Shri Vile Parle Kelavani Mandal's

Dwarkadas J. Sanghvi College of Engineering

(Autonomous College Affiliated to the University of Mumbai)

Scheme and detailed syllabus

of

Third Year B.Tech

in

Artificial Intelligence and Machine Learning

(Semester V)

Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Proposed Scheme for Third Year Undergraduate Program in Artificial Intelligence and Machine Learning: Semester V (Autonomous)
(Academic Year 2023-2024)

Sr no.	Course Code	Course	Teaching Scheme(hrs)				Continuous Assessment (A) (marks)			Semester End Assessment (B) (marks)					Aggregate (A+B)	Total Credits
			Th	P	T	Credits	Th	T/W	Total CA (A)	Th	O	P	O &P	Total SEA (B)		
1	DJ19AMC501	Digital Signal and Image Processing	3	--	--	3	25	--	25	75	--	--	--	75	100	4
	DJ19AML501	Digital Signal and Image Processing Laboratory	--	2	--	1	--	25	25	--	25	--	--	25	50	
2	DJ19AMC502	Machine Learning-II(Deep Learning)	3	--	--	3	25	--	25	75	--	--	--	75	100	4
	DJ19AML502	Machine Learning-II(Deep Learning) Laboratory	--	2	--	1	--	25	25	--	25	--	--	25	50	
3	DJ19AMC503	DevOps	3	--	--	3	25	--	25	75	--	--	--	75	100	4
	DJ19AML503	DevOps Laboratory	--	2	--	1	--	25	25	--	25	--	--	25	50	
4	DJ19AML504	Programming Laboratory-III(Full stack development using python)	--	4	--	2	--	25	25	--	--	--	25	25	50	2
5	DJ19AMEC5011	Elective 1: Cloud Computing	3	--	--	3	25	--	25	75	--	--	--	75	100	4
	DJ19AMEL5011	Cloud Computing Laboratory	--	2	--	1	--	25	25	--	25	--		25	50	
6	DJ19AMEC5012	Elective 2: Advanced Data Structures and Algorithms	3	--	--	3	25	--	25	75	--	--	--	75	100	4
	DJ19AMEL5012	Advanced Data Structures and Algorithms Laboratory	--	2	--	1	--	25	25	--	25	--	--	25	50	
7	DJ19AMEC5013	Elective 3: Recommendation Systems	3	--	--	3	25	--	25	75	--	--	--	75	100	4
	DJ19AMEL5013	Recommendation Systems Laboratory	--	2	--	1	--	25	25	--	25	--	--	25	50	
8	DJ19A5	Enviornmental Studies(1 hr)	1	--	--	--	--	--	--	--	--	--	--	--	--	--
9	DJ19ILL1	Innovative Product Development III(C)	--	2	--	1	--	25	25	--	25	--	--	25	50	1
			19	14	0	19	150	200	350	450	175	0	25	650	1000	19

Prepared by Checked by Head of the Department Vice Principal Principal

Program: Artificial Intelligence & Machine Learning	**T.Y.** **B.Tech.**	**Semester:** **V**
Course: Digital Signal and Image Processing (DJ19AMC501)		
Course: Digital Signal and Image Processing Laboratory (DJ19AML501)		

Pre-requisite:

- Ability to deal with applied and abstract mathematical concepts.

Course Objective:

1. To introduce students with the basic idea of signals and systems analysis with its characterization in time and frequency domain.
2. To implement algorithms that perform basic image processing- enhancement and filtering
3. To implement algorithms for basic image segmentation

Course Outcomes (CO): On completion of the course, the students will be able to:

1. Classify signals and systems on the basis of their properties and analyze the implications in the context of practical signals and systems.
2. Represent signals in the time and frequency domain using multiple representations and analyze LTI systems using convolution in the frequency domain.
3. Implement image enhancement techniques in spatial and frequency domain.
4. Interpret and apply image segmentation and representation techniques for object recognition.

Digital Signal and Image Processing (DJ19AMC501)

Unit	Description	Duration
1	**Discrete-Time Signal and Discrete-Time System:** Introduction to Digital Signal Processing, Sampling and Reconstruction, Standard DT Signals, Concept of Digital Frequency, Representation of DT signal using Standard DT Signals, Signal Manipulations (shifting, reversal, scaling, addition, multiplication). Classification of Discrete-Time Signals, Classification of Discrete Systems. Linear Convolution formulation for 1-D signal (without mathematical proof), Circular Convolution (without mathematical proof), Linear convolution using Circular Convolution. LTI system, Concept of Impulse Response and Step Response, Output of DT system using Time Domain Linear Convolution.	**09**

2	**Discrete Fourier Transform**: Introduction to DTFT, Relation between DFT and DTFT, DFT of DT signal, Inverse DFT. Properties of the DFT: Scaling and Linearity, Symmetry for real valued signal, Periodicity, Time Shift and Frequency Shift, Time Reversal, Convolution Property **Fast Fourier Transform:** Need of FFT, Radix-2 DIT-FFT algorithm. Flow graph for N=4 and 8 using Radix-2 DIT-FFT, Inverse FFT algorithm, Comparison of complex and real, multiplication and additions of DFT and FFT. Overlap Add Algorithm and Overlap Save Algorithm and implementation using FFT.	**10**
3	**Basics of Image Processing**- Image acquisition, Sampling, Quantization, Image Types, and Image formats. Colour models RGB and CMYK models.	**02**
4	**Image Enhancement in spatial domain** Point processing techniques, Neighborhood processing. Image Smoothing: Spatial Filters-Linear Filters-Averaging filter, Median filter, Sharpening Spatial Filters- The Laplacian, Unsharp Masking and High boost Filtering, Using First-Order Derivatives —The Gradient-Sobel, Prewitt and Roberts operator, Histogram processing (Stretching & Equalization) **Image Enhancement in Frequency domain:** 2D-DFT, Properties of 2D-DFT, Low pass, High pass and Homomorphic filtering.	**10**
5	**Image Restoration:** Overview of Degradation models –Unconstrained and constrained restorations-Inverse Filtering, WienerFilter.	**03**
6	**Image Segmentation**: Basic relationships between pixels -Neighbors, Connectivity, Detection of discontinuities- Point, Line, Edge detection, Region-based segmentation- Region Growing, Region Splitting and merging. Region Identification: chain code, simple geometric border representation, Boundary description using segment sequences. Edge Linking, Hough Transform.	**05**
	TOTAL	**39**

Books Recommended:

Text Books:

1. Rafael C. Gonzalez, Richard E. Woods, _Digital Image Processing‘, Pearson, 4th Edition, 2018.
2. Tarun Rawat, Signals and Systems, Oxford Higher Education, Paperback – 19 July 2010
3. V. Krishnaveni and A.Rajeshwari, “Signals and Systems”, Wiley-India, First Edition 2012.

