-III

[12 hours]

Data Bound Controls and Repeater Controls: Overview of List Bound Controls- List box, Dropdown list, Checkbox list, Radio Buttonlist, Gridview, Listview, Creating Repeater control, Creating datalist control

Introduction to ADO.Net: Benefits of ADO.Net, ADO.Net compared to classic ADO, ADO.Net Objects and Namespaces, Managed Providers, Disconnected Data Access- Disconnected Data Access with IDE, Disconnected Data Access without IDE, Connected Data Access, Data Binding- Manual Data binding, simple data binding, complex data binding ,Types Dataset.

UNIT-IV

[12 hours]

Creating and Consuming Web and Advance .Net Concepts.

Creating and Consuming Web: The motivation of XML web services, Designing of XML Web services, Creating XML Web services with visual studio, Creating Web service consumer, Discovering web service using UDDI.

Advance .Net Concepts: Introduction of WPF, Introduction of WCF, Exploring Silverlight, and Introduction of AJAX

Course Outcome:

After learning, the course the students should be able to:

- 1) Use .NET framework architecture, various tools, data bound control and Validation techniques for dynamic application.
- 2) Use of different templates available in Visual Studio for asp.net application
- 3) Implementation testing strategies in real time applications.
- 4) Design and develop complex concurrent programs using the .NET framework
- 5) Use advanced concepts related to Web Services, WCF, and WPF in project development.
- 6) Develop a website after learning all the concepts, knowledge gained from above subject could be further implemented into advanced level projects in coming semesters.

Text Books:

- 1) ASP.NET 2.0, Black Book, Dreamtech
- 2) ASP .NET Complete Reference, TMH
- 3) C# the Basic by Vijay Mukhi

Reference Books:

- 1) Advance .Net Technology, Dreamtech Edition, Chirag Patel
- 2) ASP.NET Developer's Guide – Greg Bucek, McGraw Hill
- 3) Programming VB.Net 2005, Julia Case Bradley, Anita Millspaugh, McGraw Hill


Web Resources

- 1) <https://www.tutorialsteacher.com/mvc/asp.net-mvc-tutorials>
- 2) <https://www.tutorialspoint.com/asp.net/>
- 3) <https://dotnet.microsoft.com/learn/dotnet/architecture-guides>
- 4) <https://www.javatpoint.com/asp-net-tutorial>

LIST OF EXPERIMENTS

Experiment. No.	Title	Learning Outcome
1	Create a website to pass username and password from one page to another within a query string. and Check if its valid or invalid user	Username and Password creation
2	Create a website applying an external skin file to a given webpage.	Apply external skin file in webpage
3	Create a webpage using checkbox, checkbox list, and radio button.	Apply checkbox, checkbox list, and radio button.
4	Create a webpage performing Image upload, Adrotator as well as a Date & Time display Calender	Use Image upload, adorator and Calendar
5	Create a website showing the usage of a Menustrip within a webpage	To understand Menu strip usage
6	Create a website performing Multiview output within a webpage	To use feature of multiview output
7	Create a website performing validation within a form.	To apply concept of validation in form.
8	Create a website showing implementation of master pages	To implement master pages in website
9	Create a website in which Multiview is used.	To use the multiview feature in website

10	Create a website demonstrating State management within a webpage.	To create website for state management
11	Create a website implementing the usage of web services from external sources within a web page.	To implement usage of web services
12	Demonstrate connected-disconnected state of a database.	Demonstration of both states
13	Create a webpage demonstrating structure of MDI form.	Demonstrate structure of MDI form
14	Create a webpage demonstrating structure of Registration Page	Demonstrate structure of Registration Page
15	Creating Web services for addition of two numbers	To create web services

 इजानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Mobile Application Development (Android & IOS)								
Program: B. Tech CE/CS/IT					Subject Code: CE0628			Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
3	0	2	4	40	40	60	60	200

Course Objectives:

1. Design and develop the basic application programs.
2. Describe, identify and debug issues related to the development of application.
3. Create a customized control application with different UI components which helps to develop skills in mobile application development.
4. Design and develop the database needed for the storing data of application.
5. Understand the different states of mobile application

CONTENTS

UNIT-I

[12 hours]

The Basics:

Hello World: Intro to Android, Why develop app for Android?, Flavors of Android operating systems, Challenges of developing for Android (multiple OS, need backwards compatibility, need to consider performance and offline capability)

Concept: Create Your First Android App: Overview of the development process - Java, Android Studio, Project layout in Android Studio, Target and minimum SDKs, Android Virtual Device (AVD) Monitor, Viewing logs in logcat and AVD, Android manifest file, App Architecture: An app consists of one or more activities. For an activity, write Java code and layout xml, and hook them together, and register the activity in the manifest file.

