PROGRAM BOOK FOR LONG – TERM INTERNSHIP

AP STATE COUNCIL OF HIGHER EDUCATION

(A STATUTORY BODY OF GOVERNMENT OF ANDHRA PRADESH)

A LONG TERM INTERNSHIP

ON

"MACHINE LEARNING WITH PYTHON INTERNSHIP"

Submitted in partial fulfillment of requirements for the

award of the degree of

MASTER OF COMPUTER APPLICATIONS

Submitted by

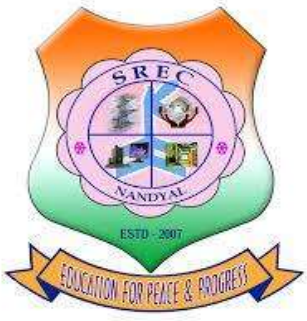
JANGITI HARITHA

(23X51F0027)

Under the Esteemed Guidance of

Mr. T. YESURAJU, MCA.

Assistant Professor, Dept of MCA



DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

SANTHIRAM ENGINEERING COLLEGE::NANDYAL

(AUTONOMOUS)

(Approved by AICTE : New Delhi, Affiliated to J.N.T.University, Ananthapuramu. A.P.)

Accredited by NAAC (Grade-A), ISO 9001:2015 Certified Institution,

2(f) and 12(b) recognition by UGC Act, 1956

YEAR: 2024-2025

SANTHIRAM ENGINEERING COLLEGE::NANDYAL

(AUTONOMOUS)

(Approved by AICTE : New Delhi, Affiliated to JNT University, Ananthapuramu. A.P.)
Accredited by NAAC (Grade-A), ISO 9001:2015 Certified Institution,
2(f) and 12(b) recognition by UGC Act, 1956

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the Long – Term Internship report on "**MACHINE LEARNING WITH PYTHON Internship**" is a Bonafide work of **JANGITI HARITHA (23X51F0027),** MCA IV - Semester in the Department of **MASTER OF COMPUTER APPLICATIONS,** Santhiram Engineering college (Autonomous), Nandyal, Affiliated to JNTUA, Anantapur, during the academic year 2024-25, in fulfilment of the requirement for the award of the degree of Master of Computer Applications.

Signature of the Guide
Mr. T. YESURAJU, MCA
Assistant professor, Dept. of MCA,
SREC, Nandyal.

Signature of the Internship Coordinator
Mrs. S. SHAKILA BHANU, MCA
Assistant Professor, Dept. of MCA
SREC, Nandyal.

Signature of HOD
Mr. M. IMDAD ALI, MCA, M.Tech.
Associate Professor
HOD of MCA,
SREC, NANDYAL.

SANTHIRAM ENGINEERING COLLEGE::NANDYAL

(AUTONOMOUS)

(Approved by AICTE : New Delhi, Affiliated to JNT University, Ananthapuramu. A.P.)
Accredited by NAAC (Grade-A), ISO 9001:2015 Certified Institution,
2(f) and 12(b) recognition by UGC Act, 1956

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

Internship on MACHINE LEARNING WITH PYTHON

Name of the College : SANTHIRAM ENGINEERING COLLEGE

Department : MASTER OF COMPUTER APPLICATIONS

Name of the Faculty Guide : Mr. T. YESURAJU, Asst prof.

Duration of the Internship : FROM DECEMBER TO MARCH

Name of the student : JANGITI HARITHA

Program of the Study : MCA

Year of Study : 2024 - 2025

Register Number : 23X51F0027

Date of submission : 24 - 03 - 2025

Signature of the Student

Signature of HOD

Program Book for

Long-Term Internship

Name of the Student	: JANGITI HARITHA
Name of the College	**:** SANTHIRAM ENGINEERING COLLEGE
Registration Number	**:** 23X51F0027
Period of Internship	**:** From: DEC 2024 to MARCH 2025
Name & Address of the Intern Organization	**:** Talent Shine India Pvt.ltd.

An Internship Report On

(MACHINE LEARNING WITH PYTHON)

Submitted in accordance with the requirement for the degree of MCA

Name of the College **:** SANTHIRAM ENGINEERING COLLEGE

Department **:** MASTER OF COMPUTER APPLICATIONS

Name of the Faculty Guide **:** Mr. T. YESURAJU, Asst prof.

Duration of the Internship **:** FROM DECEMBER TO MARCH

Name of the student : JANGITI HARITHA

Program of the Study **:** MCA

Year of Study **:** 2024-2025

Register Number **:** 23X51F0027

Date of submission **:** 24 - 03 - 2025

Student's Declaration

I am JANGITI HARITHA, a student of MCA Program, Reg.no.23X51F0027 of the MASTER OF COMPUTER APPLICATIONS, SANTHIRAM ENGINEERING COLLEGE do hereby declare that I have completed the Long-Term Internship from DEC 2024 to MARCH 2025 in Talent Shine India Pvt. Ltd. by APSCHE under the Faculty Guidance of Mr. T. YESURAJU, Department of MASTER OF COMPUTER APPLICATIONS, SANTHIRAM ENGINEERING COLLEGE.

(Signature and Date)

Endorsements

Faculty Guide

Head of the Department

Principal

INTERNSHIP CERTIFICATE

ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

(A Statutory Body of the Government of A.P)

CERTIFICATE OF COMPLETION

This is to Certify that Ms/Mr. JANGITI HARITHA
23X51F0027 of SANTHIRAM ENGINEERING COLLEGE(A), NANDYAL
has Successfully Completed long-term internship from 09/12/2024
to 20/03/2025 on MACHINE LEARNING WITH PYTHON organized by
Talent Shine India Pvt.Ltd in Collaboration with Andhra Pradesh
State Council of Higher Education .

Certificate Number : TSI24LTSR0024

Date: 20/03/2025

Place: Visakhapatnam

Director

ACKNOWLEDGEMENT

I manifest our heartier thankfulness pertaining to your contentment over our project guide **Mr. T. Yesuraju**, Assistant Professor of Master of Computer Applications department, with whose adroit concomitance the excellence has been exemplified in bringing out this project to work with artistry.

I express my gratitude to **Mr.M. Imdad Ali garu,** Head of the Department of Master of Computer Applications, all the Teaching Staff Members of the MCA departments of Santhiram Engineering College of Engineering and Technology for providing continuous encouragement and cooperation at various steps of our project.