Reference Books:

1. Simon Haykin and Barry Van Veen, “Signals and Sytems”, John Wiley and Sons, Second Edition, 2004.
2. Alan V. Oppenhiem, Alan S. Willsky and S. Hamid Nawab, “Signals and Systems”, Prentice-Hall of India, Second Edition, 2002.
3. Anil K Jain, Fundamentals of digital image processing Paperback, PHI – 1 January 2015
4. Willliam K Pratt, Digital Image Processing John Willey, 2002.
5. Malay K. Pakhira, Digital Image Processing and Pattern Recognition, First Edition, PHI Learning Pvt. Ltd., 2011.

Digital Signal and Image Processing Laboratory (DJ19AML501)	
Sr. no.	**Name of Experiment (Any Eight)**
1	Implementation of Linear and Circular Convolution of two discrete time sequences.
2	Implement Overlap and Add method for computing the convolution of two variable length sequences.
3	To understand the effect of Sampling and Quantization on image quality. Study image statistics. (Mean, Variance, Entropy)
4	To perform Spatial Domain Image Enhancement using different Point Processing techniques.
5	To perform Image steganography and retrieve the secret image from the stego image.
6	Implement homomorphic filter
7	To perform frequency domain Image Enhancement techniques.
8	Perform image zooming
9	To perform different Image Segmentation operations using different operators and Canny edge detection.

Prepared by Checked by Head of the Department Principal

Program: Artificial Intelligence & Machine Learning	**T.Y.** **B.Tech.**	**Semester:** **V**
Course: Machine Learning-II (Deep Learning)(DJ19AMC502)		
Course: Machine Learning-II (Deep Learning) Laboratory (DJ19AML502)		

Pre-requisite: Artificial Intelligence, Machine Learning

Course Objective:
1. To explore deep learning techniques with different learning strategies

Course Outcomes (CO): On completion of the course, the students will be able to:
1. Understand the fundamentals of deep learning
2. Apply supervised deep learning algorithms
3. Apply unsupervised deep learning algorithms
4. Understand the fundamentals of adversarial networks

Machine Learning-II (Deep Learning)(DJ19AMC502)		
Unit	**Description**	**Duration**
1	**Introduction to Neural Networks and Deep Learning** Fundamentals of Neural Network and Deep Learning, Machine Learning vs Deep Learning, Deep Learning applications, Artificial Neural Network (ANN): Association of biological neuron with artificial network, activation functions, weights, bias, threshold, learning rate, momentum factor; McCulloch Pitts Neuron: Theory and architecture; linear separability; Hebb Network: Theory and algorithm, ANN architectures. Hyperparameter tuning and batch normalization	05
2	**Supervised Learning with Neural Networks** Perceptron, Multilayer Perceptrons (MLPs), Representational power of Perceptron and MLPs, Training rule, Sigmoid neurons, Gradient Descent and Delta Rule, Multilayer Networks: A differentiable threshold Unit, Feedforward Neural Networks, Representation Power of Feedforward Neural Networks; **Backpropagation Algorithm**: EBPTA, Convergence and local minima, Regularization for Deep Learning: Parameter Norm Penalties, Dataset Augmentation, Noise Robustness, Early Stopping, Sparse Representation, Dropout. Principal Component Analysis and its interpretations, Singular Value Decomposition.	09
3	**Convolutional Neural Network (CNNs)** The Convolution Operation, Types of kernels, Kernel filter, Principles behind CNNs, Multiple Filters, convolution for images, sparse interactions, parameter sharing, Pooling, Convolution and Pooling as an Infinity Strong Prior, Variants of Basic Convolution Function, Efficient Convolution Algorithms. CNN applications, **ConvNet Architectures:** Discussions on famous convnet architectures: AlexNet, ZFNet, VGG, GoogLeNet, ResNet InceptionNets, DenseNets	07

4	**Recurrent Neural Networks (RNNs)** Introduction to Sequence Models and RNNs, RNN Model, Backpropagation through Time (BPTT), Different Types of RNNs: Unfolded RNNs, Seq2Seq RNNs, Long Short-Term Memory (LSTM), Bidirectional RNN, Vanishing Gradients with RNNs, Gated Recurrent Unit (GRU), RNN applications	07
5	**Unsupervised Learning with Neural Networks** Kohonen Self-Organizing Feature Maps – architecture, training algorithm, Kohonen Self Organizing Motor Map. **Autoencoders:** Introduction and their relation to PCA, Need, components, properties, architecture and types of encoders. Linear Factor Methods such as Probabilistic PCA and Factor Analysis, Independent Component Analysis, Sparse Coding; Regularization in autoencoders, Denoising autoencoders, Sparse autoencoders, Contractive autoencoders, deep autoencoders, Undercomplete Autoencoders, Stochastic Encoders and Decoders, variational autoencoders(VAEs), Applications of Autoencoders.	06
6	**Adversarial Networks** Introduction to adversarial Networks, Generative Vs Discriminative Modeling, Generative Adversarial Networks (GAN): Probabilistic Generative Model, Types of GANs, Applications of GANs. Diffusion models vs GANs vs VAEs, Neural Style Transformers and language models	05
	TOTAL	39

Books Recommended:

Text Books:

1. Simon Haykin, “Neural Networks and Learning Machines”, Pearson Prentice Hall, 3rd Edition, 2010.
2. S. N. Sivanandam and S. N. Deepa, “Introduction to Soft Computing”, Wiley India Publications, 3rd Edition, 2018.
3. David Foster, “Generative Deep Learning”, O’Reilly Media, 2019.
4. Denis Rothman, “Hands-On Explainable AI (XAI) with python”, Packt, 2020.

Reference Books:

1. Ian Goodfellow and Yoshua Bengio and Aaron Courville, “Deep Learning”, An MIT Press, 2016
2. François Chollet, “Deep Learning with Python”, Manning Publication, 2017.
3. Josh Patterson, Adam Gibson, “Deep Learning: A Practitioner's Approach”, O’Reilly Publication, 2017. Andrew W. Trask, Grokking, “Deep Learning”, Manning Publication, 2019.
4. John D. Kelleher, “Deep Learning”, MIT Press Essential Knowledge series, 2019.

