Pheba Colfer

EKU 11/17/2022

STUDENT RECORD MANAGEMENT SYSTEM

CSC 440 Applied Software Engineering

Individual Project

Final Report

Contents

[I. Introduction 2](#_Toc119589058)

[A. Problem Statement 2](#_Toc119589059)

[B. Proposal 2](#_Toc119589060)

[II. System Description 2](#_Toc119589061)

[III. System Requirements 2](#_Toc119589062)

[A. Functional Requirements 2](#_Toc119589063)

[B. Non-functional Requirements 5](#_Toc119589064)

[C. Domain Requirements 6](#_Toc119589065)

[IV. Use Case Diagram 6](#_Toc119589066)

[I. Data Flow Diagrams 6](#_Toc119589067)

[VI. Structure Flow Chart 10](#_Toc119589068)

[I. Database Design 11](#_Toc119589069)

[A. ER Diagram 11](#_Toc119589070)

[B. Table Schema 11](#_Toc119589071)

[VII. Algorithm Design 11](#_Toc119589072)

[Delete Grade 11](#_Toc119589073)

[Add New Grade 12](#_Toc119589074)

[VIII. Conclusion 13](#_Toc119589075)

# 

# Introduction

## Problem Statement

EKU’s Registrar Office needs a software system that allows for easier access to and management of student grades in courses offered at EKU.

## Proposal

We propose a software system that allows grades to be added, changed, and printed for each student and course.

# System Description

EKU’s Registrar Office needs to update student course records after receiving student grades from all of the faculty at the end of each semester. Each faculty member provides the office an Excel file with letter grades (A, B, C, D, or F) of the students who took course(s) with them. The office is asking you to develop a system to manage the data for them. They need you to develop a software system to do the following tasks:

Add new grades of courses for each student to a database: The office can gather all the Excel files from the faculty, and put them in a folder. The formats of the names for the folder and Excel files are:

Folder name: “Grades [Year] [Semester]”

(Ex., “Grades 2021 Spring”, “Grades 2023 Fall”)

File name: “[Course Prefix] [Number] [Year] [Semester]”

(Ex., “CSC 440 2021 Fall”, “MAT 234 2027 Spring”)

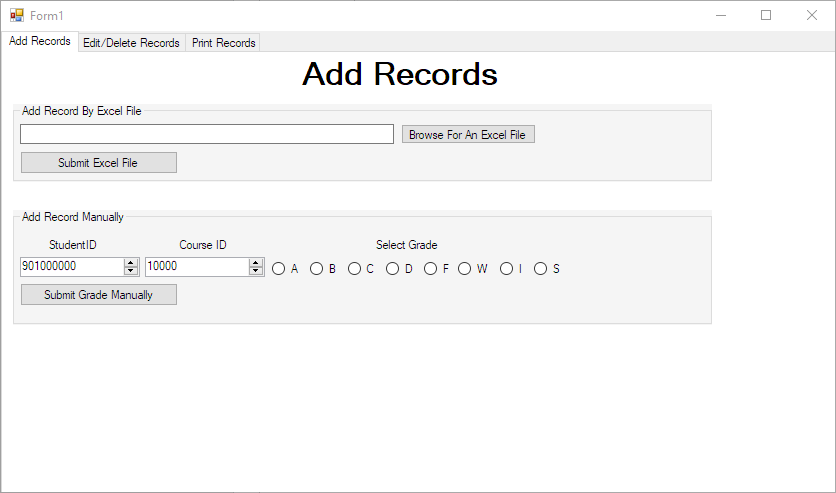
Edit a grade for a student: There might be some mistakes in the Excel files provided by the faculty. The system shall allow the office to edit the grade of a course for a student. Editing a grade means changing the grade, deleting the grade, and/or adding a grade to the database.

Print a report card (or transcript) for a student: The system shall let them print a report card (or transcript) for a selected student. The report card (or transcript) should list the student’s name, id, overall gpa, and a list of courses with grades that he/she has taken before.