Involuntarily, I am perspicuous to divulge my sincere gratitude to our Principal, **Dr. M.V. Subramanyam garu,** who has been observed posing valiance in abundance towards our individuality to acknowledge my project work tangentially.

At the outset I thank our Honourable Chairman **Dr. M. Santhi Ramudu garu**, for providing us with exceptional faculty and moral support throughout the course.

Finally I extend my sincere thanks to all the non- teaching Staff Members of MCA Department who have co-operated and encouraged us in making my project successful.

Whatever one does, whatever one achieves, the first credit goes to the Parents, be it not for their love and affection, nothing would have been responsible. I see in every good that happens to me their love and blessings.

By

JANGITI HARITHA

(23X51F0027)

INDEX

FIGURES

ABSTRACT

Present day computer applications require the representation of huge amount of complex knowledge and data in programs and thus require tremendous amount of work. Our ability to code the computers falls short of the demand for applications. If the computers are endowed with the learning ability, then our burden of coding the machine is eased (or at least reduced). This is particularly true for developing expert systems where the "bottle-neck" is to extract the expert's knowledge and feed the knowledge to computers. The present day computer programs in general cannot correct their own errors or improve from past mistakes, or learn to perform a new task by analogy to a previously seen task. The area of Machine Learning deals with the design of programs that can learn rules from data, adapt to changes, and improve performance with experience. In addition to being one of the initial dreams of Computer Science, Machine Learning has became crucial as computers are expected to solve increasingly complex problems and become more integrated into our daily lives. So, here we study how the Machine Learning take place, what are the methods, discuss various projects (Implemented during training) applications, present and future status of machine learning.

1.INTRODUCTION

Machine Learning (ML) is a core component of modern artificial intelligence that allows systems to learn from data and improve their performance without being explicitly programmed. With the growing availability of data and computational power, machine learning has found applications in various fields such as healthcare, finance, marketing, and autonomous systems. Python has become the most popular language for developing machine learning models due to its simplicity, readability, and a rich ecosystem of libraries like scikit-learn, TensorFlow, Keras, PyTorch, and pandas. These tools make it easier for developers and data scientists to preprocess data, build models, and evaluate their performance.

This introduction provides a foundational overview of how machine learning works, the types of problems it can solve, and how to implement basic ML models using Python. Through practical examples, you'll learn how to load and explore data, choose suitable algorithms, train models, and make predictions. Whether you're a beginner or brushing up your skills, this guide aims to set the stage for deeper learning and real-world application.
Machine Learning with Python is a powerful combination that enables computers to learn from data and make predictions or decisions. Python's extensive libraries, including NumPy, pandas, scikit-learn, TensorFlow, and Keras, provide efficient implementations of various ML algorithms. With Python, you can explore supervised, unsupervised, and reinforcement learning, and apply ML to image classification, natural language processing, and predictive modeling.

To get started, install necessary libraries, explore datasets using pandas, and implement algorithms using scikit-learn or Tensor Flow. Additional resources include online courses on Coursera, edX, and Udemy, books like "Python Machine Learning" and "Hands-On Machine Learning", and tutorials on library documentation.

2. MACHINE LEARNING ARCHITECTURE

Machine Learning Architecture refers to the structured design and flow of processes and components used to build, train, and deploy machine learning models. It defines how data flows through the system and how each component interacts to transform raw data into valuable predictions or decisions.

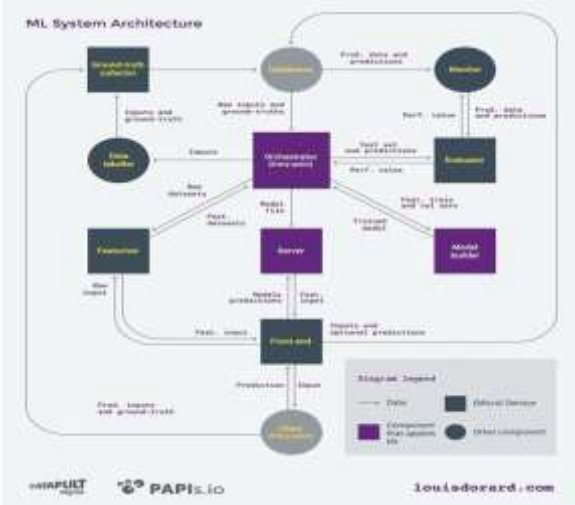


Fig. 2.1 Machine Learning with Python Architecture

Key Components of Machine Learning Architecture:

Data Collection Layer :

Gathers raw data from different sources (databases, APIs, sensors, etc.).

This is the foundation of any ML system.

Data Preparation Layer :

Involves cleaning, transforming, and organizing the data.

Tools: Pandas, NumPy, OpenCV, etc.

Feature Engineering :

Selecting and creating the most relevant input variables (features) for the model.

Helps improve model performance and accuracy.

Model Selection and Training :

Choosing an appropriate algorithm (e.g., Decision Trees, SVM, Neural Networks).

Training the model using training data.

Libraries: scikit-learn, TensorFlow, Keras.

Advantages of Machine Learning with Python:

Easy to Learn and Use:

Python has a simple syntax that closely resembles English, making it beginner-friendly and easy to write, read, and debug. This simplicity allows developers to focus more on solving ML problems rather than getting stuck in the language complexity.

Extensive Library Support:

Python boasts a rich ecosystem of libraries and frameworks tailored for machine learning and data science, including:

- NumPy and pandas for data manipulation
- Matplotlib and Seaborn for data visualization
- scikit-learn for classical ML algorithms
- TensorFlow, Keras, and PyTorch for deep learning

1. Community Support:

Python has one of the largest and most active communities. You can find extensive documentation, tutorials, and community forums to help troubleshoot issues and learn faster.

2. Platform Independence:

Python is cross-platform and can run on different operating systems like Windows, Linux, or macOS without modification, making development and deployment easier.

3. Rapid Prototyping:

Python's simplicity and available libraries enable quick testing and iteration of ML models, allowing developers to move from idea to prototype faster.

4. Integration and Compatibility:

Python can easily integrate with other languages like C/C++, Java, and R. It also supports web development frameworks and cloud platforms, making it suitable for end-to-end ML solutions.

