



National Institute of Technology Uttarakhand

GRADING SYSTEM AUTOMATION

Software Requirement Specification (SRS) Document

Submitted To: -

Dr. Maheep Singh
(Assistant Professor)
Computer Science Department
National Institute of
Technology, Uttarakhand

Submitted By: -

Anupam Panwar	BT19CSE002
Neha Dhyani	BT19CSE003
Purvi Goyal	BT19CSE011
Priyanshu Upadhyay	BT19CSE020

Table of Content

1. Introduction

- 1.1 Purpose
- 1.2 Document Convention
- 1.3 Intended Audience
- 1.4 Scope
- 1.5 Existing System
- 1.6 Proposed System

2. Overall Description

- 2.1. Product Perspective
- 2.2. Product Features
- 2.3. Constraints
- 2.4. Operation Environment
- 2.5. User Characteristics
- 2.6. Flow Chart

3. Requirements

- 3.1. Software Requirements
- 3.2. Hardware Requirements
- 3.3. Functional Requirements
- 3.4. Non-Functional Requirements

1. Introduction

1.1. Purpose

The purpose of this software will be to automate the existing system of grading which is currently done manually by the course coordinator.

1.2. Document Convention

- HTML: Hypertext Markup Language
- CSS: Cascading Style Sheets
- JS: JavaScript

1.3. Intended Audience

This software aims at engaging maximum number of course coordinator in the institute so that their work can be eased out within few steps.

1.4. Scope

- Manual entering of the data fields (i.e. Name of the student, Roll number, etc) into the system for fabrication of the control sheet
- It will provide detailed information about various fields (i.e. Marks obtained by the student, Grade obtained, Grade window, etc)

1.5. Existing System

The present system is a manual system involving paper work in the form of maintaining various files. A manual system has following limitations

- It is time consuming
- High risk in manual entering of student data as any error will cause change in grade distribution
- Chances of human and computational errors are high
- Requires high management of data, less secure
- Compilation of result is a tiresome process

1.6. Proposed System

The fully automated grading system aims at reducing the time and effort consumed in compilation of result. The system provides many advantages

- Secure- editing can be done only by registered users
- The files should be stored in centralised database maintained by the system
- Reduces error- the computation part is automated so no chance of computational error
- Editing can be done in the grade sheet
- User will provide the grading window and the result will be generated according to that

2. Overall Description

2.1 Product Perspective

The proposed automated grading system will generate the grades of students using the inputs from the course coordinator.

2.2 Product Features

The system provides following features

- Automation of calculation of total marks of students in a course
- Grading based on the grading window provided by course coordinator and total marks obtained
- Course details can be saved in a pdf file
- Personalised account for every user
- compatible with any device, more optimal with a PC
- Course details are editable
- Deletion of older course files

2.3 Constraints

- Once an individual account is created by any user, there is no option of the deletion of the account.
- Internet connection would be required.
- Only the account holders can edit/delete the information in the website.

2.4 Operating Environment

The product will be operating in Windows OS, Linux OS, Mac OS. The Grading System Automation is a website and will operate on all famous browsers like Chrome, Mozilla Firefox. It can be operated on mobile phones. Internet connection would be required for the product.