Web Links:

1. d2l-en.pdf
2. Learning Rule: http://vlabs.iitb.ac.in/vlabs-dev/labs/machine_learning/labs/explist.php
3. ANN Virtual Lab: http://cse22-iiith.vlabs.ac.in/List%20of%20experiments.html
4. Deep Learning: https://vlab.spit.ac.in/ai/#/experiments

5. NPTEL Course: Deep Learning Part 1: https://onlinecourses.nptel.ac.in/noc19_cs85/preview

Suggested List of Experiments:

Machine Learning-II (Deep Learning) Laboratory (DJ19AML502)

Sr. No.	**Title of the Experiment**
1	To study various tools: Torch, TensorFlow, Keras
2	To build own Neural Network from scratch
3	To implement hyper parameter tuning
4	To implement EBPTA algorithm.
5	Understanding ANN using Tensor Flow
6	Visualizing Convolutional Neural Network using Tensor Flow with Keras Data.
7	OCR using teassaeract
8	Object detection using RNN using YOLO and Tensor Flow
9	GAN generator/discriminator.
10	Visualizing autoencoders
11	Students are supposed to complete any one mini project not limited to following list of projects. 1. Sequence Prediction 2. Object Detection 3. Traffic Sign Classification 4. Automatic Music Generation 5. Music Genre Classification 6. Text Summarizer 7. Gender and Age Detection Using Voice 8. Chatbot Using Deep Learning 9. Neural Style Transfer 10. Face Aging 11. Driver Drowsiness Detection 12. Language Translator 13. Image Reconstruction

Minimum eight experiments and mini project from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt

Prepared by Checked by Head of the Department Principal

Program: Artificial Intelligence & Machine Learning	T.Y. B.Tech.	Semester: V
Course: Devops (DJ19AMC503)		
Course: Devops Laboratory (DJ19AML503)		

Pre-requisite: --

Objectives:

1. The objective of this course is to familiarize learners with different development frameworks.
2. Introduce the principles and processes of software engineering and Devopps.

Outcomes: On completion of the course, the students will be able to:

1. Apply software engineering principles for application development.
2. Apply various principles, phases and activities of agile as well as scrum methodology.
3. Understand and implement Devops principles for CI/CD.
4. Apply testing process for application development.
5. Apply Configuration Management Tools using Containerization.

Devops (DJ19AMC503)		
Unit	**Description**	**Duration**
1	**Introduction:** Fundamentals of Software Engineering- process framework, Software Development Life Cycle (SDLC) Process Models: Incremental and Evolutionary. **Devops:** Introduction to Devops, definition, History of Devops, Objectives, Continuous Integration & Deployment, Containers and Virtual Development, Configuration Management Tools.	07
2	**Fundamentals of Agile Process:** Need of Agile software development, Agile Manifesto and Principles, Stakeholders and Challenges, Overview of Agile Development Models: Scrum, Extreme Programming, Feature Driven Development, Crystal, Kanban, and Lean Software Development, Business benefits of software agility.	07
3	**Source Code Management:** Version Control: GIT Features, 3-Tree Architecture, GIT – Clone /Commit / Push, GIT Hub Projects, GIT Hub Management, GIT Rebase & Merge, GIT Stash, Reset, Checkout, GIT Clone, Fetch, Pull, Membership GITHUB.	06
4	**Continuous Integration:** Continuous Delivery and Deployment, Benefits of CI/CD, Metrics to track CICD practices, Continuous integration, tools, Build & Test Applications with Continuous Integration, Scheduling build Jobs, Build Scripts, Build Pipeline, Master & Slave Node Configuration, Workspace Management, Security and plugins, Other integration tools	06

5	**Continuous Testing:** Introduction to Selenium, Installing Selenium, Creating Test Cases in Selenium WebDriver, Run Selenium Tests in Jenkins Using Maven, Functionality Testing, UI Testing, Performance Testing, Security Testing.	05
6	**Configuration Management in Devops:** The Process of Configuration, Configuration Management in DevOps. **Configuration Management Tools Containerization:** Docker introduction, Docker Image, working with Docker Containers, Docker Engine, Creating Containers with an Image, Working with Images, Docker Hub, Docker Trusted Registry, Docker File & Commands. **Devops Monitoring Tool:** Introduction to Nagios, Architecture.	08
	TOTAL	**39**

Books Recommended:

Text books:

1. Software Engineering: A Practitioner's Approach, Eight Edition by Roger S. Pressman and Bruce R. Maxim, McGraw-Hill Education, 2019.
2. Karl Matthias & Sean P. Kane, Docker: Up and Running, O'Reilly Publication, 2nd edition, 2018.
3. Len Bass, Ingo Weber, Liming Zhu,"DevOps, A Software Architects Perspective", Addison Wesley Pearson Publication, Ist edition, 2015.
4. John Ferguson Smart," Jenkins, The Definitive Guide", O'Reilly Publication 1st 2011.
5. Mastering Puppet 5: Optimize enterprise-grade environment performance with Puppet, by Ryan Russell Yates Packt Publishing (September 29, 2018)

Reference Books:

1. DevOps: Continuous Delivery, Integration, and Deployment with DevOps: Dive By Sricharan Vadapalli, Packt, 2018.
2. Agile Testing: A Practical Guide For Testers And Agile Teams, Lisa Crispin, Janet Gregory, Pearson, 2010.
3. More Agile Testing: Learning Journeys for the Whole Team By Janet Gregory, Lisa Crispin, Addison Wesley, 2015.
4. DevOps: Puppet, Docker, and Kubernetes By Thomas Uphill, John Arundel, Neependra Khare, Hideto Saito, Hui-Chuan Chloe Lee, Ke-Jou Carol Hsu, Packt, 2017
5. Agile Project Management: Creating Innovative Products, Second Edition By Jim Highsmith, Addison-Wesley Professional, 2009
6. Learning Agile: Understanding Scrum, XP, Lean, and Kanban, By Andrew Stellman, Jennifer Greene, 2015, O Reilly.

