Personal Details:   
Name: Suraj J  
Date of birth: 30/01/2004

Place: Bengaluru, Karnataka, India  
Phone Number: +91 7204215514  
Email ID: [sjsurajjayakumar@gmail.com](mailto:sjsurajjayakumar@gmail.com)  
Github: [github.com/SurajJayakumar2004](https://github.com/SurajJayakumar2004" \t "_blank)   
LinkedIn: [linkedin.com/in/jayakumarsuraj](http://www.linkedin.com/in/jayakumarsuraj)  
  
Preferred Job Role:   
Data Engineer, Data Analyst, Data Scientist, ML Engineer, MLOps DevOps   
  
Education:  
1. B.Tech in computer science and engineering Specializing in Artificial Intelligence and Machine Learning  
University/college: Jain Deemed-to-be University - 2022 to Present (expected graduation 2026)  
Bengaluru, India   
CGPA: 7.87/10

2. Pre-University (2nd PUC) / Class 12 – Karnataka State Board   
KMWA Pre University College - 2020 to 2022  
Bengaluru, India  
Percentage: 72%

3. Class 10 - CBSE Board  
Sri Vidya Kendra The Smart School - 2020

Bengaluru, India  
Percentage: 76%  
  
Internship:   
This is to certify that has successfully completed an internship with Miles Education in B.Tech Computer Science & Engineering (Artificial Intelligence & Machine Learning) from April 23rd, 2024 to May 3lst, 2024. During the internship, Suraj J demonstrated commitment, enthusiasm, and a strong ability to learn. This certificate acknowledges dedication, hard work, and valuable contributions during the tenure.

1. Objective & Goal:

* Objective: To leverage raw data to uncover and present insightful information through comprehensive data analysis and visualization.
* Goal: To successfully process, analyze, and visualize data, demonstrating a strong ability to transform complex datasets into actionable insights, thereby enhancing understanding of the underlying data.

2. Logic & Methodologies Used:

* Data Provision: The internship commenced with being provided raw data, which served as the foundation for the project.
* Data Engineering: This phase involved preparing the raw data for analysis. It likely included tasks such as data cleaning, transformation, structuring, and potentially loading it into a format suitable for analysis and visualization.
* Data Analysis: Extensive analysis was performed on the prepared data to identify patterns, trends, anomalies, and correlations. This process was aimed at extracting meaningful and "very insightful information" from the dataset.
* Data Visualization: A core component of the internship involved translating the analytical findings into visual representations. This was done to effectively communicate the insights derived from the data, making complex information accessible and understandable.

3. Statistical Data & Models Used:

* Data Analysis: The project heavily relied on statistical methods implicitly used during data analysis to derive insights. This would involve summarizing data, identifying distributions, correlations, and potentially basic hypothesis testing, though no specific complex statistical models or machine learning algorithms are mentioned as being explicitly developed or used for predictive purposes in this particular internship description. The focus was on identifying existing patterns and insights from raw data.

4. Skills Used:

* Data Engineering: Skills in transforming and preparing raw data.
* Data Analysis: Proficiency in interpreting complex datasets and identifying key insights.
* Data Visualization: Expertise in creating clear, informative, and impactful visual representations of data.
* Programming Languages: Python (specifically for data visualization libraries).
* Business Intelligence Tools: PowerBI.
* Problem-Solving: Ability to approach raw, unrefined data and extract valuable information.
* Commitment & Enthusiasm: Demonstrated a strong work ethic and positive attitude throughout the tenure.
* Learning Ability: Showcased a strong capacity to acquire and apply new knowledge and skills within the internship period.

Experience:   
I possess project-based experience, and the following are the primary projects that have significantly contributed to my hands-on experience:

Project 1:  
Title - Author Identification (NLP, Machine Learning & Information Retrieval)  
Role – Developer(Team Project)   
Project details:  
Objective & Goal

* Primary Objective: To develop an automated author identification system that can accurately determine the author of a document among a predetermined set of candidates.
* Problem Addressed: Authorship attribution is complex due to subtle, variable writing styles, high-dimensional feature spaces, style and semantic ambiguity, data variability, and imbalanced feature representation.
* Key Objectives:
  + Develop a robust preprocessing pipeline to clean and normalize raw textual data, organizing it with clear author labels.
  + Create partitioned datasets (training and testing sets) using stratified sampling to preserve class distribution.
  + Build a robust, scalable, and efficient system using machine learning and deep learning techniques to consistently perform in real-world conditions.