5. Visualization Tools:

Python offers powerful libraries for data visualization, such as Matplotlib, Plotly, and Seaborn, which help in understanding data and interpreting model results effectively.

6. Scalability and Flexibility:

Python can be used for small-scale testing as well as production-level ML applications. It supports both object-oriented and functional programming styles, giving developers flexibility in their approach.

Machine Learning with Python Frameworks:

Machine learning with Python is widely supported thanks to a rich ecosystem of development frameworks and libraries. Here's a breakdown of some popular Python machine learning frameworks and their typical use cases.

Some popular Python machine learning frameworks and libraries include:

1. **Scikit-learn:**
   - ❖ Classification, regression, clustering, dimensionality reduction.
   - ❖ Scikit-learn provides simple and efficient tools for data mining and data analysis.
2. **TensorFlow:**
   - ❖ Deep learning, neural networks, large-scale distributed training.
   - ❖ TensorFlow is an open-source framework for building and training deep learning models.
3. **Keras:**
   - ❖ Deep learning, neural networks, rapid prototyping.
   - ❖ Keras is a high-level neural networks API that can run on top of TensorFlow or Theano.
4. **PyTorch:**
   - ❖ Deep learning, neural networks, rapid prototyping.
   - ❖ PyTorch is an open-source framework that provides a dynamic computation graph.
5. **Light GBM:**
   - ❖ Gradient boosting, classification, regression.
   - ❖ LightGBM is a fast and efficient gradient boosting framework.
6. **Cat Boost**
   - ❖ Gradient boosting, classification, regression.

- ❖ Cat Boost is a gradient boosting framework that provides efficient and accurate models.

7. **XG Boost**
   - ❖ Gradient boosting, classification, regression.
   - ❖ XG Boost is a popular gradient boosting framework that provides efficient and accurate models.

These frameworks and libraries make it easy to build, train, and deploy machine learning models in Python.

Tools and Technologies Of Machine Learning with Python:

Here are some popular tools and technologies for Machine Learning with Python development:

- ➢ Machine Learning Libraries:
  - Scikit-learn: A widely used library for machine learning algorithms.
  - TensorFlow: An open-source framework for deep learning.
  - Keras: A high-level neural networks API.
  - PyTorch: An open-source framework for deep learning.
- ➢ Data Science Libraries:
  - NumPy: A library for numerical computations.
  - Pandas: A library for data manipulation and analysis.
  - Matplotlib: A library for data visualization.
- ➢ Deep Learning Frameworks:
  - TensorFlow: An open-source framework for deep learning.
  - PyTorch: An open-source framework for deep learning.
  - Keras: A high-level neural networks API.
- ➢ Model Deployment:
  - TensorFlow Serving: A framework for deploying machine learning models.
  - AWS SageMaker: A cloud-based platform for machine learning.
  - Azure Machine Learning: A cloud-based platform for machine learning.
- ➢ Data Storage:
  - Pandas DataFrames: A data structure for storing and manipulating data.

- NumPy arrays: A data structure for numerical computations.
- Databases: Relational databases (e.g., MySQL) and NoSQL databases (e.g., MongoDB).

➢ Development Tools:

- Jupyter Notebook: An interactive environment for data science.
- PyCharm: An integrated development environment (IDE) for Python.
- Visual Studio Code: A lightweight, open-source code editor.

These tools and technologies enable developers to build, train, and deploy machine learning models efficiently.

3. COURSE OVERVIEW

Here's a course overview of Machine Learning with Python:

- **Course Description**
  - This course covers the fundamentals of machine learning with Python, including supervised and unsupervised learning, regression, classification, clustering, and neural networks.

- **Course Topics**
  - Introduction to Machine Learning: Overview of machine learning, types of machine learning, and applications.
  - Python Basics: Introduction to Python programming language.
  - Data Preprocessing: Handling missing data, data normalization, and feature scaling.
  - Supervised Learning: Regression, classification, and logistic regression.
  - Unsupervised Learning: Clustering, dimensionality reduction, and principal component analysis.
  - Neural Networks: Introduction to neural networks, deep learning, and TensorFlow/Keras.
  - Model Evaluation: Metrics for evaluating model performance.

- **Course Objectives**
  - Understand machine learning concepts: Learn the basics of machine learning and its applications.
  - Implement machine learning algorithms: Use Python libraries like scikit-learn and TensorFlow/Keras to implement machine learning algorithms.
  - Work with real-world data: Apply machine learning concepts to real-world datasets.

- **Target Audience**
  - Data scientists: Professionals working with data who want to learn machine learning.
  - Python developers: Developers who want to learn machine learning with Python.

- Students: Students interested in machine learning and data science.

➢ **Prerequisites**

- Python programming: Basic knowledge of Python programming language.
- Data analysis: Familiarity with data analysis concepts.

This course provides a comprehensive overview of machine learning with Python, covering both theoretical and practical aspects.

4. TECHNOLOGY STACK

4.1. Python

Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy which emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly braces or keywords), and a syntax which allows programmers to express concepts in fewer lines of code than possible in languages such as C++ or Java. The language provides constructs intended to enable writing clear programs on both a small and large scale. Python features a dynamic type system and automatic memory management and supports multiple programming paradigms, including object oriented, imperative, functional programming, and procedural styles. It has a large and comprehensive standard library.

4.2 NumPy

NumPy is a Python library used for working with arrays.It also has functions for working in domain of linear algebra, fourier transform, and matrices.It is an open source project and you can use it freely.NumPy stands for Numerical Python.In Python we have lists that serve the purpose of arrays, but they are slow to process.NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.Arrays are very frequently used in data science, where speed and resources are very important.Data Science is a branch of computer science where we study how to store, use and analyze data for deriving information from it.NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently.This behavior is called locality of reference in computer science.This is the main reason why NumPy is faster than lists. Also it is optimized to work with latest CPU architectures.NumPy is a Python library and is written partially in Python, but most of the parts that require fast computation are written in C or C++.

