Software Requirements Specification

for

RelativeEase:
A Relative Grading Software for Professors

Prepared by Arshad Ahmed, Sanjana Gummuluru,
and Harshel Swain

CSPE41 project, Spring 2024
National Institute of Technology, Tiruchirappalli

Table of Contents

1. Introduction

1.1 Purpose

This SRS document presents a detailed description of the RelativeEase software, a relative grading software aimed at university professors. It represents the client requirements analysis that defines the functional and nonfunctional requirements of the software and its different functionalities. It defines the abilities, reactions from stimuli, guidelines and limitations of the system. This document will be complete in its scope of the software and the functions required. The system provides a solution to allow the user to assign grades with ease satisfying the user criteria, allowing for customization of the method, pushing and pulling students' grades, setting limits, and finally a downloadable record of the class grades along with the final data of cutoffs and number of students per grade.

1.2 Document Conventions

This document follows the IEEE format standard (IEEE Std. 830 – 1998).

1.3 Intended Audience and Reading Suggestions

The intended audiences of this document are the respected Professors of NIT-T, software engineers, the Spring 2024 CSPE41 Software Engineering class and for anyone who has interest in software engineering.

1.4 Project scope

The software is intended for use by NIT-T professors who wish to automate the process of grading their students relatively. It offers several different options for the process, along with custom inputs for pushing/pulling students to adjacent grades, and floor/ceil of decimal marks.

2. Product Perspective

This project represents the initial version of an Automated Grading System for Faculty. All requirements listed herein describe a self-contained system. This project will allow users to segregate and grade students according to their performance, score and the category they fall into according to the standards of the university. The goal is to allow Professors to grade and categorize students more efficiently and conveniently, saving efforts and valuable time.

2.1 Product Features

The system shall allow anyone with the application to convert the document according to their grading requirements. There are no login requirements as such.

The system checks if all required data is provided and then will prompt the user to enter additional information, if required. After all required information is provided, the system generates a pdf (.pdf) or excel sheet (.xlsx) depending on the user's preferences. The resultant file is protected by a password for confidentiality and privacy purposes.

2.1.1 Upload excel file

The user needs to upload an Excel file in the following format:

- Document type: Microsoft Excel Sheet. (.xlsx)
- Column 1: Roll no
- Column 2: Name
- Column 3: Assessment-1 Score
- Column 4: Assessment-2 Score
- Column 5: Assignment Score
- Column 6: End Semester Score
- Column 7: Total Score

2.1.2 Set Requirements

The user needs to set the following prerequisite requirements:

Total number of students:

The user needs to input the total number of students for which he/she needs to calculate the grades.

Percentage per Category:

The user needs to input the percentage of students he wishes to assign grades for each grade category.

For example, S Grade: 7.6% of students.

Gap between Grades:

The user has to decide the gap between the grades category of students to avoid clash and error.

Push and Pull records:

The user can Push and Pull the marks of respective students according to their need and requirements.

Floor and ceil:

The user can Floor or Ceil respective decimal values and scores of students according to his will.

For example, floor values below the 0.5 range and ceil those above.

Marks Manipulation:

If the user desires to manipulate records or separate records with the same total marks based on a priority, they can do so.

3. System Features

This section provides detailed requirements for the application design, including functional requirements.

3.1 General Requirements

The system shall allow anyone with the application to convert their document according to their grading requirements.There are no login requirements as such.

The system checks if all required data is provided and then will prompt the user to enter additional information, if required. After all required information is provided, the system generates a pdf and/or excel sheet, along with the statistical record of students per grade and its corresponding graph.

Inputs: The Excel sheet/file provided by the user in the given format.

Source: All inputs are provided by the users.