Concept: Layouts, Views and Resources: Layout elements can be viewed and edited in Layout Editor and XML, Introduction to the range of UI elements, Resources (layouts, strings, styles, themes), Identifying resources with IDs, Programmatically referencing, resources using resource IDs, on Click attribute, Getting user input from a view, Programmatically changing UI elements, Layout Managers, Defining layouts for activities, inflating the layout

Concept: Scrolling Views: How to make activities scrollable: compare ScrollView, ListView, RecyclerView, Getting the resource ID for a UI element by inflating a layout (needed for RecyclerView), How to implement RecyclerView (requires layout managers and ViewHolders) , Performance implications of different kinds of scrolling UI elements

Concept: Resources to Help You Learn: Resources to help you learn: Samples that ship with the SDK, Templates for projects, developer.android.com, Android developer blog , Android developer YouTube channel, Source code and samples in github, Stack overflow, Google search!

Activities and Intents : About activities, Defining Activities , Activity Lifecycle , Activity navigation , About intents ,Explicit vs Implicit intents ,Passing info to new activity ,Returning data from activity

The Activity Lifecycle and Managing State: Activity lifecycle , Activity lifecycle callback methods , Activity instance state.

Starting Activities with Implicit Intents: Starting activities by sending implicit intents, Intent filters and enabling your activities to receive intents, ShareCompat.

Testing and Debugging, and Backwards Compatibility: Debugging your apps, Testing your app, Support libraries

UNIT-II

[12 hours]

User Interface:

User Input Controls: Getting user input , Changing keyboards , Buttons , Dialogs and pickers , Spinners, checkboxes, and radio buttons

Menus: Options menu, contextual menus (floating and action bar), and popup menu, Adding menu items. Handling on Clicks from menus.

Screen Navigation: Terminology, Different ways a user can navigate through an app, Action bar, Settings menu, Navigation drawer, Directed workflow (funnels), Best practices for navigation

Themes and Styles: Best practices for themes and styles, Performance benefits for themes, When and how to use drawables, best practices for drawables, When and how to use nine-patches, best practices for nine-patches, Tools for creating drawables.

Material Design: What is material design? Material design best practices. Material Design guidelines, Implementing Material Design look and feel, with compatibility with previous versions, Support library for Material Design , Transitions and Animations

Adapt layouts for multiple devices and orientations: Why we need to consider different screen sizes and orientations , Screen density (dip or dp), How to create adaptive layouts using resources folders , Different ways to create images that scale nicely, Images and image formats and how they affect performance (download speeds).

Accessibility: Why accessibility matters, Accessibility considerations: Color blindness, poor vision, poor hearing, physical limitations, Accessibility guidelines, testing for accessibility, Screen readers, making your app more accessible: Color and Contrast, button size Material Design guidelines, considerate layouts and navigation

Localization: How to prep your app for localization, LTR and RTL (eg Arabic) text.

Testing the User Interface: Automated testing of UIs, User testing your UI with real users, Using the Espresso and UI Automator frameworks for testing UIs

UNIT-III

[12 hours]

Background Tasks:

Connect to the Internet: Background Tasks

Synchronous versus async tasks, what is the UI thread and when should you use it? , Example of a background task - retrieving data over the internet, Creating background tasks. (Schedule, send data, etc.) , Implementing AsyncTask (doInBackground(), callbacks) , Limitations of AsyncTask , Passing info to background tasks, Initiating background tasks, Scheduling background tasks (intro only, more later).

Connecting to the Internet:

Permissions, Building URIs, Opening and closing Internet connections, Parsing JSON in Android. (Because it's common.) , Sending requests and parsing response.

AsyncTaskLoader:

Intro to AsyncTaskLoader , loadInBackground() , AsyncTaskLoader callbacks , Benefits of loaders

Broadcast Receivers:

What is a Broadcast Receiver and a Broadcast Intent? , Broadcast Receiver Security and Lifecycle

Services:

What is a service? Long running task without a UI, Difference between Activity and Service , Start and stop services, Lifecycle methods, Foreground services, IntentService class, App priority (critical, high, low), How to create a new Service.

Notifications:

What is a Notification? , Notification Design Guidelines.

