

# ***System Requirements Specification Report***

**BACHELOR OF TECHNOLOGY**

**in**

**COMPUTER SCIENCE & ENGINEERING**

**by**

**Piyush Gupta - 500106034**

**Kartikey Bhasin - 500106973**

**Aryan Rana - 500107156**

**Parineeta Dhanial – 500105658**

**under the guidance of**

**Dr. Shaurya Gupta**

**Associate Professor**



**School of Computer Science**

**University of Petroleum & Energy Studies**

**Bidholi, Via Prem Nagar, Dehradun, Uttarakhand**

**(AUG-DEC 2024)**

**System Requirements Specification (SRS)**

# 1. Introduction

## 1.1 Purpose

The purpose of the "University Feedback Analysis" system is to focus on analysing string data collected from CSV files. The system will process this data using statistical methods and sentiment analysis to generate comprehensive reports. These reports will assist university administrators in making informed decisions related to curriculum, faculty, and infrastructure, ensuring anonymity and data security.

## 1.2 Scope

The scope of the "University Feedback Analysis" project includes analyzing student feedback data from CSV files, applying statistical analysis and sentiment detection to uncover trends and insights. The project will generate detailed reports for university administrators, highlighting areas of improvement and strengths in teaching, curriculum, and infrastructure. The system will ensure data privacy and maintain feedback anonymity while focusing on efficient data processing and visualization.

## 1.3 Definitions, Acronyms, and Abbreviations

- **CSV** – Comma-Separated Values: A file format used for storing tabular data, where values are separated by commas.
  - **NLP** – Natural Language Processing: A branch of artificial intelligence used to analyze and interpret human language, such as sentiment analysis.
  - **Sentiment Analysis** – A method to determine the emotional tone behind a series of words, used to understand opinions and feedback.
- 

# 2. Overall Description

## 2.1 Product Perspective

The system provides insights and detailed reports to university administrators –

- **Input:** CSV files containing feedback data.
- **Processing:** Statistical analysis and sentiment analysis to derive insights.
- **Output:** Comprehensive reports highlighting strengths and areas of improvement.
- **Interfaces:** A straightforward method for loading CSV files and generating reports for administrators.

## 2.2 Product Functions

The system will include:

1. **CSV Data Ingestion:** Ability to upload and read feedback data stored in CSV format, ensuring compatibility with various feedback collection methods.
2. **Data Preprocessing:** Cleansing and organizing the raw data from CSV files, such as handling missing values, standardizing formats, and organizing the feedback for further analysis.
3. **Statistical Analysis:** Perform statistical computations on the feedback data to identify patterns and trends (e.g., average ratings for courses or instructors).

4. **Sentiment Analysis:** Apply natural language processing (NLP) techniques to identify the sentiment (positive, negative, or neutral) of textual feedback to understand overall student satisfaction.
5. **Report Generation:** Automatically generate comprehensive reports that include statistical summaries, sentiment trends, and graphical representations of feedback, providing actionable insights.
6. **Data Visualization:** Visualize feedback patterns using charts, graphs, and other tools to highlight trends, strengths, and areas that need improvement.

## 2.3 Design and Implementation Constraints

- Reliance on structured CSV files may lead to analysis issues if there are inconsistencies in format or missing values.
  - The system's performance could be impacted by large datasets, requiring optimization for efficient processing.
  - Compliance with data protection regulations is essential to ensure the confidentiality and security of student feedback.
- 

## 3. Functional Requirements

### 3.1 Data Loading and Processing

- **CSV File Upload:** The system shall allow users to upload CSV files containing student feedback data through a simple interface.
- **Data Validation:** The system shall validate the uploaded CSV file for correct format, checking for proper delimiters, missing values, and required columns.
- **Data Preprocessing:** The system shall preprocess the validated data by cleaning, organizing, and transforming it into a structured format suitable for analysis.
- **Error Handling:** The system shall provide clear error messages for any issues encountered during file upload or data validation, guiding users on necessary corrections.

### 3.2 Data Analysis and Reporting

- **Statistical Analysis:** Analyze feedback data to compute key metrics like average ratings, using Java libraries for efficient calculations.
  - **Sentiment Analysis:** Implement natural language processing techniques to classify feedback sentiment (positive, negative, neutral) using existing libraries.
  - **Report Generation:** Automatically generate comprehensive reports summarizing statistical findings and sentiment results, incorporating visualizations with libraries like JFreeChart.
  - **Customization of Reports:** Allow users to customize report parameters, such as specific courses or time frames, and enable exporting reports in formats like PDF or Excel for easy sharing.
- 

## 4. Non-Functional Requirements

### 4.1 Performance Requirements

- The system should have a response time of no more than 2 seconds for user actions such as file uploads, data processing initiation, and report generation.
- The system must handle up to 500 feedback submissions per hour without performance degradation.
- The system should be able to process datasets of up to 10,000 entries within 5 minutes for efficient analysis and reporting.

## **4.2 Usability Requirements**

- The system should provide clear and detailed error messages during data loading and processing to assist users in resolving issues quickly.
- Documentation should be available to explain the data analysis processes, statistical methods, and sentiment analysis techniques used in the system.
- The system should ensure that the analysis results and generated reports are presented in a clear and comprehensible manner for users.
- Users should be able to easily customize the parameters for analysis, such as selecting specific courses or time periods for generating reports.

## **4.3 Reliability Requirements**

- The system should maintain data integrity during loading, processing, and analysis, ensuring that the results accurately reflect the input data.
- The system must have backup mechanisms to prevent data loss in case of unexpected failures during processing or report generation.
- The system should be able to recover from errors or interruptions without losing any data or requiring extensive user intervention.
- The data analysis processes should produce consistent results across multiple runs with the same input data, ensuring reliability in findings.

---

# **5. System Architecture**

## **5.1 Overview**

The system architecture will be based on a Java application, utilizing Java libraries for data processing, analysis, and reporting.

A modular design will separate data loading, analysis, and reporting components to enhance maintainability and scalability.

## **5.2 Components**

- **Data Processing Component:** Utilizes the OpenCSV library to read and validate CSV files, storing the data in an ArrayList for further analysis.
- **Sentiment Analysis Component:** Integrates Stanford CoreNLP to analyze the sentiment of each feedback entry, categorizing them into positive, negative, or neutral sentiments.
- **Data Visualization Component:** Employs JFreeChart to create visual representations of the feedback data, such as bar charts, allowing for easier interpretation of results.
- **Error Handling Component:** Captures exceptions during data loading and processing, providing error messages to assist in troubleshooting and ensuring system reliability.