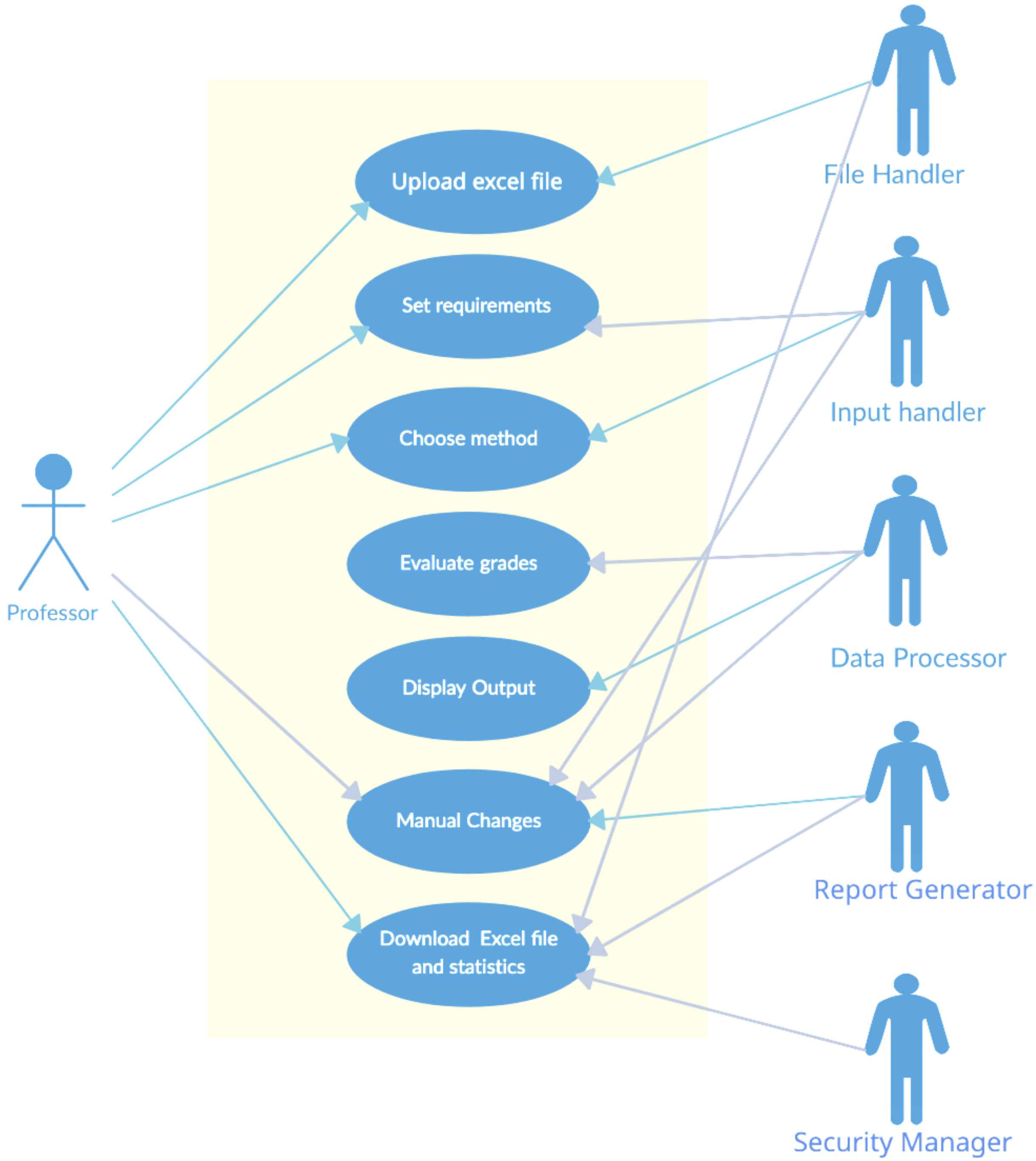
Outputs: Indication that the user has the output from the system.

Requires: The user must provide the Data in the .xlsx format, according to the aforementioned column specifications.