Triggering, Scheduling, and Optimizing Background: AlarmManager

Storing Data in your app:

Internal versus external storage, Privacy, sharing, security, encryption of your data , Shared Preferences: Store private primitive data in key-value pairs , SQLite Databases: Store structured data in a private database , Store data on the web with your own network server, Firebase for storing and sharing data in the cloud, Concept: Preferences , What are Settings and Preferences? , Settings best practices (harder to take away settings than to add, for usability reasons, Storing and retrieving preferences as key/value pairs using SharedPreferences, Different Settings types, Settings menu, Using Activity and PreferenceFragments to allow users to set preferences

Store data using SQLite database:

Overview of SQLite,OpenHelper Android class, Querying (dev) Searching (user) databases, Best practices for using databases in Android, Best practices for testing your database

Using Content Resolvers to access data: Content Providers and Content Resolvers work together, what is a content provider? , What is a content resolver? , How do they work together? , How to implement and use Content Resolvers

Content Providers: When to implement content providers , How to implement content providers (overview), Content URIs , UriMatcher, Content Provider authorities , Required methods on ContentProvider (query, insert, delete, update) , MIME types , Contracts , Making content provider data accessible to other apps by modifying manifest, and protecting data with permissions.

Using Loaders to Load and Display Data: Using loaders to asynchronously load data into an activity or fragment, Benefits of Loaders -- why use them? , Loader states (started, stopped, reset) , LoaderManager , Methods & callbacks to implement in Loaders: loadInBackground(), deliverResult() onStart/StopLoading(), onReset/Cancelled()),Registering listeners , Using CursorLoader with ContentProviders

UNIT-IV

[12 hours]

Fundamentals of iOS: Overview of MAC OS and X-CODE, Introduction to iPhone Architecture, Essential COCOA Touch Classes, Interface Builder, Nib File, COCOA and MVC Framework, Overview of features of latest iOS.

iPhone application development: Auto Layout, Views, Outlets and Actions, Different View Controller: single view Controller, Master-Detail View Controller, Navigation View Controller, Managing Application Memory, Application delegate, Handling Keyboard Input, UI Controllers: Label, Button, Text Field, Slider, Switch, Progress View, Page Control, Table View, Collection View, Image View, Text View, Web View, Map View, Date Picker, Picker View, Search Bar, Gestures, push notification, Image Picker, QR Code Scanner, Audio and Video.

Course Outcomes:

At the end of this subject, students should be able to:

1. Understand the existing state of mobile app development via researching existing apps, meeting with industry professionals, and formulating new ideas.
2. Understand the limitations and features of developing for mobile devices.
3. To be able to create mobile applications involving data storage in SQLite and other database tools.
4. Create a complete Mobile app with a significant programming component, involving the sensors and hardware features of the phone.
5. Build their ability to develop software with reasonable complexity on the android platform.
6. Design & Develop iphone and ipad application.

Text Books:

1. Android Programming (Big Nerd Ranch Guide), by Phillips, Stewart, Hardy and Marsicano.
2. Android Programming – Pushing the limits by Hellman

Reference Books:

1. Programming Android: Java Programming for the New Generation of Mobile Devices 2nd, Kindle eBook by Zigurd Mednieks (Author), G. Blake Meike (Author), Laird Dornin (Author).
2. Professional Android 4 Application Development Paperback – 2012 by Reto Meier
3. Learning Android Paperback – 2014 by Gargenta Marko


Web Resources:

1. <https://www.youtube.com/watch?v=SLNTnJkg6EE>
2. <https://www.youtube.com/watch?v=taSwS5rhtmc>
3. <https://www.youtube.com/watch?v=myjSxtAk9XM>
4. <https://www.youtube.com/watch?v=odqACn2Vgic>

LIST OF EXPERIMENTS

Experi ment. No.	Title	Learning Outcomes
1.	Basic of Android Programming	
1.1	To print “hello world” using string.xml file.	Understand about basic of android IDE
1.2	Android Life Cycle - Android system initiates its program within an Activity starting with a call on onCreate() callback method. There is a sequence of callback methods that start up an activity and a sequence of callback methods that tear down an activity as shown in the below Activity life cycle program	Understand about how application work.
2.	User Interface	
2.1	Different ways of handling button click event. a. Specifying the function in xml file: b. MainActivity implements listener class c. Anonymous Inner Class	CO2
2.2	Arithmetic Operations (Addition, Subtraction, Division, Multiplication)	CO2,CO3
2.3.	Custom Toast & Dialog Box	CO2
2.4	UI Controls (Android AutoCompleteTextView , Android spinner , Android progress bar , Image Button, Toggle Button, Button, CheckBox, RadioGroup, Raadio Button)	CO2
3.	Background Task	
3.1	Implicit Intent - Messages wiring components together. The source and destination for the content transfer are not known. Only the task and the action to be performed are known.	CO3
3.2	Explicit Intent - Messages wiring components together. The source and destination are known as well as the task and actions to be performed are known.	CO3
3.3	SMS Sending – Message can be sent using 2 methods – using	CO3