Statistical Data

* Dataset: Reuters-50 dataset.
* Number of Authors: 50 known Reuters journalists.
* Evaluation Metrics: Accuracy, Precision, Recall, F1-score, and Confusion Matrices.
* Illustrative Model Accuracies:
  + Logistic Regression: 68.2%
  + Random Forest: 72.0%
  + XGBoost: 74.5%
  + RoBERTa Classifier: 77.8%
  + Ensemble Model (Averaging): 81.3%
* Sentence-BERT Embedding Dimension: 768-dimensional vector.

Logic Used

* Hybrid Approach: Combines classical stylometric techniques and modern deep learning models.
* Closed-Set Multi-Class Classification: Assumes the true author is within a predefined group of 50 candidates.
* Feature Extraction: Generates a diverse feature set including:
  + Stylometric Features: Average word length, average sentence length, punctuation ratio.
  + Character N-grams: Specifically trigrams, using CountVectorizer.
  + Part-of-Speech (POS) Features: Reflect syntactic tendencies.
  + Deep Semantic Embeddings: From transformer-based models.
* Ensemble Averaging Strategy: Aggregates class probabilities from multiple models to produce a final prediction, chosen for its simplicity, effectiveness, and ability to smooth individual model errors and balance diverse features.

Models Used

* Individual Classifiers:
  + Logistic Regression
  + Random Forest
  + XGBoost
* Deep Learning Model:
  + Shallow neural classifier built on RoBERTa embeddings.
* Ensemble Model: Soft Voting Ensemble based on probability averaging of the individual classifiers.

LLMs Used

* RoBERTa (specifically roberta-base)
* Sentence-BERT (SBERT)
* BERT Embeddings (general reference for transformer models)

Skills Used

* Natural Language Processing (NLP): Authorship Attribution, Stylometry, Text Preprocessing, Text Classification, POS Tagging, TF-IDF, Semantic Embeddings, Character n-grams.
* Machine Learning (ML): Supervised Learning, Ensemble Learning, Multiclass Classification, Feature Engineering, Model Training, Model Evaluation, Voting Classifiers, Gradient Boosting.
* Deep Learning: Transformer Models, Shallow Neural Networks, Transfer Learning.
* Programming Languages & Libraries:
  + Python
  + scikit-learn
  + spaCy
  + NLTK
  + Hugging Face
  + Transformers
  + Sentence-Transformers
  + xgboost
  + PyTorch
  + Pandas
  + NumPy
  + Joblib
  + Matplotlib & Seaborn (for visualization)
* Data Science & Analysis: Data Collection, Data Preprocessing, Feature Extraction, Model Evaluation, Error Analysis, Interpretability.

Project 2:  
Title - Stock Prediction & Recommendation System (Data Analysis, ML & Modeling):  
Role - Data Analyst(Independent Project)  
Project details:

This project, "Stock\_Prediction\_Options," is designed to provide stock and options recommendations for Nifty 50 companies through a combination of data analysis, technical indicators, sentiment analysis, and machine learning models.

Here are the details in pointers:

1. Objective & Goal:

* Objective: To analyze Nifty 50 stocks and provide recommendations, including both stock buy/hold signals and options pricing/recommendations.
* Goal: To download the latest stock data, analyze it using technical indicators and sentiment, train predictive models, and offer insights for potential trading decisions.