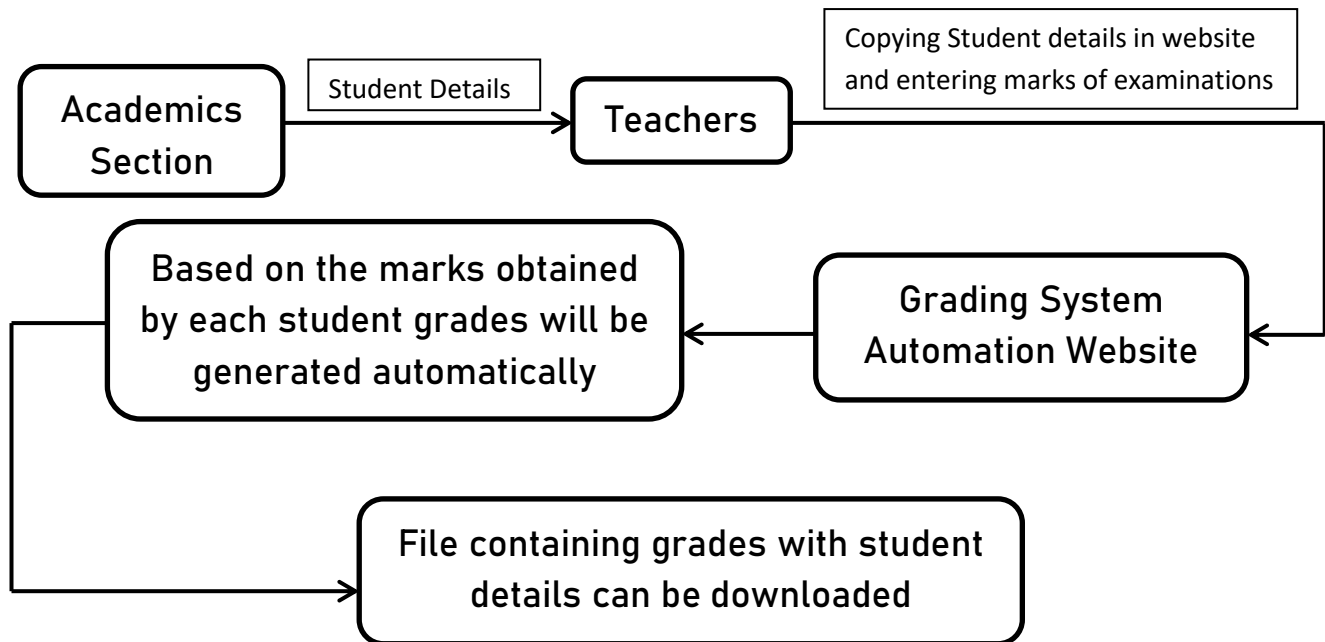
2.5 User Characteristics

This product is designed for Teachers to ease the process of assigning grades to the students.

The features that are available to the users are:

- They can create their personal accounts.
- They can add the details of the students (Name, Roll No.)
- They can edit/delete the information whenever they want.
- They can search for a particular file.
- They can download the information in the local system.

2.6 Flow Chart



3. Requirements

3.1 Software Requirement

Front End

- Html
- CSS (Bootstrap)
- JavaScript
- React JS

Back End

- Mongo DB
- Express JS
- Node JS

3.2 Hardware Requirement

- Any device with proper internet connection and a web browser

3.3 Functional Requirement

Register/Sign Up

- First time users will need to register on the website.
- Input: Users can sign up using Google authentication as well as by providing Name, Email (nituk domain), Contact No. and a password.
- Output: Confirmation of registration. Or if any error is encountered an error message is displayed and registration will be cancelled.

Login

- Input: User enters the email and the password or logins with google authentication.
- Output: User will be let in after authentication.

Data Entry

- Input
 - An add (+) button will let user enter course details namely: course name, course code, course type (lecture or lab), course co-ordinator.
 - Further User can enter the name of exam (like mid-term 1 or class test), maximum marks, student name marks obtained by the student, and Grade window (optional).
- Output
 - When details are entered through the add button, a new class will be created and named as course code provided. (these classes will contain all the information about the students)

- Totals marks will be generated by the website when use submits the details. If grade windows are provided grade will be generated too.

Print

- The application lets the user print the pdf format of each class/course details.

Personalised account

- Each user will have a personalised compilation of each course.

Edit

- If user need to edit any information of any student, it can be done by the edit option which can be used to edit the whole grade sheet at once.

Deletion

- If one wishes to delete course details, a delete option will let the user do so.
- An alert will be displayed for confirmation to delete the intended class before its deletion.

3.4 Non – Functional Requirement

Performance Requirement

- The changes made are reflected to the database within seconds. Retrieving information is simple and convenient.

Usability Requirement

- The WebApp should allow users to access the resources using a proper internet connection via any browser.

Availability Requirement

- Application shall be available to users at all times.

Security Requirement

- Only verified users will be able to access the database and one user cannot view other user's activity and class details.
- Google auth will provide additional security so that third party do not get unnecessary access to the data.