	Intent, using SMS Manager.	
3.4	Plotting a location on Google Map	CO2,CO3
3.5	GPS Tracking	CO2,CO3
3.6	Implement the concept of Async Task in Android App	CO2,CO3
3.7	Implement the concept of Shared preference in Android.	CO1,CO2,CO3
3.8	Demonstrate the use of shared preference as session in Android	CO2,CO3
4	Database	
4.1	Create Login & Registration Form using Sqlite Database	CO2,CO3
4.2	Create an Android App to display student details in ListView (using Database helper class and Adapter class).	CO2,CO3
4.3	Create an Android App to display student details in ListView (List must contain image and textview)	CO2,CO3
4.4	Implement the concept of Insert, Update and Delete Student facilities using fragment and database helper	CO2,CO3
5.	Multimedia	
5.1	Playing audio files in Android App	CO3
5.2	Playing video files in Android App.	CO3
6.	iOS Practical	
6.1	Installation of x-code on MAC	CO4
6.2	Write an application to demonstrate the use of table control & views.	CO4
6.3	Write an i-phone application which can play audio and video files	CO4

 ज्ञानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Distributed Systems								
Program: B. Tech CE/CSE/IT				Subject Code: CE0633			Semester: VI	
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
2	0	2	3	40	40	60	60	200

Course Outcome:

1. To understand foundations of Distributed Systems.
2. To understand the concepts of Remote Communication and Interprocess Communication
3. To study about various distributed client server models
4. To create an awareness of the major technical challenges in distributed systems design and implementation.
5. Know about emerging trends in distributed computing.

CONTENTS

UNIT-I

[12 hours]

Introduction Distributed System Concepts:

Introduction, Distributed Computing Models, Software Concepts, Issues in Designing Distributed Systems ,Client-Server Model , Case Studies: WWW 1.0 , 2.0 , 3.0, Protocols for Distributed Systems, Examples of Distributed Systems–Trends in Distributed Systems

UNIT-II

[12 hours]

Distributed computing paradigm and model:

Inter process Communication – the API for internet protocols – External data representation and Multicast communication. Network virtualization: Overlay networks. Case study: MPI Remote Method Invocation And Objects: Remote Invocation – Introduction – Request-reply protocols – Remote procedure call – Remote method invocation. Case study: Java RMI – Group communication – Publish-subscribe systems – Message queues – Shared memory approaches – Distributed objects – Case study: Enterprise Java Beans -from objects to components.

UNIT-III

[12 hours]

Peer to Peer Utilities:

Peer-to-peer Systems – Introduction – Napster and its legacy – Peer-to-peer – Middleware – Routing overlays. Overlay case studies: Pastry, Tapestry- Distributed File Systems –Introduction – File service architecture – Andrew File system. File System: Features-File model -File accessing models – File sharing semantics Naming: Identifiers, Addresses, Name Resolution – Name Space Implementation – Name Caches – LDAP

UNIT-IV

[12 hours]

Duplication and Synchronization:

Introduction – Clocks, events and process states – Synchronizing physical clocks- Logical time and logical clocks – Global states – Coordination and Agreement – Introduction – Distributed mutual exclusion – Elections – Transactions and Concurrency Control– Transactions -Nested transactions – Locks – Optimistic concurrency control – Timestamp ordering – Atomic Commit protocols - Distributed deadlocks – Replication – Case study – Coda.

Process Management:

Process Migration: Features, Mechanism – Threads: Models, Issues, Implementation. Resource Management

Course Outcomes:

At the end of this subject, students should be able to:

1. Understand the need and requirements of Distributed System
2. Discuss trends in Distributed Systems.

3. Apply network virtualization.
4. Apply remote method invocation and objects.
5. Design process and resource management systems.
6. Understand application of Distributed Systems

Text Books:

1. Distributed Systems: Principles and Paradigms, A S Tanenbaum & Martin Stee, 2/E, PHI, 2006
2. Distributed Systems Concepts & Design, Colouris, Dollimore, Kindberg, Pearson
3. Distributed Computing, Sunita Mahajan and Seema Shah, Oxford University
4. Distributed Operating Systems by P. K. Sinha, PHI

Reference Books:

1. Distributed Systems: Principles and Paradigms, Tanenbaum
2. Distributed Computing, Fundamentals, Simulations and Advanced topics, 2nd Edition, Hagit Attiya
3. Distributed Systems: Concepts and Design, G. Coulouris, J. Dollimore, and T. Kindberg,
4. Java Network Programming & Distributed Computing by David Reilly, Michael Reill


Web Resources

1. [https://nptel.ac.in/courses/Distributed System](https://nptel.ac.in/courses/Distributed%20System)

LIST OF EXPERIMENTS

Experiment No.	Title	Learning Outcomes
1	Write a program to implement hello world service using RMI	Students will get hands on practice about RMI .
2	Write a program to implement calculator using RMI	Students will get hands on practice about RMI .
3	Write a program to implement time service using RMI	Students will get hands on practice about RMI .