2. Logic Used:

* Data Acquisition and Storage:
  + Historical stock data for Nifty 50 symbols is downloaded from Yahoo Finance (yfinance).
  + This data is stored in a MySQL database named nifty50.
  + The system checks if the database and specific stock tables exist, creating them if necessary.
  + Data is updated incrementally, downloading only new data since the last stored date.
  + Table schema includes Date (PRIMARY KEY), Symbol, Open, High, Low, Close, and Volume.
* Technical Analysis Logic:
  + Calculates various technical indicators using the ta library, including:
    - MACD Difference (MACD\_diff).
    - Simple Moving Averages (SMA) for windows of 10 (SMA20), 15 (SMA50), and 200 (SMA200).
    - RSI (Relative Strength Index) with a window of 50.
    - Volume Average (Volume\_Avg) over a 20-day window.
  + Generates specific buy signals:
    - Decision MACD: MACD\_diff turning positive. 1212121212
    - Decision Golden Cross (Decision GC): SMA20 > SMA50, RSI < 100, MACD\_diff > 0, and Volume > Volume\_Avg.
    - Decision RSI/SMA: Close > SMA200 and RSI < 30.
  + A "Strong Buy" technical recommendation is given if 2 or more buy signals are present.  A "Weak Buy" is given if at least 1 buy signal is present. Otherwise, it's a "Hold".
* Sentiment Analysis Logic:
  + Fetches news headlines for each stock symbol from Google News.
  + Calculates a sentiment score for the headlines using TextBlob.
  + A "Strong Buy" sentiment recommendation is given if the sentiment\_score is greater than 0.1.
* Options Pricing Logic:
  + Uses the Black-Scholes model to calculate theoretical call and put option prices.
  + Inputs to the Black-Scholes model include: current stock price, strike price (at-the-money is used, equal to current price), time to expiration (4 weeks, converted to years), risk-free rate (0.07 or 7% annual yield), and historical volatility (annualized from 30-day percentage change).
  + Calculates Greeks (Delta, Gamma, Vega, Theta) for options.
* Recommendation Generation:
  + Combines sentiment and technical recommendations.
  + For call options, a "Sell Signal" is triggered if the current Black-Scholes Call Price reaches 10% above the recorded "Buy Price."

3. Statistical Data & Models Used:

* Statistical Data Points Used for Indicators and Models:
  + Close price, Open price, High price, Low price, Volume.
  + Calculated indicators: RSI, SMA10, SMA15, SMA200, MACD\_diff, Volume\_Avg.
* Machine Learning Models:
  + RandomForestClassifier: Used for stock prediction.
    - Hyperparameter Optimization: GridSearchCV and RandomizedSearchCV are used for tuning n\_estimators ([50, 100, 200]), max\_depth ([10, 20, 30]), and min\_samples\_split ([2, 5, 10]).
    - Evaluation Metric: accuracy\_score is used.
  + XGBClassifier: Also used for stock prediction, evaluated using accuracy\_score.
  + LSTM (Long Short-Term Memory) Model: A sequential Keras model with two LSTM layers (50 units) and dropout (0.2).
    - Input Shape: (X\_train.shape[1], X\_train.shape[2])
    - Training: 10 epochs, batch size 32.
    - Loss Function: mean\_squared\_error.
    - Data Preparation: MinMaxScaler is used, and a 60-time-step input sequence is created.
* Statistical Models (Option Pricing):
  + Black-Scholes Option Pricing Model: Used to calculate call and put option prices based on stock price, strike price, time to expiration, risk-free rate, and volatility.
  + Normal Distribution (scipy.stats.norm): Utilized within the Black-Scholes formula.

4. Skills Used:

* Programming Languages: Python.
* Data Science/Machine Learning:
  + Machine learning model development (Random Forest, XGBoost, LSTM).
  + Hyperparameter tuning (GridSearchCV, RandomizedSearchCV).
  + Feature engineering (creating technical indicators).
  + Model evaluation (accuracy score).
  + Time series data handling.
* Financial Knowledge:
  + Understanding of stock market concepts (buy, sell, hold).
  + Technical analysis indicators (MACD, SMA, RSI, Golden Cross).
  + Options trading concepts (call, put, strike price, expiration, implied volatility).
  + Option pricing models (Black-Scholes).
  + Greeks calculation (Delta, Gamma, Vega, Theta).
* Database Management:
  + MySQL database interaction (creating databases/tables, reading/writing data).
  + SQL queries.
* Web Scraping/API Interaction:
  + Fetching financial data from Yahoo Finance API.
  + Basic web scraping for news headlines (using requests and BeautifulSoup).
* Data Handling and Preprocessing:
  + Pandas DataFrames manipulation.
  + Handling missing data (dropna).
  + Data scaling (MinMaxScaler).
* Software Engineering:
  + Modular code design (separation into app, database, models, scripts).
  + Error handling (try-except blocks).
  + Command-line interface execution (if \_\_name\_\_ == "\_\_main\_\_":).

Project 3:  
Title – Personalized Recommendation Engines

Role - Data Analyst(Independent Project)  
Project details:

This project focused on understanding and implementing the core mechanics of personalized recommendation systems, with the ultimate goal of enhancing user experience in large-scale applications.