4.3. Pandas

Pandas is a Python library used for working with data sets.It has functions for analyzing, cleaning, exploring, and manipulating data. The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008.Pandas allows us to analyze big data and make conclusions based on statistical theories. Pandas can clean messy data sets, and make them readable and relevant. Relevant data is very important in data science. Data Science is a branch of computer science where we study how to store, use and analyze data for deriving information from it. Pandas gives you answers about the data. Like - Is there a correlation between two or more columns, average value, Max value, Min value. Pandas are also able to delete rows that are not relevant, or contains wrong values, like empty or NULL values. This is called cleaning the data.

4.4. Machine Learning

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves.

Features of Machine Learning:

• Machine Learning is computing-intensive and generally requires a large amount of training data.

• It involves repetitive training to improve the learning and decision making of algorithms.

• As more data gets added, Machine Learning training can be automated for learning new data patterns and adapting its algorithm.

Applications of Machine Learning :

➢ Image Processing

- ▪ Optical Character Recognition (OCR)
- ▪ Self-driving cars ▪ Image tagging and recognition

➢ Robotics

- ▪ Industrial robotics

- Human simulation

➢ Data Mining

- Association rules
- Anomaly detection
- Grouping and Predictions

➢ Video games

- Pokémon
- PUBG

➢ Text Analysis

- Spam Filtering
- Information Extraction
- Sentiment Analysis

➢ Healthcare

- Emergency Room & Surgery
- Research
- Medical Imaging & Diagnostics

5. INTERNSHIP PART

Here's a potential internship project idea in Machine Learning with Python:

- **Project Idea:**
  - Predicting Customer Churn for a Telecom Company
  - Objective: Develop a machine learning model using Python to predict customer churn for a telecom company based on historical customer data.
- **Tasks:**
  - Data Preprocessing: Clean and preprocess the dataset, handling missing values and outliers.
  - Feature Engineering: Extract relevant features from the dataset that can help predict customer churn.
  - Model Development: Train and evaluate different machine learning models (e.g., logistic regression, decision trees, random forests) using Python libraries like scikit-learn.
  - Model Evaluation: Evaluate the performance of the models using metrics like accuracy, precision, recall, and F1-score.
- **Required Skills:**
  - Python programming: Proficiency in Python programming language.
  - Machine learning: Knowledge of machine learning concepts and algorithms.
  - Data preprocessing: Experience with data preprocessing techniques.
  - Model evaluation: Understanding of model evaluation metrics.
- **Tools and Technologies:**
  - Python libraries: scikit-learn, pandas, NumPy, Matplotlib.
  - Data storage: CSV files or databases.
  - Model deployment: Flask or Django.
- **Deliverables:**
  - Code repository: A Git repository containing the code for the project.
  - Model performance report: A report detailing the performance of the models.
  - Deployment documentation: Documentation on how to deploy the model.

ACTIVITY LOG FOR THE FIRST WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Introduction to ML, its need, and real-life applications.	Understand what Machine Learning is and its importance.	
Day-2	Supervised, Unsupervised, Reinforcement learning.	Classify different types of ML.	
Day-3	Syntax, variables, data types, input/output.	Introduction to Python basics.	
Day-4	If-else, loops, functions.	Control flow in Python.	
Day-5	Introduction to NumPy and Pandas.	Use Python libraries for data handling.	
Day-6	Build a simple linear regression model.	Simple ML implementation using scikit-learn.	

WEEKLY REPORT

WEEK – 1 (From Dt : 09/12/2024 To Dt: 14/12/2024)

1. Understood the basics of Machine Learning and its importance.
2. Differentiated between supervised, unsupervised, and reinforcement learning.
3. Learned Python syntax, variables, control structures.
4. Explored Python libraries like NumPy and Pandas.
5. Built a simple regression model using scikit-learn.

ACTIVITY LOG FOR THE SECOND WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Exploring pandas data structures.	Understand data frames and series.	
Day-2	Indexing, slicing, filtering.	Perform data operations with Pandas.	
Day-3	Arrays, broadcasting, vectorization.	Handle numerical operations with NumPy.	
Day-4	Handling nulls, outliers, duplicates.	Data cleaning techniques.	
Day-5	Normalization, scaling, encoding.	Data preprocessing basics.	
Day-6	Load and explore datasets from UCI or Kaggle.	Real-life dataset hands-on.	

WEEKLY REPORT

WEEK – 2 (From Dt : 16/12/2024 To Dt: 21/12/2024)

1. Learned about Data Frames, Series, and array structures.
2. Performed data cleaning: handling nulls, outliers, and duplicates.
3. Applied normalization and feature scaling techniques.
4. Practiced feature extraction and data transformation.
5. Handled real-world datasets from open sources.

ACTIVITY LOG FOR THE THIRD WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Line, bar, histogram, scatter.	Learn basic plots using Matplotlib.	
Day-2	Box plots, violin plots, heatmaps.	Enhance visual appeal using Seaborn.	
Day-3	Use .corr() and Seaborn pair plots.	Plot correlation and distribution.	
Day-4	Visualize regression/classification results.	Plot ML results for insights.	
Day-5	End-to-end EDA with a dataset.	Exploratory Data Analysis (EDA).	
Day-6	Use Plotly or Dash basics.	Build a visualization dashboard.	

WEEKLY REPORT

WEEK – 3 (From Dt : 23/12/2024 To Dt:28/12/2024)

1. Visualized data using Matplotlib and Seaborn.
2. Plotted distributions, correlations, and trends.
3. Interpreted visual results to understand data patterns.
4. Conducted Exploratory Data Analysis (EDA).
5. Created dashboards with Plotly/Dash.

ACTIVITY LOG FOR THE FOURTH WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Linear vs nonlinear regression.	Understand regression fundamentals	
Day-2	Use scikit-learn on housing data.	*:* Implement Linear Regression.	
Day-3	MAE, MSE, RMSE, R^2.	Evaluate regression performance.	
Day-4	Feature transformation.	Polynomial Regression.	
Day-5	Avoiding overfitting.	Regularization (Ridge & Lasso).	
Day-6	Predict car prices using regression.	Regression project.	