4	Write a program to implement hello world service using RPC	Students will get hands on practice about RPC .
5	Write a program to implement date service using RPC	Students will get hands on practice about RPC .
6	Write a program to implement Echo SOCKET in JAVA	Students will get hands on practice about SOCKET Programming .
7.	Write a program to implement Echo server using RPCGEN	Students will get hands on practice about RPCGEN .
8	Write a program to implement producer-consumer concept using THREAD	Understand threading.
9.	Write a program to find the length of string using THREAD	Understand threading.

 ज्ञानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Cryptography & Network Security								
Program: B. Tech CE/CSE/IT					Subject Code: CE0634			Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
3	0	2	4	40	40	60	60	200

Course Objectives

1. To understand various Cryptographic Techniques.
2. Acquire background on hash functions; authentication; firewalls; intrusion detection techniques.
3. To understand the various Security Applications.
4. To understand various protocols for network security to protect against the threats in the networks.
5. Identify and mitigate software security vulnerabilities in existing systems.

CONTENTS

UNIT-I

[12 hours]

Fundamentals: Basic objectives of cryptography, Security mechanisms, OSI Security Architecture, Classical Encryption techniques, Cipher principles, cryptanalysis, Attack models.

Block ciphers: Block cipher design principles and modes of operation, Feistel cipher structure, Data Encryption standard (DES), International Data Encryption Algorithm, Blowfish, variants of DES, AES with structure, its transformation functions, key expansion.

UNIT-II

[12 hours]

Public Key Cryptography: Overview of Asymmetric Key Cryptography, RSA algorithm, its computational aspects and security, Elliptic Curve Cryptography, Knapsack Algorithm, Diffie-Hellman key Exchange, Man-in-Middle attack.

Key management: Key management and distribution, symmetric key distribution using symmetric and asymmetric encryptions, distribution of public keys.

UNIT-III

[12 hours]

Hash Function: Message Authentication Codes, its requirements and security, MACs based on Hash Functions, Security of Hash Functions, Secure Hash Algorithm, HMAC, Digital Signature, its properties, requirements and security, various digital signature schemes (Elgamal and Schnorr), NIST digital Signature algorithm.

Network Security: Authentication Applications like Kerberos, X.509 Authentication Service, PGP.

UNIT-IV

[12 hours]

IP& Web Security: IPSec architecture, Applications of IPSec, Benefits of IPSec, and IPSec protocols, Web Security threats, Secure Socket Layer, Secure Electronic Transaction.

System Level Security: Intrusion detection, Viruses and related Threats - Virus Counter measures, Firewall Design Principles, Trusted Systems.

Course Outcomes

At the end of this subject, students should be able to:

1. To understand the fundamental principles of access control models and techniques, authentication and secure system design.
2. To understand and apply the various symmetric key algorithms and asymmetric key algorithms.
3. To understand the concepts of hashing with algorithms, digital signature and apply them.
4. To understand and use the message authentication and its requirement.
5. Analyze and design network security protocols.

6. To acquire the hands-on skills and the knowledge required for job competency.

Text Books:

William Stallings, "Cryptography and Network Security - Principles and Practices", Prentice Hall of India, Third Edition, 2003.

Reference Books:

- 1) Cryptography & Network Security, Forouzan, Mukhopadhyay, McGrawHill
- 2) Cryptography and Network Security (2nd Ed.), Atul Kahate, TMH
- 3) Information Systems Security, Godbole, Wiley-India
- 4) Information Security Principles and Practice, Deven Shah, Wiley-India
- 5) Bruce Schneier, "Applied Cryptography", John Wiley & Sons Inc, 2001.
- 6) A. Das and C. E. Veni Madhavan, Public-Key Cryptography: Theory and Practice, Pearson Education Asia.


Web Resources

1. Software: cryptool (www.cryptool.org)
2. Software: Wireshark (www.wireshark.org)
3. <http://www.cryptix.org/>
4. williamstallings.com/Extras/Security-Notes/
5. www.cs.bilkent.edu.tr/~selcuk/teaching/cs519/
6. <http://freevideolectures.com/Course/3027/Cryptography-and-Network-Security>
7. http://cs.brown.edu/courses/csci1510/2013_lectures.html

LIST OF PRACTICALS

Sr. No.	Title	Learning Outcomes
1.	To implement Caesar Cipher Encryption - Decryption.	CO1
2.	To implement Mono-alphabetic Cipher Encryption – Decryption.	CO1
3.	To implement Hill Cipher Encryption	CO1
4.	To implement Poly-alphabetic Cipher (Vigener Cipher) Technique	CO1