1. Objective & Goal:

* Objective: To delve into the fundamental principles and practical implementation of personalized recommendation systems.
* Goal: To develop robust recommendation algorithms capable of generating personalized recommendations, while also gaining crucial insights into the challenges associated with scalability and performance optimization for large datasets.

2. Logic & Methodologies Used:

* Data Preprocessing: Involved cleaning, transforming, and preparing raw data for analysis and model training.
* Feature Extraction: Identifying and extracting relevant features from user interaction data that would be indicative of preferences and patterns.
* Feature Engineering: Creating new features from existing ones to improve the performance and accuracy of recommendation algorithms.
* Extensive Data Analysis: Deep-diving into user interaction patterns to uncover intricate relationships and behaviors that inform personalization. This involved analyzing large volumes of data to understand trends, preferences, and implicit signals.
* Recommendation Algorithm Development: Designing, implementing, and experimenting with various recommendation algorithms. While specific algorithms were not detailed, this likely included approaches such as collaborative filtering (user-based or item-based), content-based filtering, or hybrid models.
* Scalability and Performance Optimization: A key aspect of the project involved addressing the challenges of processing and analyzing vast amounts of data efficiently, focusing on strategies to ensure the system remains performant even as data scales.

3. Statistical Data & Models Used:

* Data Analysis: The project heavily relied on quantitative data analysis to identify patterns in user interactions. This would involve statistical methods to summarize, visualize, and interpret large datasets to derive insights into user behavior.
* User Interaction Patterns: Identification of these patterns would involve statistical clustering, association rule mining, or other data mining techniques to group similar users or items.
* Recommendation Algorithms: While specific models aren't named, recommendation engines often employ a range of statistical and machine learning models, including:
  + Matrix Factorization techniques (e.g., Singular Value Decomposition - SVD, Alternating Least Squares - ALS).
  + Clustering algorithms (e.g., K-Means, DBSCAN) for user or item segmentation.
  + Classification/Regression models for predicting user preferences or ratings.
  + Tree-based models (e.g., Decision Trees, Random Forests, Gradient Boosting Machines) for learning complex relationships.
  + Potentially, nearest-neighbor algorithms for collaborative filtering.
  + Evaluation metrics like Precision, Recall, F1-score, RMSE, MAE, or coverage would have been used to assess the performance of the developed algorithms.

4. Skills Used:

* Programming Languages: Python.
* Data Analysis & Manipulation: Hands-on experience with Python libraries like Pandas and NumPy for complex data manipulation and large-scale data processing.
* Machine Learning/Algorithm Development: Deep understanding and implementation skills in various recommendation algorithms.
* Big Data Analysis: Expertise in processing and analyzing vast datasets.
* Data Preprocessing & Feature Engineering: Proficient in preparing data for modeling and creating impactful features.
* Performance Optimization: Skills in optimizing data processing and algorithm execution for scalability.
* Problem Solving: Identifying challenges in personalized recommendations and devising effective strategies.
* Analytical Thinking: Ability to identify intricate user interaction patterns and derive insights from complex data.

Project 4:  
Title – Real-Time Multi-Device Tracking System:

Role - Full Stack Developer - Independent Project

Project details:

This project focuses on developing a web application designed for administrators and managers to monitor the locations of multiple devices in near real-time.

1. Objective & Goal:

* Objective: To design and implement a comprehensive Real-Time Multi-Device Tracking System.
* Goal:
  + To provide a centralized and intuitive dashboard for authorized users to securely log in and gain comprehensive visibility into their tracked devices.
  + To enable viewing a list of all monitored devices and accessing detailed information, including their most recently reported locations.
  + To incorporate advanced features such as geofencing (defining virtual boundaries and generating alerts) , route dispatch functionality , and overall analytics for data-driven decision-making.
  + To include user and driver management features for administrative control.