WEEKLY REPORT

WEEK – 4 (From Dt : 30/12/2024 To Dt: 04/01/2025)

1. Understood linear, polynomial, and regularized regression.
2. Trained models with scikit-learn on real datasets.
3. Evaluated models using RMSE, MAE, and R^2 metrics.
4. Applied Ridge and Lasso to prevent overfitting.
5. Completed a car price prediction project

ACTIVITY LOG FOR THE FIFTH WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Logistic Regression intro.	Understand classification models.	
Day-2	Logistic regression with Scikit-learn.	Implement classification.	
Day-3	Accuracy, Precision, Recall, F1.	Confusion matrix & metrics.	
Day-4	KNN classifier from scratch.	K-Nearest Neighbors (KNN).	
Day-5	Tree-based models.	Decision Trees and Random Forests.	
Day-6	Titanic survival prediction.	Classification project.	

WEEKLY REPORT

WEEK – 5 (From Dt : 06/01/2025 To Dt: 11/01/2025)

1. Studied logistic regression, KNN, and decision trees.
2. Applied classification algorithms to labeled datasets.
3. Measured model accuracy using confusion matrix and F1-score.
4. Built and optimized tree-based classifiers.
5. Completed a Titanic survival prediction project.

ACTIVITY LOG FOR THE SIXTH WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	K-Means clustering theory.	Introduction to clustering.	
Day-2	Customer segmentation case study.	Implement K-Means.	
Day-3	Optimal cluster determination.	Elbow method and Silhouette Score	
Day-4	Dendrograms and linkage.	Hierarchical clustering.	
Day-5	Density-based clustering.	DBSCAN.	
Day-6	Market basket analysis.	Clustering mini-project.	

WEEKLY REPORT

WEEK – 6 (From Dt : 13/01/2025 To Dt: 18/01/2025)

1. Explored clustering techniques like K-Means and DBSCAN.
2. Determined optimal clusters with Elbow and Silhouette.
3. Performed customer segmentation and market analysis.
4. Visualized cluster groupings and patterns.
5. Completed a real-world clustering mini-project.

ACTIVITY LOG FOR THE SEVENTH WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Need for reducing features.	Understand curse of dimensionality.	
Day-2	Eigenvalues, variance explanation.	PCA theory	
Day-3	PCA with scikit-learn.	Implement PCA.	
Day-4	Visualize high-dimensional data	t-SNE & UMAP overview.	
Day-5	Reduce features before modeling.	Apply PCA in ML pipelines.	
Day-6	Use PCA on customer data.	Dimensionality reduction project.	

WEEKLY REPORT

WEEK – 7 (From Dt : 20/01/2025 To Dt: 25/01/2025)

1. Learned PCA for feature reduction.
2. Visualized high-dimensional data using t-SNE and UMAP.
3. Applied dimensionality reduction before modeling.
4. Analyzed variance explained by principal components.
5. Enhanced model performance by reducing noise.

ACTIVITY LOG FOR THE EIGHTH WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Train/test split, cross-validation.	Understand model evaluation strategies.	
Day-2	Regression and classification metrics.	Performance metrics recap.	
Day-3	Grid Search CV and Randomized Search CV.	Hyperparameter tuning.	
Day-4	Compare ML models on same dataset.	Model selection project.	
Day-5	Understand underfitting vs overfitting.	Bias-Variance tradeoff.	
Day-6	Bagging, boosting (Intro to XGBoost).	Ensemble methods overview.	

WEEKLY REPORT

WEEK – 8 (From Dt : 27/01/2025 To Dt: 01/02/2025)

1. Applied cross-validation and test/train splits.
2. Tuned hyperparameters with GridSearchCV.
3. Compared multiple model performances.
4. Understood the bias-variance tradeoff.
5. Explored ensemble methods like Bagging and Boosting.

ACTIVITY LOG FOR THE NINTH WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Learn about neurons, layers, and activations	Understand the structure of neural networks.	
Day-2	Use the Sequential API	Build your first neural network using Keras.	
Day-3	Use datasets like MNIST.	Train and test neural networks.	
Day-4	Apply techniques like dropout and batch normalization	Improve model performance.	
Day-5	Use Tensor Board for visualization.	Implement callbacks and monitoring.	
Day-6	Build a digit recognition model.	Neural network project	

WEEKLY REPORT

WEEK – 9 (From Dt : 03/02/2025 To Dt: 08/02/2025)

1. Built simple neural networks using Keras.
2. Trained and validated models using MNIST
3. Improved accuracy with batch norm and dropout.
4. Visualized model training with TensorBoard.
5. Completed a digit recognition mini-project.

ACTIVITY LOG FOR THE TENTH WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Learn about filters, strides, pooling	Understand CNN architecture	
Day-2	Apply to image classification.	Build a CNN using Keras.	
Day-3	Improve with data augmentation	Train CNNs on datasets like CIFAR-10	
Day-4	Apply VGG16, ResNet.	Use pretrained models (Transfer Learning)	
Day-5	Freeze and unfreeze layers	Fine-tune models	
Day-6	Image classification with transfer learning	CNN project	

WEEKLY REPORT

WEEK – 10 (From Dt : 10/02/2025 To Dt: 15/02/2025)

1. Learned convolutional layers and pooling mechanisms.
2. Built CNNs for image classification.
3. Applied transfer learning with VGG16/ResNet.
4. Fine-tuned models and improved generalization.
5. Completed an image classification project.

ACTIVITY LOG FOR THE ELEVEN WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Tokenization, stemming, lemmatization.	Intro to NLP and text data.	
Day-2	Use Count Vectorizer, TF-IDF	Text vectorization	
Day-3	Sentiment analysis with logistic regression.	Train ML models on text	
Day-4	Basic sequence modeling	Intro to RNNs and LSTMs.	
Day-5	Pretrained embeddings in Keras	Use embeddings like Word2Vec and GloVe.	
Day-6	Build a spam classifier or sentiment analyzer	NLP project.	

WEEKLY REPORT

WEEK – 11 (From Dt : 17/02/2025 To Dt: 22/02/2025)

1. Preprocessed text with tokenization and stemming.
2. Vectorized text using TF-IDF and Count Vectorizer.
3. Built classifiers for sentiment analysis and spam detection.
4. Introduced RNNs and LSTMs for sequence modeling.
5. Completed a text-based ML project.

ACTIVITY LOG FOR THE TWELVE WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Importance of ML model serving	Intro to model deployment	
Day-2	Create REST endpoints	Build a simple Flask API	
Day-3	Input/output handling	Integrate ML model with Flask	
Day-4	Build a simple HTML form to interact with API	Frontend basics	
Day-5	Use Postman or browser to test endpoints	Test and run locally.	
Day-6	Deploy a trained model with a Flask app	Mini deployment project.	