5.	To implement Play-Fair Cipher Technique.	CO1
6.	Write a program to implement Rail-Fence, Simple columnar Encryption Technique.	CO1
7.	To implement S-DES algorithm for data encryption.	CO2
8.	Write a program to implement RSA asymmetric (public key and private key)-Encryption.	CO2
9.	Implement Diffi-Hellmen Key exchange Method.	CO2
10.	Implement a digital signature algorithm.	CO3
11.	Perform various encryption-decryption techniques with cryptool.	CO4
12.	Study and use the Wireshark for the various network protocols.	CO4

 ज्ञानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Data Compression								
Program: CE/CS/IT				Subject Code: CE0629			Semester: VI	
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
2	0	2	4	40	40	60	60	200

Course Objectives:

1. To make known to students to basic applications, concepts, and techniques of Data compression.
2. To develop skills for using recent data compression software to solve practical problems in a variety of disciplines.
3. To gain experience doing independent study and research.
4. Improve fundamental understanding of data compression methods for text, images, and video and related issues in the storage, access, and use of large data sets.
5. Select and give reasons that are sensitive to the specific application and particular circumstance, most appropriate compression techniques for text, audio, image and video information

CONTENTS

UNIT-I

[12 hours]

Compression Techniques

Lossless Compression, Lossy Compression, Measures of Performance Mathematical Preliminaries for Lossless Compression Models : Physical Models, Probability Models, Markov Models, Composite Source Model Coding, Uniquely Decodable Codes, Prefix Codes, Algorithmic Information Theory, Minimum Description Length Principle, Huffman Coding, The Huffman Coding Algorithm, Minimum

Variance Huffman Codes, Adaptive Huffman Coding, Applications of Huffman Coding, Lossless Image Compression, Text Compression and Audio Compression

UNIT-II

[12 hours]

Arithmetic Coding Introduction

Coding a Sequence, Generating a Tag, Deciphering the Tag, Generating a Binary Code, Uniqueness and Efficiency of the Arithmetic Code, Algorithm Implementation, Integer Implementation, Comparison of Huffman and Arithmetic Coding, Adaptive Arithmetic Coding

UNIT-III

[12 hours]

Dictionary Techniques

Static Dictionary Diagram, Coding Adaptive Dictionary the LZ77 Approach the LZ78 Approach Applications File Compression —UNIX compress Image Compression—The Graphics Interchange Format (GIF) Image Compression—Portable Network Graphics (PNG) Compression over Modems — V.42 bis

UNIT-IV

[12 hours]

Lossless Compression

Standards zip, gzip, bzip, unix compress, GIF, JBIG. Image & Video compression Basis functions and transforms from an intuitive point, JPEG, MPEG, Vector Quantization, case study of WinZip, WinRar Wavelet based compression Fundamentals of wavelets, various standard wavelet bases, Multi resolution analysis and scaling function and JPEG 2000.

Course Outcomes

At the end of this subject, students should be able to:

1. Understand importance of data compression.
2. Develop a reasonably sophisticated data compression application.
3. Select methods and techniques appropriate for the task.
4. Develop the methods and tools for the given task.
5. Illustrate the concept of various algorithms for compressing text, audio, image and video information.
6. To develop a research oriented thinking in the area of Data Compression.

Text Books:

1. Introduction to Data Compression, Khalid Sayood, Morgan Kaufmann
2. The Data Compression book, Mark Nelson, Jean Loup Gailly

Reference Books:

1. Data Compression: “The Complete Reference”, David Saloman, Springer.
2. An Introduction to Information Retrieval, Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze, Cambridge University Press, Cambridge, England.

Web Resources:


1. <https://www.barracuda.com/glossary/data-compression>
2. <https://developer.mozilla.org/en-US/docs/Web/HTTP/Compression>
3. <http://www.data-compression.info/>

LIST OF EXPERIMENTS

Experiment. No.	Title	Learning Outcomes
1.1	Write a program that compresses and displays uncompressed windows BMP image file.	Basic Knowledge of compression
1.2	Write a program to generate binary code in case of arithmetic coding.	Basic Knowledge of compression
1.3	Write a program to count the occurrences of different letters by reading the given text file and also find the probability of each letter with number of bits required for them using the formula: No. of bits= $1/\log_2$ prob.	Basic Knowledge of compression
1.4	Write a Program to check whether the given code is prefix or not.	Basic Knowledge of compression
1.5	Write a program to determine whether the set of given codes is uniquely decodable or not.	Basic Knowledge of compression