Online References:

1. www.javatpoint.com, https://www.javatpoint.com/devops
2. www.guru99.com, https://www.guru99.com/devops-tutorial.html
3. www.tutorialspoint.com, https://www.tutorialspoint.com/devops_tutorials.htm
4. www.simplilearn.com, https://www.simplilearn.com/tutorials/devops-tutorial
5. www.edureka.co, https://www.edureka.co/blog/devops-tutorial
6. https://www.jenkins.io, https://www.jenkins.io/doc/tutorials/
7. https://github.com, https://github.com/learn/devops
8. www.dotnettricks.com, https://www.dotnettricks.com/learn/devops

List of Experiments:

Devops Laboratory (DJ19AML503)	
Sr. No	**Suggested experiments**
1	Write code for a simple user registration form for an event. To Study DevOps: Principles, Practices, and DevOps Engineer Role and Responsibilities.
2	To carry out Version Control System / Source Code Management, install git and create a GitHub account.
3	To Perform various GIT operations on local and Remote repositories using GIT Cheat-Sheet.
4	Continuous Integration, install and configure Jenkins with Maven/Ant/Gradle to setup a build Job.
5	To Build the pipeline of jobs using Maven / Gradle / Ant in Jenkins, create a pipeline script to Test and deploy an application over the tomcat server.
6	To Setup and Run Selenium Tests in Jenkins Using Maven.
7	To study Docker Architecture and Container Life Cycle, install Docker and execute docker commands to manage images and interact with containers.
8	To study Dockerfile instructions, build an image for a sample web application using Dockerfile.
9	To install and Configure Pull based Software Configuration Management and provisioning tools using Puppet.
10	To perform Software Configuration Management and provisioning using Puppet Blocks (Manifest, Modules, Classes, Function).

Minimum eight experiments from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Prepared by Checked by Head of the Department Principal

Program: Artificial Intelligence & Machine Learning	**T.Y. B.Tech.**	**Semester: V**
Course: Programming Laboratory-III (Full Stack Development using Python) DJ19AML504		

Prerequisite: Programming in python

Course Objective:

1. To create routes and views to handle different HTTP requests.
2. Gain knowledge of working with databases, including connecting to databases, performing CRUD operations
3. Understand the fundamental concepts of backend development and the role of Python in building backend systems.

Course Outcomes: On completion of the course, the students will be able to:

1. Analyze and evaluate different approaches for building front-end web applications.
2. Utilize the Flask framework to build a web server that can handle HTTP requests
3. Apply the knowledge of different libraries to establish connections with databases and perform CRUD operations.
4. Implement API endpoints, handle incoming requests, validate data, and generate suitable responses.

Programming Laboratory-III (Full Stack Development using Python) DJ19AML504		
Unit	**Description**	**Duration**
1	**HTML, CSS, and JavaScript for web development:** **Introduction to HTML and CSS:** Understanding HTML tags, attributes, and elements, Creating HTML documents and basic structure, Working with common HTML elements such as headings, paragraphs, lists, links, images, forms, and tables, Introduction to CSS and its syntax, CSS selectors, properties, and values, Styling HTML elements with CSS, including text formatting, colors, backgrounds, margins, padding, and borders, Creating layouts with CSS using floats, flexbox, and grid, Creating responsive and accessible layouts with CSS **Introduction to JavaScript**: variables, data types, operators, control structures, and functions, Document Object Model (DOM) and how JavaScript interacts with HTML elements, Handling events with JavaScript, such as button clicks, form submissions, and keypresses, Manipulating DOM elements dynamically with JavaScript, Introduction to error handling, debugging in JavaScript coding, Building interactive web pages with JavaScript	4

2	**Introduction to Flask:** Flask micro web framework for Python, Flask Application Structure, application file, templates directory, and static files directory, Setting up a basic Flask application with minimal configuration. **Creating Routes and Views in Flask:** Understanding Routes, Creating Routes: Defining routes in Flask using decorators, including handling different HTTP methods (GET, POST, etc.) for different routes, Creating Views, view functions in Flask, handle requests from routes and return responses to the client	4
3	**Handling User Input and Forms with Flask:** retrieving data from request objects and processing it, Flask's built-in form handling functionality to validate and process form data, using third-party libraries for form validation, Flask's flash messages to display feedback or error messages. **Using Templates to Render Dynamic Content in Flask:** Introduction to Jinja2, creating templates in Flask using Jinja2 syntax, working with template inheritance, template variables, and control structures, Using Flask's template rendering functionality to render dynamic content in HTML pages, passing data from views to templates.	4
4	**Introduction to Django:** Understanding Django the model-view-controller (MVC) architectural pattern, Django project directory, settings file, URL configuration, and application structure, Creating a Django Project, structure of the data in a Django web application and map them to database tables, Defining models in Django using Python classes, including fields, relationships, and model methods, Django's built-in database migration functionality to create and apply database schema changes **Building Views and Templates in Django:** Understanding Views , handle HTTP requests and generate HTTP responses in Django, Creating view functions in Django that handle requests from URLs, handling different HTTP methods, processing data, and rendering responses, Templates in Django, Creating templates in Django, working with template tags, filters, and template inheritance, Using Django's template rendering functionality to render dynamic content in HTML pages. **Forms and User Authentication in Django:** Django's built-in form handling, Creating forms in Django, validating form data, and handling form submissions, rendering form fields in templates, user authentication.	6
5	**Database in python:** Using SQL with python, retrieving rows from a table, inserting rows into a table, deleting rows from a table, updating rows in a table, creating database tables through python, Exception handling in databases. **Networking**: Protocols, server-client architecture, TCP/IP and UDP communication, Network Fundamentals and Socket Programming, Client-side programming, Internet Data Handling, Web Programming.	4

6	**Building RESTful APIs with Python:** Understanding RESTful architecture, Using Flask or Django to build RESTful APIs, including handling requests, routing, authentication, and authorization, Serializing and deserializing data in Python using formats like JSON or XML for API communication, Writing tests for APIs to ensure their functionality and documenting APIs using tools like Swagger or Postman.	4
	TOTAL	26

Text Books:

1. Learn Web Development with Python: Get hands-on with Python Programming and Django web development by Fabrizio Romano , Gaston C. Hillar, Arun Ravindran(2018)
2. Advanced Python Programming by Dr. Gabriele Lanaro, Quan Nguyen, Sakis Kasampalis(2019)
3. “Mastering Flask Web Development” Second Edition (2018) by Daniel Gaspar, Jack Stouffer

Reference Books:

1. "Python Coockbook" by Brian K. Jones and David M. Beazley.(2013)
2. Flask Web Development 2e: Developing Web Applications with Python by Miquel Grinberg (2018)
3. “Django for beginners: Build websites with Python & Django” William Vincent(2018).