WEEKLY REPORT

WEEK – 12 (From Dt : 24/02/2025 To Dt: 01/03/2025)

1. Learned to build and serve models using Flask.
2. Integrated ML models with frontend HTML forms.
3. Tested APIs using Postman and browser tools.
4. Explored model deployment workflows.
5. Completed a deployed ML web application.

ACTIVITY LOG FOR THE THIRTEEN WEEK

DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	AWS Sage Maker, Azure ML, Google AI Platform.	Overview of cloud ML platforms.	
Day-2	Intro to AutoML tools	Deploy using Google Colab and Hugging Face	
Day-3	Upload model, inference, and test	Hands-on with AWS SageMaker.	
Day-4	Build and deploy a pipeline	Azure ML Studio overview	
Day-5	Dockerfile basics	Intro to Docker and model containerization	
Day-6	Choose one platform to deploy an end-to-end model	Cloud deployment project	

WEEKLY REPORT

WEEK – 13 (From Dt : 03/03/2025 To Dt: 08/03/2025)

1. Explored AWS SageMaker, Azure ML, and GCP.
2. Used AutoML tools for faster model deployment.
3. Practiced containerizing ML models with Docker.
4. Deployed end-to-end ML models on the cloud.
5. Completed a cloud deployment case study.

ACTIVITY LOG FOR THE FOURTEEN WEEK

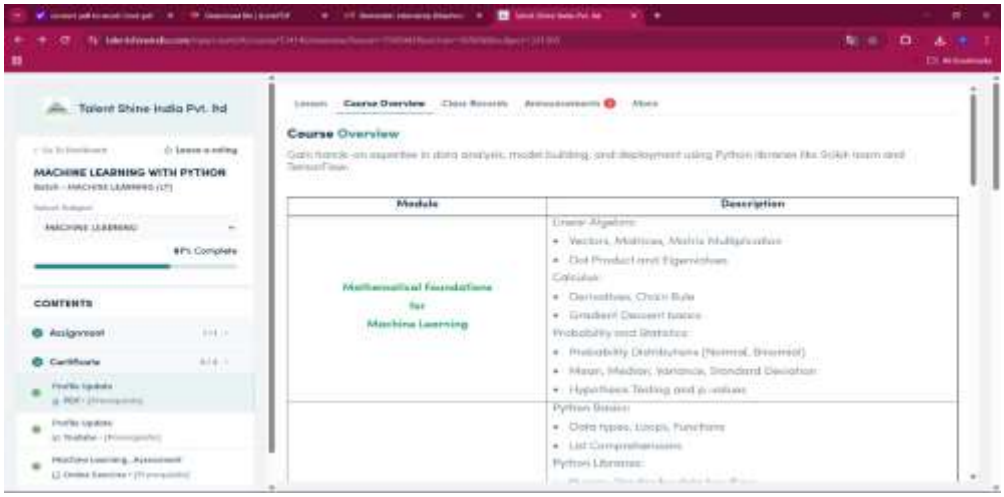
DAY & DATE	BRIEF DESCRIPTION OF THE DAILY ACTIVITY	LEARNING OUTCOME	Person In-charge Signature
Day-1	Choose domain and problem statement	Project planning.	
Day-2	Prepare dataset for training	Data collection and EDA.	
Day-3	Train models and evaluate	Model selection and training	
Day-4	Optimize performance	Model tuning and evaluation	
Day-5	Build frontend/backend	Deployment setup	
Day-6	Document and present the project.	Presentation and submission.	