2.	Arithmetic and Huffman Coding	
	Write a program to generate binary code in case of arithmetic coding.	Knowledge of Arithmetic compression
2.1	Implement Huffman Code (HC) to generate binary code when symbol and probabilities are given	Knowledge of Huffman compression
3.	Arithmetic and Huffman Coding	
3.1	Implement Huffman code which can compress given file and decompress compressed file	Knowledge of Huffman compression
3.2	Implement adaptive Huffman program to compress decompressed file.	Knowledge of Huffman compression
4	Text Compression Techniques	
4.1	Write a program to Implement LZ77 algorithm.	Knowledge of Text Compression Techniques
4.2	Write a program to Implement LZ55 algorithm.	Knowledge of Text Compression Techniques
5.	Text Compression Techniques	
5.1	Write a program to Implement LZ78 algorithm	Knowledge of Text Compression Techniques
6.	Image Compression Techniques	
6.1	Write a program which performs JPEG compression, process step by step for given 8x8 block and decompression also	Knowledge of Image Compression Techniques

6.2	Study of Speech Compression Techniques.	Knowledge of Speech Compression Techniques
------------	---	---

 इजानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Quantum Computing								
Program: B.Tech CSE				Subject Code: CS0604				Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
2	0	2	4	40	40	60	60	200

Pre-requisite: Discrete Structures.

Course Objectives:

1. The objective of this course is to provide the students an introduction to quantum computation.
2. Much of the background material related to the algebra of complex vector spaces and quantum mechanics is covered within the course.

CONTENTS

UNIT-I

Introduction to Quantum Computation

[12 hours]

Motivation, foundations, and prominent applications. Review of linear algebra in the context of quantum information, Dirac's bracket notation, limitation of classical algorithms. Quantum bits, Bloch sphere representation of a qubit, multiple qubits. Background Mathematics and Physics: Hilber space, Probabilities and measurements, entanglement, density operators and correlation, basics of quantum mechanics, Measurements in bases other than computational basis.

Unit 2:

Quantum Circuits

[12 hours]

Single qubit gates, multiple qubit gates, design of quantum circuits. Quantum Information and Cryptography: Comparison between classical and quantum information theory. Bell states. Quantum teleportation. Quantum Cryptography ((post-quantum security, quantum key distribution), no cloning theorem, Simon's problem and the Bernstein -V-azirani algorithm. Grover's quantum search algorithm, the BBBV Theorem, and applications of Grover's algorithm. RSA, and Shor's integer factorization algorithm

Unit 3:

Quantum Algorithms

[12 hours]

Classical computation on quantum computers. Relationship between quantum and classical complexity classes. Deutsch's algorithm, Deutsch's-Jozsa algorithm, Shor factorization, Grover search, analyzing quantum algorithms, and implementing quantum circuits via QISKIT

Unit 4:

Noise and error correction

[12 hours]

Quantum information (superdense coding, no-cloning theorem, quantum teleportation) Applications (quantum money, the Elitzur-Vaidman bomb). Graph states and codes, Quantum error correction, fault-tolerant computation, Distance measures, Knill-Laflamme conditions, quantum error-correcting codes, Hamming bound.

OUTCOME:

Upon Completion of the course, the students should be able to:


1. Learn quantum computation and quantum information
2. Understand quantum entanglement, quantum algorithms
3. Understand quantum channels
4. Learn quantum information theory

Suggested Books:

1. Nielsen M. A., Quantum Computation and Quantum Information, Cambridge University Press.
2. Benenti G., Casati G. and Strini G., Principles of Quantum Computation and Information, Vol. I: Basic Concepts, Vol II: Basic Tools and Special Topics, World Scientific
3. Pittenger A. O., An Introduction to Quantum Computing Algorithms

List of Experiments

Experiment No.	Experiment Title	Learning Outcomes
1	Linear algebra simulation with quantum amplitudes	Learn quantum computation and quantum information
2	Quantum machine learning algorithms based on Grover search	Learn quantum computation and quantum information
3	Implementations of Quantum sampling techniques	Learn quantum computation and quantum information
4	Simulation of Quantum annealing algorithm	Understand quantum entanglement, quantum algorithms
5	Implementation of Quantum-KNN algorithm	Understand quantum entanglement, quantum algorithms
6	Implementation of Quantum-Support Vector Machines algorithm	Understand quantum entanglement, quantum algorithms
7	Implement the Quantum neural networks	Understand quantum channels
8	Simulation of Quantum-enhanced reinforcement learning	Learn quantum information theory
9	Simulation of Quantum-enhanced clustering techniques	Learn quantum information theory
10	Simulation of Quantum-enhanced ensemble techniques	Understand quantum entanglement, quantum algorithms