2. Logic Used:

* Overall Architecture: The system employs a standard client-server architecture. The frontend is a web application built with React, and it communicates with a Node.js/Express.js backend. A MySQL database is used for data storage. This modular design enhances maintainability, scalability, and cross-platform compatibility.
* Frontend Design:
  + Developed using the React JavaScript library, following a component-based approach for dynamic and interactive user interfaces.
  + Utilizes Chakra UI for accessible and customizable styling and UI components.
  + React Router DOM manages application navigation.
  + A custom API utility (api.js) handles communication with the backend.
  + React Hooks (useState, useEffect, useContext) are used for efficient state management, ensuring a dynamic and responsive user experience.
* Backend Design:
  + Built with Node.js and the Express.js framework for scalable, high-performance server-side operations.
  + The mysql2/promise library enables asynchronous interaction with the MySQL database.
  + Security features include bcrypt for secure password hashing and JSON Web Tokens (JWTs) for stateless authentication and authorization.
  + Environment variables are managed using dotenv.
  + The codebase follows an MVC (Model-View-Controller) or similar architectural pattern, organizing logic into models (database interaction), controllers (business logic), routes (API endpoints), and middleware (request processing like verifyToken).
  + Provides a RESTful API with well-defined URLs and standard HTTP methods (GET, POST, PUT, DELETE) for CRUD operations.
  + Uses WebSockets for real-time data updates to provide current device locations.
* Database Design (MySQL): The relational database schema is organized into several interconnected tables:
  + users table: Stores user accounts, including roles and hashed passwords.
  + devices table: Holds details about each tracked device.
  + locations table: Stores historical geographical coordinates (latitude, longitude, timestamp) reported by devices, along with optional data like speed, accuracy, and altitude.
  + geofences table: Defines geographical boundaries with names, shapes (e.g., GeoJSON), and types.
  + tasks table: Stores information about assigned tasks (title, description, location, status).
  + assignments table: Links tasks to devices or drivers.

3. Statistical Data & Models Used:

* The project utilizes Geographic Information Systems (GIS) for visualizing movement data.
* The implemented system primarily focuses on real-time location tracking and data management. While the literature survey mentions machine learning techniques for stock price prediction (which appears to be an error in the document, as the project is about device tracking), the current system description does not detail the use of complex statistical models or machine learning algorithms for tasks like predictive analytics or anomaly detection. These are listed as potential areas for future enhancement.

4. Skills Used:

* Programming Languages: JavaScript (for both Frontend and Backend).
* Frontend Development: React, Chakra UI, React Router DOM, React Hooks.
* Backend Development: Node.js, Express.js, RESTful API Design.
* Database Management: MySQL, SQL.
* Security: bcrypt for password hashing, JSON Web Tokens (JWT) for authentication.
* Real-time Communication: WebSockets.
* Mapping & GIS: Leaflet.js, React-Leaflet, and understanding of GIS concepts for movement visualization.
* System Design: Client-server architecture, MVC architectural patterns, modular development.
* Problem Solving: Addressing challenges related to real-time data handling, scalability, and data integrity.
* Hardware & Software Management: Familiarity with typical hardware and software requirements for system development and deployment (e.g., Linux servers, Node.js, MySQL, web browsers).

Certification:

1. AI and Machine Learning Full Course - IBM Skillsbuild – 2023
2. Google Data Analytics - Coursera (Google) – 2024

This professional certificate program is designed to provide a comprehensive understanding of the practices and processes employed by a junior or associate data analyst in their daily work.

Key Concepts & Objectives:

* Data Analyst Role Immersion: Gain an in-depth understanding of the day-to-day responsibilities, practices, and processes of a junior or associate data analyst.
* Core Analytical Skills: Master essential analytical skills including data cleaning, data analysis, and data visualization.
* Key Analytical Tools: Learn to effectively use industry-standard tools such as spreadsheets, SQL, R programming, and Tableau.
* Data Preparation & Analysis: Understand the methodologies for cleaning and organizing data for effective analysis, as well as how to perform detailed analysis and calculations using spreadsheets, SQL, and R programming.
* Data Visualization & Presentation: Acquire the skills to visualize and present data findings clearly and compellingly in dashboards, presentations, and commonly used visualization platforms.

Skills Gained:

* Tableau Software
* Spreadsheet Software
* Data Cleansing
* Stakeholder Communications
* Data Analysis
* Professional Networking
* Data Presentation
* Data Storytelling
* Rmarkdown
* LinkedIn
* Data Validation
* Sampling (Statistics)

1. Machine Learning in Production - Coursera (DeepLearning.AI) – 2025

This certification focuses on the practical aspects of deploying and maintaining machine learning models in real-world production environments.