Useful Links:

1. https://www.coursera.org/learn/django-build-web-apps
2. https://www.coursera.org/learn/developing-applications-with-sql-databases-and-django
3. https://www.coursera.org/projects/django-for-beginners-creating-applications-and-views
4. https://www.coursera.org/specializations/django
5. https://www.coursera.org/projects/python-flask
6. https://www.coursera.org/projects/web-development-in-flask-build-your-first-website
7. **https://www.w3schools.com/html/html_css.asp**

List of Experiments:

Programming Laboratory-III (Full Stack Development using Python) DJ19AML504	
Sr. no.	**Name of Experiment**
1	Creating an HTML webpage
2	Adding styles to an HTML webpage with CSS and Creating responsive and accessible layouts with CSS.
3	Write a Python Program to work with databases in Python to perform operations such as a. Connecting to database b. Creating and dropping tables c. Inserting and updating into tables.
4	Write a Python program to create server-client and exchange basic information
5	Set up a Flask development environment: Install Flask and set up a virtual environment for development and create a simple Flask application: Build a simple Flask application with a single route that returns a "Hello World" message.
6	Add routes and views to the Flask application: Create additional routes and views in the Flask application to handle different URLs and HTTP methods.
7	Using templates in Flask: Use Jinja2 templates to create dynamic content in the Flask application
8	Handling user input with forms: Create a form in the Flask application to accept user input.
9	Set up a Django development environment: Install Django, create a new Django project, and set up a virtual environment.
10	Creating a Django app: Learn how to create a new app within a Django project, configure the app's settings, and add the app to the project's URL configuration.
11	Building views and templates: Build a set of views and templates for your app, including a homepage, an about page, and a detail page for the blog post model you created
12	Handling user authentication and authorization: Add user authentication to your app, allowing users to create accounts, log in, and log out.
13	Building RESTful APIs with Django

Prepared by checked by Head of the Department Principal

Program: Artificial Intelligence & Machine Learning	**T.Y.** **B.Tech.**	**Semester:** **V**
Course: Cloud Computing (DJ19AMEC5011)		
Course: Cloud Computing Laboratory (DJ19AMEL5011)		

Course Objective:

1. To familiarize with cloud computing terminologies and industry standards.
2. To gain Hands-on experience in virtualizing resources in a cloud environment.
3. To introduce the best practices for securing cloud services and managing user access.
4. To Introduce distributed database design principles and techniques.

Course Outcomes: On completion of the course, the students will be able to:

1. Understand the fundamental concepts and principles of cloud computing.
2. Use virtualization technologies and tools.
3. Understand the concept of identity and access management in cloud computing.
4. Utilize big data processing frameworks such as Hadoop, Spark, and Flink to process and analyze large datasets..

Cloud Computing (DJ19AMEC5011)		
Unit	**Description**	**Duration**
1	**Introduction to Cloud Computing:** Overview of cloud computing, Fundamentals of cloud computing ecosystem, cloud computing characteristics, Components of cloud computing, peer-to-peer, client-server, grid computing, Cloud Architecture, Introduction to distributed computing, need of distributed computing, Introduction to Parallel computing, Parallel computing platforms.	**4**
2	**Cloud Services and Deployment Models:** Infrastructure as a service (IaaS), Platform as a service (PaaS), Software as a service (SaaS), NIST Model for cloud, Public cloud, Private cloud, Hybrid cloud, Community cloud. Computing services, Storage services, Networking services, Database services. **Virtualization:** Full Virtualization, Para-Virtualization, Hardware/ Memory Virtualization, KVM architecture. **Cloud Infrastructure mechanism**: Logical network perimeter, virtual server, cloud storage devices, cloud usage monitor, resource replication, readymade environment	**8**
3	**Cloud Programming:** Programming Support for Amazon EC2: Amazon S3, EBS and Simple DB	**7**

	Programming Support for Google Apps engine: GFS, Big 4 Tables, Googles NO SQL System, Chubby, Google Distributed Lock Service	
4	**Cloud Security:** Securing the Cloud, The security boundary, Security service boundary, Security mapping. Host security for SaaS, 4 PaaS and IaaS, **Data Security:** Data Security Concerns, Data Confidentiality and Encryption, Data Availability, Data Integrity, Cloud Storage Gateways, Cloud Firewall **Establishing Identity and Presence: Identity and Access Management (IAM)**	8
5	**Serverless Computing**: overview of serverless computing, serverless architecture and design patterns, Microservices and event-driven architectures, Functions and triggers Serverless Platforms and Providers: Overview of serverless platforms and providers (AWS Lambda, Google Cloud Functions, Azure Functions, etc.) Features and capabilities of serverless platforms, Overview of serverless security and governance	6
6	**Cloud Applications:** MapReduce: Paradigm, Programming Model, Applications, Scheduling, Fault-Tolerance, Implementation Overview, Examples Introduction to Spark: Resilient Distributed Datasets (RDDs), RDD Operations, Spark applications: Page Rank Algorithm, GraphX, GraphX API, GraphX working **Introduction to Kafka**: What is Kafka, Use cases for Kafka, Data model, Architecture, Types of messaging systems, Importance of brokers	6

Text Book:

1. "Cloud Computing Bible" by Barrie Sosinsky (2018)
2. "Cloud Computing" by Mehul Mahrishi Kamal Kant Hiran, Ruchi Doshi, Dr. Fagbola Temitayo.(2019)

Reference books:

1. Judith Hurwitz ,"Cloud Computing for Dummies" , Wiley Publication.(2020)
2. "Cloud Security and Privacy: An Enterprise Perspective on Risk and Compliance" by Tim Mather, Subra Kumaraswamy, Shahed Latif.(2019)

Useful Links:

1. https://www.coursera.org/learn/introduction-to-cloud
2. https://www.coursera.org/projects/googlecloud-introduction-to-cloud-dataproc-hadoop-and-spark-on-google-clou-j3jfl
3. https://www.coursera.org/learn/ibm-cloud-essentials
4. https://www.coursera.org/learn/microsoft-azure-cloud-services