 इजानेन प्रकाशते जगत् INDUS UNIVERSITY				INDUS INSTITUTE OF TECHNOLOGY & ENGINEERING Constituent Institute of Indus University				
Subject: Internet of Things								
Program: B. Tech CE/CS/IT					Subject Code: CE0622			Semester: VI
Teaching Scheme (Hours per week)				Examination Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credits	University Theory Examination	University Practical Examination	Continuous Internal Evaluation (CIE)- Theory	Continuous Internal Evaluation (CIE)- Practical	Total
2	0	2	4	40	40	60	60	200

Course Objectives

1. Introduce evolution of internet technology and need for IoT.
2. Train the students to build IoT systems using sensors, single board computers and open source IoT platforms that help in skill development.
3. To identify the design, development and security challenges in IoT Systems.
4. To study IoT Applications in Different Domains and be able to measure their performance that enhances the employability skills of students.
5. To implement basic IoT Applications on Embedded Platforms to enhance entrepreneurship skills in students.

CONTENTS

UNIT-I

[12 hours]

Introduction to IoT

Defining IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT, Communication models & APIs, Machine to Machine, IoT versus Machine to Machine, Challenges in IoT: Design challenges, Development challenges, Security challenges

Application of IoT: Home automation, Industry applications, Surveillance applications

UNIT-II

[12 hours]

IoT Reference Architecture- Getting Familiar with IoT Architecture, Various architectural views of IoT such as Functional, Information, Operational and Deployment.

Constraints affecting design in IoT world- Introduction, Technical design Constraints.

Web Infrastructure for managing IoT Resources: Introduction, Open IoT Architecture for IoT/Cloud Convergence, Scheduling Process and IoT Service Lifecycle, Device/Cloud Collaboration Framework

UNIT-III

[12 hours]

Internet of Things Privacy, Security and Governance: Introduction, Overview of Governance, Privacy and Security Issues, Contribution from FP7 Projects, Security, Privacy and Trust in IoT-Data-Platforms for Smart Cities, First Steps Towards a Secure Platform, Smartie Approach. Data Aggregation for the IoT in Smart Cities and Security.

UNIT-IV

[12 hours]

PREPARING IOT PROJECTS: Interoperability in IoT, Introduction to Arduino Programming, Integration of Sensors and Actuators with Arduino, Introduction to Python programming, Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi.

Course Outcomes

At the end of this subject, students should be able to:

1. Explain the Principles of Internet of Things
2. Design and develop IoT based sensor systems.
3. Employ IoT Solutions to Real Time Engineering Problems
4. Familiar with the Data Management Techniques, Architectures and various key enablers to enable practical IoT systems
5. Identify the Challenges and Research Scope in Communication Protocols used in IoT Applications.
6. Solve IoT security problems using light weight cryptography

Text Books:

1. Internet of Things Principles and Paradigms, Edited By Rajkumar Buyya, Amir Vahid Dastjerdi, Morgan Kaufmann, ELSEVIER
2. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stamatis Karnouskos, Stefan Avesand, David Boyle, “From Machine to Machine to Internet of Things”, Elsevier Publications, 2014

Reference Books:

1. Fundamentals of Wireless Sensors Networks Theory and Practice, Waltenegus Dargie and Christian Poellabauer, WILEY Series
2. Rethinking the Internet of Things A Scalable approach to connecting everything, Francis daCosta, Apress Open
3. Arduino Cookbook, Michael Margolis, O'Reilly
4. Internet of Things – From Research and Innovation to Market Deployment, Edited By Ovidiu Vermesan and Peter Friess, River Publishers

Web Resources:

1. NPTEL Lecture: <https://nptel.ac.in/courses/106105166/>

LIST OF EXPERIMENTS

Experi- ment. No.	Title	Learning Outcomes
1	Familiarization with Arduino/Raspberry Pi and perform necessary software installation.	CO-1
2	To interface Push button/Digital sensor (IR/LDR) with Arduino/Raspberry Pi and write a program to turn ON LED when push button is pressed or at sensor detection.	CO-2,3
3	Interface analog sensor (PIR Sensor, temperature sensor LM35, Ultrasonic Sensor) with - Arduino	CO-2
4	To interface Bluetooth with Arduino/Raspberry Pi and write a program to turn LED ON/OFF when '1'/'0' is received from a smart phone using Bluetooth.	CO-2,3
5	To install MySQL database on Raspberry Pi and perform basic SQL queries.	CO-3,4
6	Write a program on Arduino/Raspberry Pi to publish temperature data to MQTT broker.	CO-1,3

7	Write a program on Arduino/Raspberry Pi to retrieve temperature and humidity data from thingspeak cloud.	CO-3,4
8	Write a program to create a TCP server on Arduino/Raspberry Pi and respond with humidity data to the TCP client when requested.	CO-4,5
9	Creating a webpage and display the values available through Arduino	CO-4,5
10	Open Ended Experiment-Mini Project: Working on any IOT Application	CO-5,6