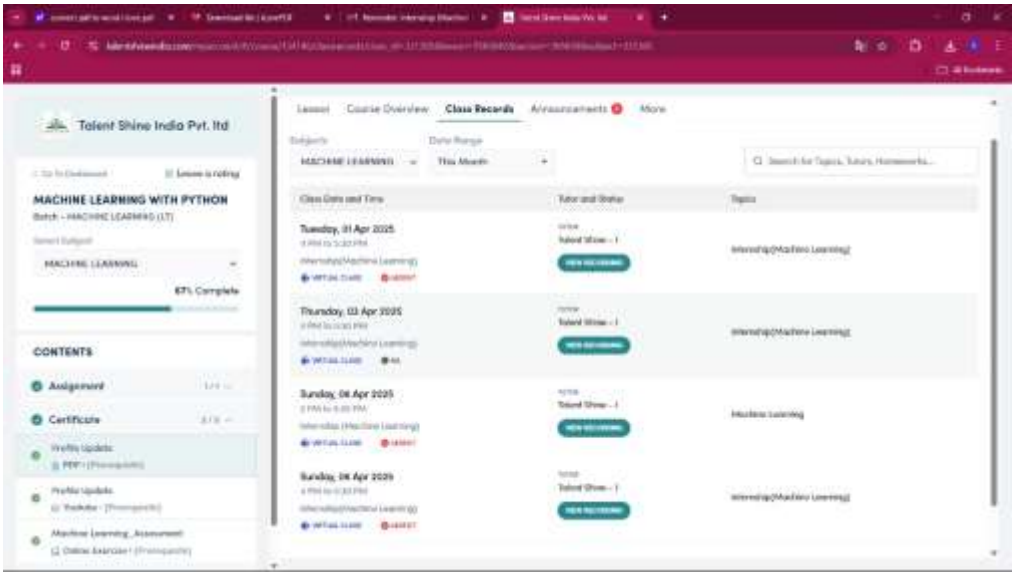
WEEKLY REPORT

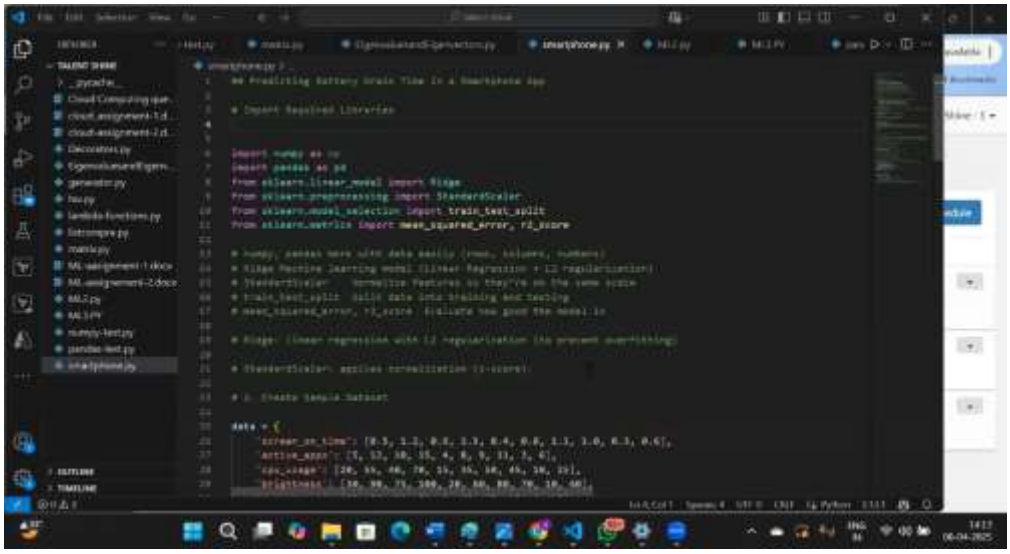
WEEK – 14 (From Dt : 10/03/2025 To Dt: 15/03/2025)

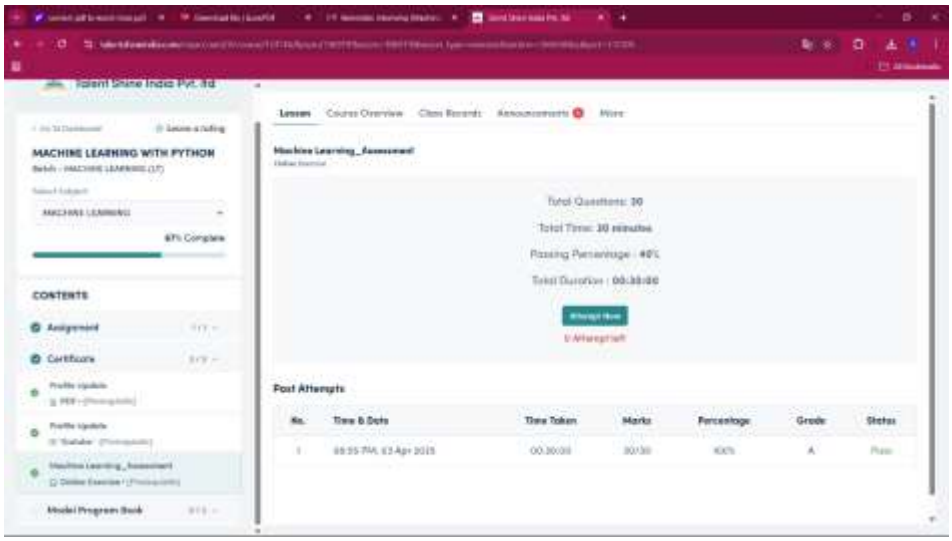
1. Chose a domain-specific ML problem.
2. Collected, cleaned, and explored the dataset.
3. Built and evaluated a full ML pipeline.
4. Deployed the model with Flask/cloud.
5. Presented and documented the final capstone project.