List of Experiments

Cloud Computing Laboratory (DJ19AMEL5011)	
Sr. no.	**Name of Experiment**
1	Virtualization: Hosted Virtualization and Bare Metal Virtualization
2	Creating a virtual machine on a public cloud platform: how to create a virtual machine on a public cloud platform like Amazon Web Services (AWS) or Microsoft Azure and configure it to run a basic application.
3	Host a Static Website on cloud.
4	Configure Identity and Access Management (IAM) for secure access.
5	Create and migrate relational database on cloud.
6	Study different Infrastructure as a service (IaaS), Platform as a service (PaaS), Software as a service (SaaS) application.
7	Install Google App Engine. Create simple web applications using python/java
8	Building a simple web application on the cloud: how to build a simple web page using the cloud infrastructure mechanism, including a virtual server, cloud storage, and networking services.
9	Automate Infrastructure Development using IaaS.
10	Implement serverless architecture and configure notification services.
11	Cloud case studies and future trends Case studies of successful cloud deployments in AI/ML Emerging cloud technologies and trends Ethical and legal considerations in cloud computing

Prepared by checked by Head of the Department Principal

Program: Artificial Intelligence & Machine Learning	T.Y. B.Tech.	Semester: V
Course: Advanced Data Structures and Algorithm (DJ19AMEC5012)		
Course: Advanced Data Structures and Algorithm Laboratory (DJ19AMEL5012)		

Pre-requisite: --

1. Knowledge Any Programming Language
2. Data structures and Analysis
3. Discrete mathematics
4. Analysis of Algorithm and Basics of Machine Learning

Objectives:

1. To provide conceptual and practical knowledge of Advance Data Structures and Algorithm

Outcomes: On completion of the course, the learner will be able to:

1. Analyze the chosen algorithm.
2. Choose appropriate data structure and algorithm for given problem statement.
3. Apply best suitable algorithms for a specific task.
4. Classify the algorithms based on the complexity
5. Design the algorithm.

Advanced Data Structures and Algorithm (DJ19AMEC5012)		
Unit	**Description**	**Duration**
1	**Analysis of Algorithm Based on Time:** i. **Asymptotic notations:** Omega, Theta, Big-O, Small-o, small Omega and Tilde ii. **Amortized Analysis:** Aggregate Method, Accounting Method, Potential Method (for Stack data structure) iii. **Probabilistic and Randomized Algorithm:** Probabilistic approach to algorithm and Randomized Analysis, Indicator Random Variable (IRV), Analysis of Hiring Problem **Complexity Analysis of Machine Learning Algorithms:** iv. Training Time Complexity and Testing Time Complexity v. Train/Test Complexity of Linear Regression vi. Train/Test Complexity of Naïve Bayes Classifier	**06**
2	**Advanced Data Structures:** i. **Balanced Search Trees:** Red-Black Tree, Tango Tree, 2-3 Tree, B+ Tree, Splay Tree	**14**

	ii. **Spatial Data Structure:** KD Tree, R Tree iii. **Probabilistic Data Structure:** Bloom filter, LogLog and HyperLogLog, Count Min sketch, MinHash with Machine Learning context (Vector Representation) iv. **Functional Data Structures:** Binomial Tree, Binomial Heap	
3	**Algorithms for Machine Learning:** i. **Dimension Reduction Algorithms:** Rank-k approximation ii. **Continuous Algorithms:** Online gradient descent algorithm iii. **Online Learning:** Concept of Experts and Bandit (e.g. Zero-Sum game)	06
4	**Graph Based Algorithms:** i. **Flow Network Introduction:** Residual Network, Augmenting Path, Ford-Fulkerson Method, Edmonds-Karp Method, Push-Relable Algorithm ii. **Bipartite Matching:** Maximum Bipartite Matching, Red-Blue Matching, Micali Vaziarni Algorithm	05
5	**Classification of Algorithms:** i. **Algorithm Classes:** P, NP, NP Hardness and NP Completeness ii. **Np Completeness Proofs:** Satisfiability (3 sat), Reducibility, Cook's Theorem, Traveling Salesman Problem iii. **Approximation Algorithms:** Vertex Cover Problem, Travelling Salesman problem iv. **Special Topic:** Turing Machine Halting Problem (time and space bounds, nondeterminism)	06
6	**Computational Algorithms:** **Computational Geometry:** Line Segment Properties, Convex Hull Graham's scan algorithm	02
	Total	39

Books Recommended:

Text books:

1. Introduction to Algorithms by Thomas H Cormen, Charles E. Leiserson, Ronald L Rivest, Clifford Stein, Third Edition, 2009.
2. Design and analysis of algorithms by S. Sridhar Oxford First Edition, 2014
3. Horowitz, Sahani and Rajsekaran, ―Fundamentals of Computer Algorithms, Galgotia, 2nd Edition,1998
4. Harsh Bhasin, Algorithms Design and Analysis, Oxford, 1st Edition, 2015.
5. Giuseppe Bonaccorso, Machine Learning Algorithms" by Packt, 2019

Reference Books:

1. Rajeev Motwani, Prabhakar Raghavan, Randomized Algorithm, Cambridge University, 2004
2. S. K. Basu, Design Methods and Analysis of Algorithm, PHI, 2005
3. Vijay V. Vajirani, Approximation Algorithms, Springer, 2003
4. Computational Complexity, Stanford University, 2010
5. Jason Brownlee, “Master Machine Learning Algorithms”, by Machine Learning Mastery, 2020

Useful Links:

1. https://levelup.gitconnected.com/train-test-complexity-and-space-complexity-of-linear-regression-26b604dcdfa3
2. https://7-hiddenlayers.com/time-complexities-of-ml-algorithms/
3. https://towardsdatascience.com/importance-of-understanding-the-complexity-of-a-machine-learning-algorithm-9d0532685982
4. https://www.thekerneltrip.com/machine/learning/computational-complexity-learning-algorithms/
5. https://medium.com/ai-ml-at-symantec/ai-ml-security-pro-tips-understanding-minhash-in-a-security-context3dd0dd2ffe8#:~:text=MinHash%20is%20not%20typically%20thought,commonly%20used%20in%20machine%20learning.
6. http://ccf.ee.ntu.edu.tw/~yen/courses/ds17/chapter-6d.pdf
7. https://betterprogramming.pub/compressing-puppy-image-using-rank-k-approximation-a-doodle-explanation-c19de5dfd951
8. https://parameterfree.com/2019/09/11/online-gradient-descent/

List of Experiments:

Advanced Data Structures and Algorithm Laboratory (DJ19AMEL5012)	
Exp.	**Suggested experiments**
1	Experiment on Amortized Analysis.
2	Experiment on Randomized Algorithms (Randomized Quick Sort)
3	Experiment on Advanced Data Structure (Red-black Tree Operations)
4	Experiment on Advanced Data Structure (B+ Tree Operations)
5	Experiment on Advanced Data Structure (MinHash Vector Representation)
6	Experiment on Machine Learning Algorithms (Ford Fulkerson Method)