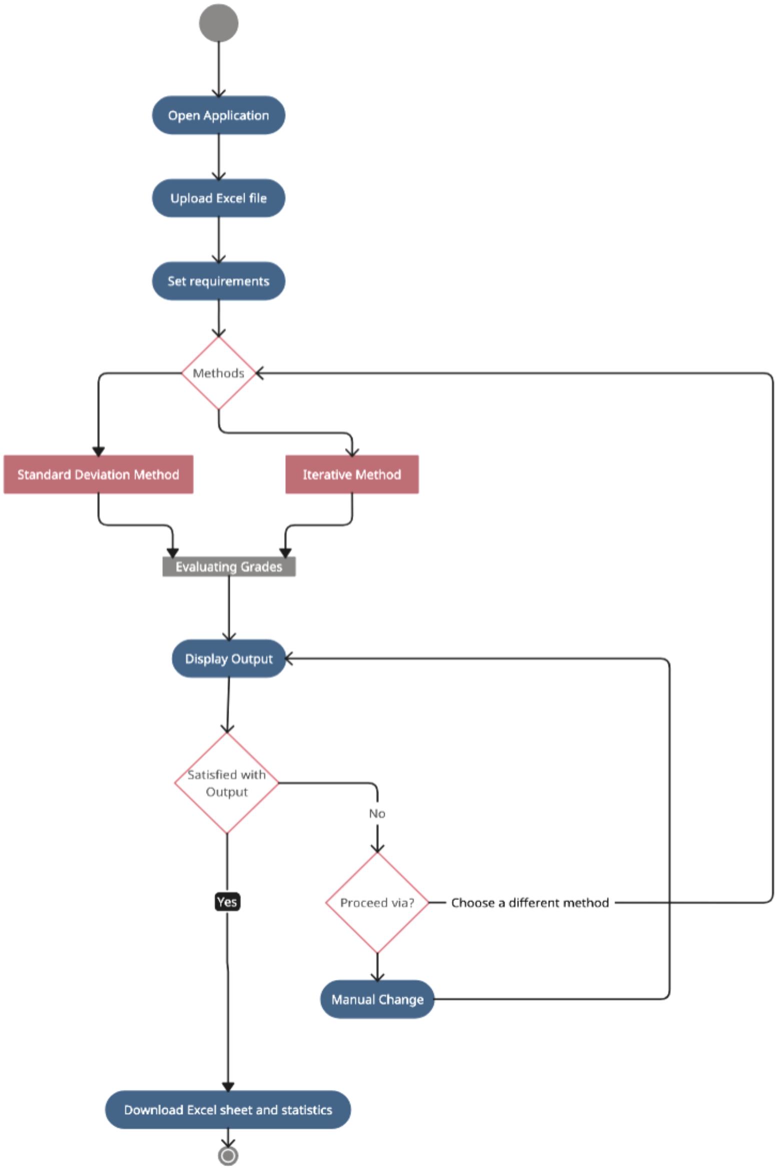
Pre-Conditions: The user must have the input document in the form of an excel sheet.

Post-Conditions: The user can get irrelevant outputs and error statements if the above conditions are not met.

Use Case:


Upload excel file
Set requirements
Choose method
Evaluate grades
Display Output
Manual Changes
Download Excel file and statistics
Professor
File Handler
Input handler
Data Processor
Report Generator
Security Manager

Activity Diagram:



1. Open Application

2. Upload Excel File: The user needs to upload the excel file in the following format:

- Column 1: Roll no
- Column 2: Name
- Column 3: Assessment-1 Score
- Column 4: Assessment-2 Score
- Column 5: Assignment Score
- Column 6: End Semester Score
- Column 7: Total Score

3. Set Requirement: The user needs to set the following values according to his/her requirements:

1. Total number of students
2. Percentage per Category
3. Gap between Grades
4. Push and Pull records
5. Floor and ceil
6. Marks Manipulation

4. Methods: The user has the liberty to choose the method using which they would like to grade their students.

1. Standard Deviation Method
2. Iterative Method

5. Evaluating Grades: According to the method selected by the user, the grades will be evaluated and recorded.

6. Display output: The following output will be displayed

1. The number of students in a particular grade category.
2. The percentage of total no. of students per grade category.
3. Graph of grade vs number of students attaining it will be displayed.

7. Satisfied with output: If the user is not satisfied with the output provided by the system they can resort to the following options:

1. The user can select another method and re-evaluate the result.
2. The user can make the changes manually according to their requirement.