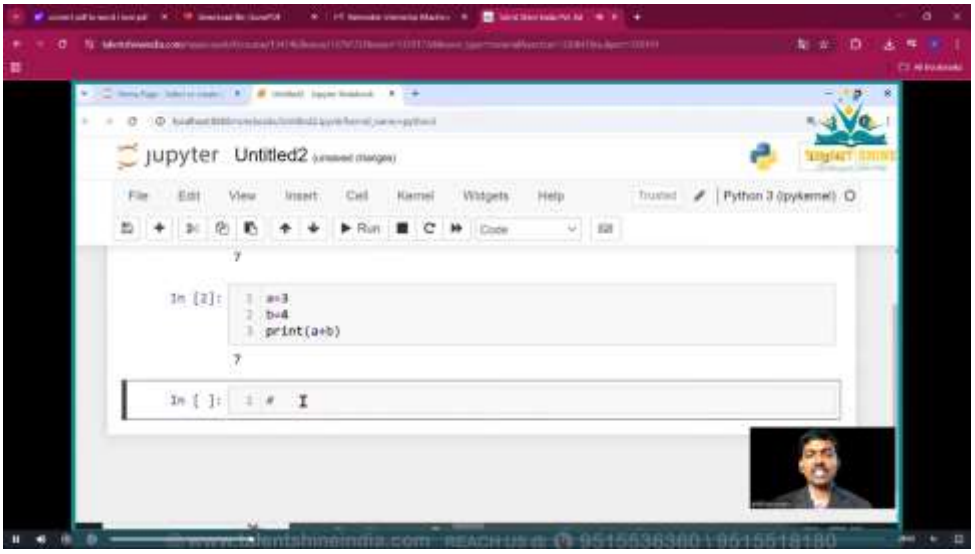
7. SCREEN SHOTS

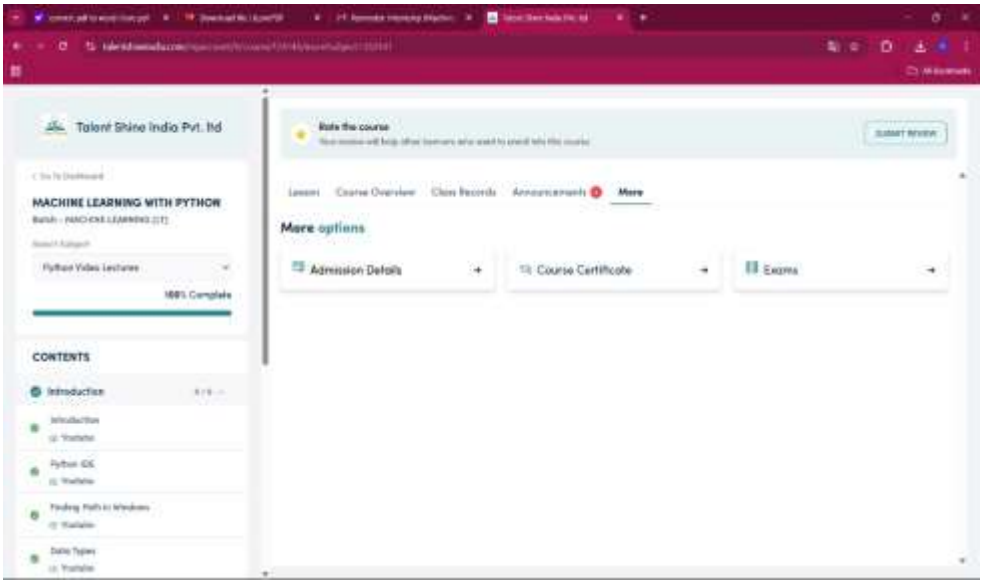







Talent Shine India Pvt. ltd
MACHINE LEARNING WITH PYTHON
CONTENTS
Lessons
Course Overview
Class Records
Announcements
More
Machine Learning_Assessment
Total Questions: 20
Total Time: 30 minutes
Passing Percentage : 40%
Past Attempts
No.
Time & Date
Time Taken
Marks
Percentage
Grade
Status


jupyter Untitled2
File Edit View Insert Cell Kernel Widgets Help
Python 3 (ipykernel)
In [2]:
a=3
b=4
print(a+b)
7
In [ ]:


Talent Shine India Pvt. ltd
MACHINE LEARNING WITH PYTHON
CONTENTS
Rate the course
Lessons
Course Overview
Class Records
Announcements
More
More options
Admission Details
Course Certificate
Exams

8. CONCLUSION

Machine Learning with Python is a powerful combination that enables developers to build intelligent systems, analyze complex data, and drive business insights. With its extensive libraries, including scikit-learn, TensorFlow, and Keras, Python provides a versatile platform for machine learning development. By leveraging machine learning algorithms and techniques, developers can build predictive models, classify objects, and identify patterns in data. As the field of machine learning continues to evolve, Python remains a popular choice for developers and data scientists alike. Machine Learning with Python has numerous applications across industries, and its potential continues to grow as the field evolves. In conclusion, Machine Learning with Python offers a robust platform for building intelligent systems, analyzing complex data, and driving business insights, empowering developers to unlock new possibilities and achieve remarkable results. In the field of machine learning, several advanced areas are being actively explored to enhance performance and expand capabilities. Deep learning, a subset of machine learning, focuses on using neural networks with many layers to handle complex tasks such as image recognition, speech processing, and natural language understanding.

Another growing area is real-time processing, where systems are designed to make immediate decisions or predictions as data is received, which is crucial for applications like fraud detection or autonomous driving. Additionally, there is an increasing emphasis on explainability, where researchers and developers work on techniques that make machine learning models more interpretable and transparent, allowing users to understand how decisions are made. These developments support a wide range of applications, including image classification, where systems can automatically categorize images; natural language processing, which involves analyzing and generating human language for tasks like translation or sentiment analysis; and predictive modeling, which uses historical data to forecast future events or trends in areas like finance, healthcare, and marketing.

9. REFERENCES

All Content used in this report is from

https://www.simplilearn.com/

https://www.wikipedia.org/

https://towardsdatascience.com/

https://www.expertsystem.com/

https://www.coursera.org/

https://www.edureka.co/

https://subhadipml.tech/

https://www.forbes.com/

https://medium.com/

https://www.google.com

Book I referred are

• Hands-on Machine Learning with Scikit-learn & Tensor flow By Aurelien Geron

• Python Machine Learning by Sebastian Raschk

Student *Self Evaluation of the Long-Term Internship*

Student Name & Registration
Term of Internship: From To
Date of Evaluation:
Organization & Address:
Name & Address of the Supervisor with Mobile Number:

Please rate our performance in the following areas:

Rating Scale: Letter grade of CGPA calculation to be provided

1) Oral communication	**1**	**2**	**3**	**4**	**5**
2) Written communication	**1**	**2**	**3**	**4**	**5**
3) Initiative	**1**	**2**	**3**	**4**	**5**
4) Interaction with staff	**1**	**2**	**3**	**4**	**5**
5) Attitude	**1**	**2**	**3**	**4**	**5**
6) Dependability	**1**	**2**	**3**	**4**	**5**
7) Ability to learn	**1**	**2**	**3**	**4**	**5**
8) Planning and organization	**1**	**2**	**3**	**4**	**5**
9) Professionalism	**1**	**2**	**3**	**4**	**5**
10) Creativity	**1**	**2**	**3**	**4**	**5**
11) Quality of work	**1**	**2**	**3**	**4**	**5**
12) Productivity	**1**	**2**	**3**	**4**	**5**
13) Progress of learning	**1**	**2**	**3**	**4**	**5**
14) Adaptability to organization's culture/policies	**1**	**2**	**3**	**4**	**5**
15) OVERALL PERFORMANCE	**1**	**2**	**3**	**4**	**5**

Signature of the Student

Evaluation by the Supervisor of the Intern Organization

Student Name & Registration

Term of internship From To

Date of Evaluation:

Organization Name & Address:

Name & Address of the
Supervisor with Mobile Number:

Please rate the student's performance in the following areas:

Please note that your evaluations hall be done independent of the student's

self-evaluation Rating Scale:

1 is lowest and 5 is highest rank

1) Oral communication	**1**	**2**	**3**	**4**	**5**
2) Written communication	**1**	**2**	**3**	**4**	**5**
3) Initiative	**1**	**2**	**3**	**4**	**5**
4) Interaction with staff	**1**	**2**	**3**	**4**	**5**
5) Attitude	**1**	**2**	**3**	**4**	**5**
6) Dependability	**1**	**2**	**3**	**4**	**5**
7) Ability to learn	**1**	**2**	**3**	**4**	**5**
8) Planning and organization	**1**	**2**	**3**	**4**	**5**
9) Professionalism	**1**	**2**	**3**	**4**	**5**
10)Creativity	**1**	**2**	**3**	**4**	**5**
11)Quality of work	**1**	**2**	**3**	**4**	**5**
12)Productivity	**1**	**2**	**3**	**4**	**5**
13)Progress of learning	**1**	**2**	**3**	**4**	**5**
14)Adaptability to organization's culture/policies	**1**	**2**	**3**	**4**	**5**
15)OVERALL PERFORMANCE	**1**	**2**	**3**	**4**	**5**

Signature of the Supervisor