7	Experiment on Machine Learning Algorithms (Ford Fulkerson Method)
8	Experiment on Graph Based Algorithms (Ford Fulkerson Method)
9	Experiment on Graph Based Algorithms (Push Relable Algorithm)
10	Experiment on Computational Geometry Algorithms (Graham Scan)
11	Experiment on Online Algorithms (K-Server algorithm)
12	Experiment on Approximation Algorithms (Vertex Cover)
13	Development of new algorithm by students based on any one topic of above mentioned syllabus

Minimum eight experiments from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Prepared by checked by Head of the Department Principal

Program: Artificial Intelligence & Machine Learning	**T.Y. B.Tech.**	**Semester: V**
Course: Recommendation systems (DJ19AMEC5013)		
Course: Recommendation systems Laboratory (DJ19AMEL5013)		

Pre-requisite: Statistics for Data Science, and Machine Learning – I.

Course Objectives:

1. To provide students with the basic concepts of Recommender Systems, design space, trade-offs and its application in various domain.

Outcomes: On completion of the course, learner will be able to:

2. Compare different types of Recommender Systems.
3. Understand various issues related to recommender system development.
4. Design a recommender system for a given problem.
5. Relate data collected from a recommender system to understand user preferences and/or behavior.

Recommendation systems (DJ19AMEC5013)		
Unit	**Description**	**Duration**
1	**Introduction to Recommender Systems** Introduction to Recommendation engine, Framework of recommendation systems, Domain, Purpose, Context, Personalization: Personalized vs. Non-Personalized, Semi/Segment - Personalized, Privacy: users data and trustworthiness Recommender Systems Function and Techniques, Conversational Systems, Issues working with RSs data sets: The cold-start problem.	**08**
2	**Collaborative filtering-based Recommender System** Understanding ratings and rating data, User-based nearest-neighbour recommendation: Similarity Function, User-Based Algorithms, Item-based nearest neighbour recommendation: Similarity Function, Item-Based Algorithms, Comparing User-Based and Item-Based recommendations, data drift and concept drift.	**06**
3	**Content-based Recommender System:** Architecture of Content-based Systems, Advantages and Drawbacks, Content representation and content similarity, Item profiles, discovering features of data, obtaining item features from tags, representing item profiles, Learning User Profiles and Filtering, Similarity-based retrieval, Classification algorithms, Knowledge base recommendation: Knowledge representation and reasoning, Case based recommenders.	**06**

4	**Neighbourhood-based Recommendation System:** Neighbourhood-based Recommendation, advantages, User-based Rating Prediction, User-based Classification, Item-based Recommendation, Rating Normalization, Similarity Weight Computation, Neighbourhood Selection	**06**
5	**Constraint-based Recommendation System:** Development of Recommender Knowledge Bases, User Guidance in Recommendation Processes, Calculating Recommendations. **Context-Aware Recommender Systems** Trust Context in Recommender Systems, Modeling Contextual Information in Recommender Systems. Paradigms for Incorporating Context in Recommender Systems: Contextual Pre-Filtering, Contextual Post-Filtering, Contextual Modeling, Combining Multiple Approaches, Issues in Context-Aware Recommender Systems.	**07**
6	**Evaluating Recommendation System:** Evaluation Paradigms, General Goals of Evaluation Design, Design Issues in Offline Recommender Evaluation, Online Recommender evaluation techniques, Case Study of the Netflix Prize Data Set, Segmenting the Ratings for Training and Testing, Hold-Out, Cross-Validation, Concept of Bandit Algorithm to balance exploration and exploitation. Accuracy Metrics Evaluation: RMSE versus MAE, Impact of the Long Tail, Evaluating Ranking via Correlation, Evaluating Ranking via Utility, Evaluating Ranking via Receiver Operating Characteristic	**06**
	TOTAL	**39**

Books Recommended:

Textbooks:

1. C.C. Aggarwal, “Recommender Systems: The Textbook”, Springer, 1 st Edition, 2016.
2. Jannach D., Zanker M. and FelFering A., “Recommender Systems: An Introduction”, Cambridge University Press, 1st Edition, 2011.
3. Kim Falk, “Practical Recommender Systems”, Manning, 1st Edition, 2019
4. “Hands-On Recommendation Systems with Python: Start building powerful and personalized, recommendation engines with Python” by Rounak Banik , 2018.

Reference Books:

1. M.D. Ekstrand, J.T. Riedl, J.A. Konstan, “Collaborative filtering recommender systems”, Now publishers, 1 st Edition, 2011.
2. J. Leskovec, A. Rajaraman and J. Ullman, “Mining of massive datasets”, Cambridge, 2 nd Edition, 2012.
3. Rounak Banik, “Hands-On Recommendation Systems with Python: Start building”, Ingram short title, 2018

4. P. Pavan Kumar, S. Vairachilai, Sirisha Potluri, “Recommender Systems: Algorithms and Applications”, CRC Press, 1st edition, 2021.

Web Links:

1. Udemy course on Recommender Systems and Deep Learning in Python: https://realpython.com/build-recommendation-engine-collaborative-filtering
2. Coursera course on Recommender Systems Specialization: https://www.coursera.org/specializations/recommender-systems

List of Experiments

Recommendation systems Laboratory (DJ19AMEL5013)	
Sr. No.	**Title of the Experiment**
1	Build a Recommendation Engine with Item-Based Collaborative Filtering.
2	Build Content-based recommendation engine on different datasets.
3	Build recommender system using association rule mining.
4	Implement Recommendation System using K-Nearest Neighbors
5	Build Context-Aware Recommender Systems.
6	Build Constraint-based Recommenders.
7	Implement knowledge-based recommender system.
8	Evaluate the recommendation system with evaluation matrix.
9	Compare the performance of different recommender systems
10	Mini Project.