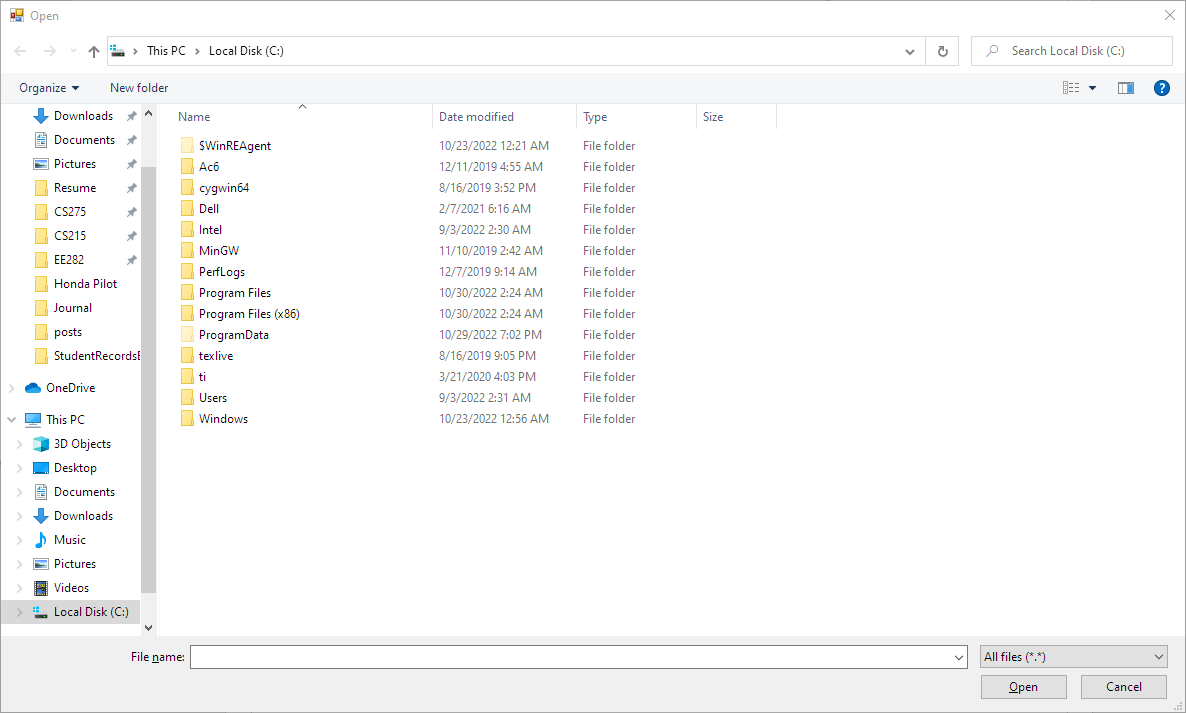
# System Requirements

## Functional Requirements

1. The system shall allow a user to add grades for a student from an Excel file.
   1. The user shall navigate to the ‘Add Records tab’
   2. The system shall present the employee with an interface to add grades

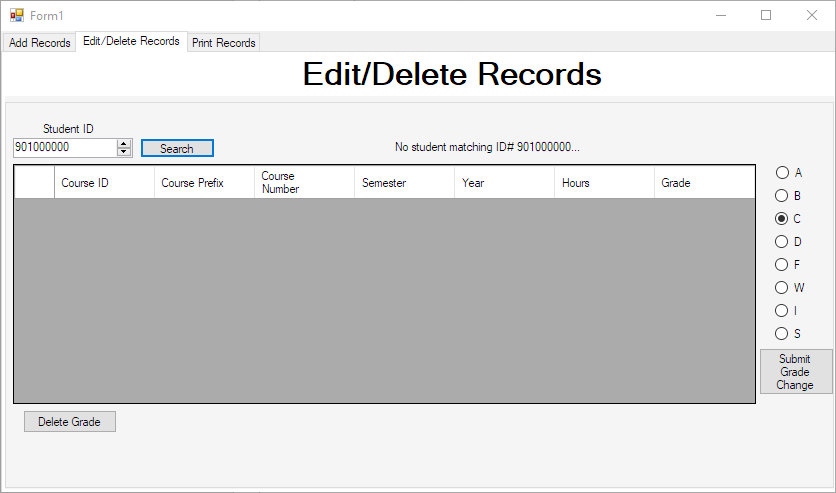


* 1. The user shall be able to select ‘Browse For an Excel File’
  2. The system shall present a prompt for a file selection.