Key Concepts & Objectives:

* ML Project Lifecycle & Pipeline: Identify and understand the essential components of a machine learning project lifecycle and its corresponding pipeline.
* Deployment & Monitoring Patterns: Learn to select the most suitable deployment and monitoring strategies for various production scenarios.
* Model Performance Optimization: Acquire techniques to optimize model performance and metrics by focusing on disproportionately important examples and key slices within a dataset.
* Data Challenges in Production: Address common production challenges related to structured, unstructured, small, and big data.
* Label Consistency: Understand the critical importance of label consistency in ML datasets and discover methods to improve it for better model reliability.

Skills Gained:

* Machine Learning
* Software Development Life Cycle
* MLOps (Machine Learning Operations)
* Application Deployment
* Debugging
* Applied Machine Learning
* Continuous Monitoring
* Data Pipelines
* Continuous Deployment
* Data Quality

1. Cloud ML Engineer - Coursera (Google Cloud) – 2025

This program is designed to equip individuals with the necessary skills for a machine learning engineering role and to prepare them for the Google Cloud Professional Machine Learning Engineer certification exam.

Key Concepts & Objectives:

* ML Engineering Role Skills: Acquire the fundamental and advanced skills required to excel as a machine learning engineer in a professional setting.
* Certification Exam Preparation: Specifically prepare for the Google Cloud Professional Machine Learning Engineer certification exam, covering all relevant domains and topics.
* ML Model Productionization on Google Cloud: Understand how to design, build, and productionalize (deploy and manage in production) machine learning models using various Google Cloud technologies to solve real-world business challenges.
* Certification Purpose & Relationships: Gain clarity on the purpose and value of the Professional Machine Learning Engineer certification and how it relates to other Google Cloud certifications within the broader Google Cloud ecosystem.

Skills Gained:

* Data Management
* Data Pipelines
* Prompt Engineering
* Data Governance
* Apache Airflow
* Dataflow
* Application Deployment
* Cloud Infrastructure
* MLOps (Machine Learning Operations)
* Workflow Management
* Hybrid Cloud Computing
* Cloud Platforms

Technical Skills:

* Programming Languages: Python, C++, JavaScript
* Core Concepts: Object-Oriented Programming (OOP), Data Structures, Algorithms
* Artificial Intelligence & Machine Learning:
  + Frameworks: TensorFlow, Keras, PyTorch (backend for Transformers)
  + Models/Techniques: LSTM, Random Forest, XGBoost, BERT, RoBERTa, Ensemble Learning, Black-Scholes Model
  + Libraries: Scikit-learn, XGBoost, Keras, joblib (for model persistence)
* Natural Language Processing (NLP): NLTK, SpaCy, Gensim, TextBlob, Transformers
* Data Analysis & Visualization: Pandas, NumPy, Matplotlib, Seaborn, SciPy, R, SQL, Tableau, Spreadsheets
* Database Management: MySQL (Workbench), MongoDB, SQLAlchemy, Pymysql
* Operating Systems: Basic to Intermediate understanding of OS fundamentals
* Version Control: Git

Soft Skills:

* Communication: Clearly articulate complex technical concepts; effective interpersonal and team collaboration. (Evidenced by Event Organizer role, project teamwork)
* Problem-Solving: Proactively identify challenges and develop innovative solutions, as demonstrated in project development and debugging. (Evidenced by all projects)
* Analytical Thinking: Strong ability to analyze data, identify patterns, and derive insights for model development and decision-making. (Evidenced by all projects, especially data analysis in Stock Prediction, Recommendation Engines, Author Identification)
* Adaptability & Learning Agility: Enthusiastic to learn new technologies and take on new problems, as shown by rapid skill acquisition in various libraries and ongoing research in AI development tools. (Evidenced by range of skills, ongoing research, certifications)
* Organization & Proactiveness: Highly organized in managing project components and proactive in seeking enhancements and new knowledge. (Evidenced by structured projects, research initiative, event organization)
* Teamwork & Collaboration: Experience working effectively within teams to achieve common goals. (Evidenced by Event Organizer role, collaborative nature of software development projects).

Extracurricular Activities:

Event Organizer at Jain Deemed-to-be University - 2023 to Present

Collaborated with a team to plan and execute various college events and fests, gaining experience in coordination, communication, and problem-solving.

Languages:   
English, Kannada, Hindi