Prepared by checked by Head of the Department Principal

Program: Artificial Intelligence & Machine Learning	T.Y B.Tech	Semester: V
Course: Environmental Studies (DJ19A5)		

Pre-requisite: Interest in Environment and its impact on Human

Objectives:

1. Understand environmental issues such as depleting resources, pollution, ecological problems and the renewable energy scenario.
2. Familiarise environment related legislation

Outcomes: On completion of the course, learner will be able to:

1. Understand how human activities affect environment
2. Understand the various technology options that can make a difference

Environmental Studies (DJ19A5)		
Unit	**Description**	**Duration**
1	**Social Issues and Environment:** Ecological footprint and Carrying Capacity, Depleting nature of Environmental resources such as soil, water minerals and forests, Carbon emissions and Global Warming.	**4**
2	**Technological Growth for Sustainable Development:** Social, Economical and Environmental aspects of Sustainable Development, Renewable Energy Harvesting, Concept of Carbon credit, Green Building, Power and functions of Central Pollution Control Board and State Pollution Control Board.	**4**
3	**Green Technology:** History, Agenda, and Challenges Ahead. Sustainable Cloud Computing, and Risk Management, Sustainable Software Design, Data Center Energy Efficiency, Thin-Client and Energy Efficiency.	**5**

Books Recommended:

Text books:

1. Environmental Studies From Crisis to Cure, R. Rajagopalan, 2012
2. Textbook for Environmental Studies For Undergraduate Courses of all Branches of Higher Education, Erach Bharucha
3. Green Information Technology A Sustainable Approach, Mohammad Dastbaz, Colin Pattinson, Babak Akhgar, Morgan and Kaufman, Elsevier, 2015.

Reference Books:

1. Information Technologies in Environmental Engineering: New Trends and Challenges, Paulina Golinska, Marek Fortsch, Jorge Marx-Gómez, Springer, 2011.

Program: Artificial Intelligence & Machine Learning	**T.Y B.Tech**	**Semester: V**
Course: Innovative Product Development-III (DJ19ILL1)		

Objectives:

1. To acquaint the students with the process of identifying the need (considering a societal requirement) and ensuring that a solution is found out to address the same by designing and developing an innovative product.
2. To familiarize the students with the process of designing and developing a product, while they work as part of a team.
3. To acquaint the students with the process of applying basic engineering fundamentals, so as to attempt at the design and development of a successful value added product.
4. To inculcate the basic concepts of entrepreneurship and the process of self-learning and research required to conceptualise and create a successful product.

Outcome:

Learner will be able to:

1. Identify the requirement for a product based on societal/research needs.
2. Apply knowledge and skills required to solve a societal need by conceptualising a product, especially while working in a team.
3. Use standard norms of engineering concepts/practices in the design and development of an innovative product.
4. Draw proper inferences through theoretical/ experimental/simulations and analyse the impact of the proposed method of design and development of the product.
5. Develop interpersonal skills, while working as a member of theteam or as theleader.
6. Demonstrate capabilities of self-learning as part of the team, leading to life-long learning, which could eventually prepare themselves to be successful entrepreneurs.
7. Demonstrate product/project management principles during the design and development work and also excel in written (Technical paper preparation) as well as oral communication.

Guidelines for the proposed product design and development:

- Students shall form a team of 3 to 4 students (max allowed: 5-6 in extraordinary cases, subject to the approval of the department review committee and the Head of the department).
- Students should carry out a survey and identify the need, which shall be converted into conceptualization of a product, in consultation with the faculty supervisor/head of department/internal committee of faculty members.

- Students in the team shall understand the effective need for product development and accordingly select the best possible design in consultation with the faculty supervisor.
- Students shall convert the best design solution into a working model, using various components drawn from their domain as well as related interdisciplinary areas.
- Faculty supervisor may provide inputs to students during the entire span of the activity, spread over 2 semesters, wherein the main focus shall be on self-learning.
- A record in the form of an activity log-book is to be prepared by each team, wherein the team can record weekly progress of work. The guide/supervisor should verify the recorded notes/comments and approve the same on a weekly basis.
- The design solution is to be validated with proper justification and the report is to be compiled in a standard format and submitted to the department. Efforts are to be made by the students to try and publish a technical paper, either in the institute journal, “Techno Focus: Journal for Budding Engineers” or at a suitable publication, approved by the department research committee/ Head of the department.
- The focus should be on self-learning, capability to design and innovate new products as well as on developing the ability to address societal problems. Advancement of entrepreneurial capabilities and quality development of the students through the year long course should ensure that the design and development of a product of appropriate level and quality is carried out, spread over two semesters, i.e. during the semesters V and VI.

Guidelines for Assessment of the work:

- The review/ progress monitoring committee shall be constituted by the Head of the Department. The progress of design and development of the product is to be evaluated on a continuous basis, holding a minimum of two reviews in each semester.
- In the continuous assessment, focus shall also be on each individual student’s contribution tothe team activity, their understanding and involvement as well as responses to the questions being raised at all points in time.
- Distribution of term work marks during the subsequent semester shall be as given below:
  - Marks awarded by the supervisor based on log-book10
  - Marks awarded by review committee 10
  - Quality of the write-up 05

In the last review of the semester VI, the term work marks will be awarded as follows.

- Marks awarded by the supervisor (Considering technical paper writing) 15
- Marks awarded by the review committee 10

Review/progress monitoring committee may consider the following points during the assessment.

- In the semester V, the entire design proposal shall be ready, including components/system selection as well as the cost analysis. Two reviews will be conducted based on the presentation given by the student's team.

- First shall be for finalisation of the product selected.

Second shall be on finalisation of the proposed design of the product.

- In the semester VI, the expected work shall be procurement of components/systems, building of the working prototype, testing and validation of the results based on work completed in semester III.

- First review is based on readiness of building the working prototype.
- Second review shall be based on a presentation as well as the demonstration of the working model, during the last month of semester IV. This review will also look at the readiness of the proposed technical paper presentation of the team.

The overall work done by the team shall be assessed based on the following criteria;

1. Quality of survey/ need identification of the product.
2. Clarity of Problem definition (design and development) based on need.
3. Innovativeness in the proposed design.
4. Feasibility of the proposed design and selection of the best solution.
5. Cost effectiveness of the product.
6. Societal impact of the product.
7. Functioning of the working model as per stated requirements.
8. Effective use of standard engineering norms.
9. Contribution of each individual as a member or the team leader.
10. Clarity on the write-up and the technical paper prepared.

- The semester reviews (V and VI) may be based on relevant points listed above, as applicable.

Guidelines for Assessment of Semester Reviews:

- The write-up should be prepared as per the guidelines given by the department.
- The design and the development of the product shall be assessed through a presentation and demonstration of the working model by the student team to a panel of Internal and External Examiners, preferably from industry or any research organisations having an experience of more than five years, approved by the Head of the Institution. The presence of the external examiner is desirable only for the 2^nd^ presentation in semester IV.Students are compulsorily required to present the outline of the technical paper prepared by them during the final review in semester VI.