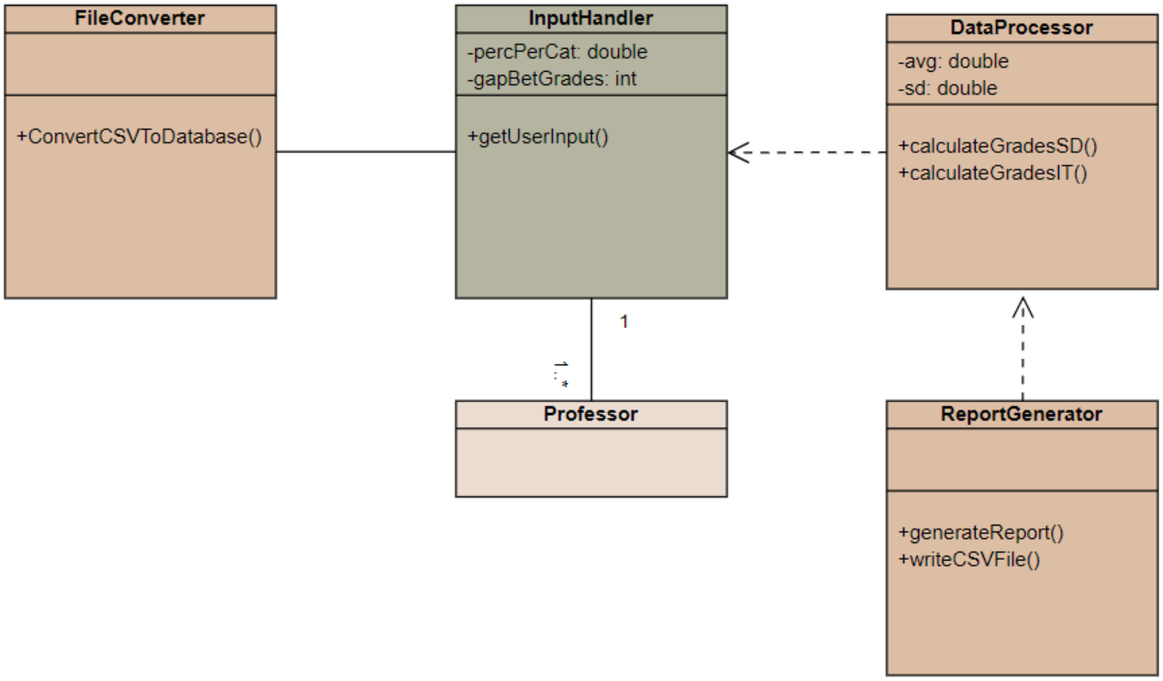
8. Download the output: The user can download the output data if they are satisfied with the results provided by the application.

The user can download the data in the following format/file type:

1.Microsoft Excel file.

2.PDF file.

Class Diagram:



The design of the system is composed of 4 different **Classes.**

1. File Converter.
2. Input Handler.
3. Data Processor.
4. Report Generator.

File Converter: The file converter converts a CSV file to a table in a database by parsing through each row ,extracting the tokens and inserting them into respective columns.

Input Handler: The input handler handles the aforementioned inputs provided by the user.

Data Processor: Data processor calculates, evaluates and assigns grades for the data input provided by the user according to the method selected by the user i.e

1.Standard Deviation Method.

2.Iterative Method.

Report Generator: The report Generator converts back the table into a CSV file.

4. Interface

4.1 User Interface

The software has a clean, easy-to-use interface that allows professors to grade students with just the click of a few buttons

4.2 Communication

The C++ application communicates with a MySQL Database using the MySQL Connector / C++ Library. This library provides a C++ API for interacting with MySQL databases.

The result of the SQL query is retrieved as a ResultSet object, which allows the application to iterate over the rows returned by the query and retrieve the data.

Exception handling is implemented to catch and handle any errors that may occur during the database connection and query execution.

5. Security Requirements

- The Resultant data file will be password protected.
- All exchanges from client to database involving private data shall occur using the highest available level of secure connection.