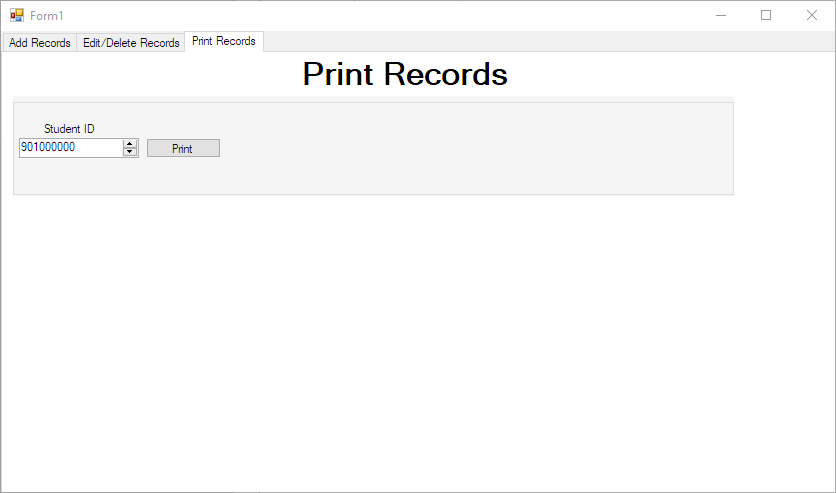
* 1. The user shall be able to import an Excel file from this interface by selecting the ‘open’ button
  2. The pass/fail grade import status shall be presented to the user via a pop-up box.

1. The user shall be able to import grades manually
   1. From the interface in 1.2, the user shall be able to enter a student ID, course ID, and a grade to add to the database.
   2. A popup box shall be displayed to inform the user about the success or failure of adding a new grade.
2. The user shall be able to edit / delete records
   1. The user shall navigate to the ‘Edit/Delete Records’ tab.
   2. The system shall present the user with the edit grade interface.



* 1. The user shall enter a student ID into the search box.
  2. The system shall validate the user ID
     1. If the user ID is not found in the system, an error message will be presented to the user
     2. If the user ID is found in the system, the table of user grades shall be populated with course information.
  3. The user shall select a course listing from the table of entries
  4. The user shall select an option to edit or delete a grade
     1. If the user selects “Delete Grade”, the grade will be removed from the system
     2. If the user selects “Submit Grade Change”, the grade shall be updated in the system
  5. The system shall pop-up a confirmation of the action and update the view accordingly.

1. The system shall allow a user to print a student’s transcript.
   1. The user shall select the “Print Records” tab
   2. The system shall present the user with an interface to enter a student ID.



* 1. The user shall enter a student ID
  2. The system shall validate the user ID
     1. If the user ID is not found in the system, an error message will be presented to the user
     2. If the user ID is found in the system, a report will be generated for the student and opened.

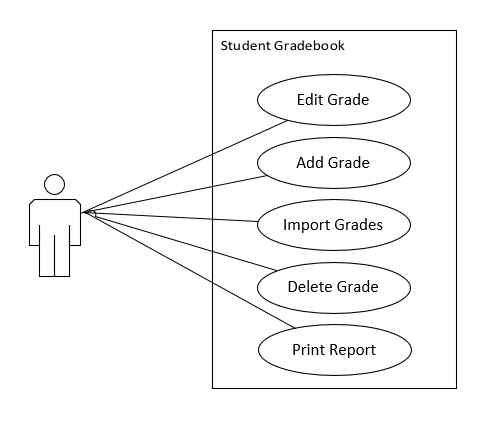
## Non-functional Requirements

1. The system shall allow users to upload electronic folders of Excel spreadsheets containing the students that took each course, along with the grades they received in the course.
2. The system shall prevent unauthorized users from accessing or changing the information within the database.
3. The system shall prevent employees from performing invalid modifications to courses or students.

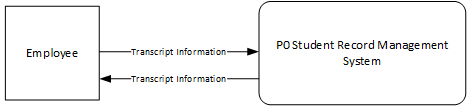
## Domain Requirements

1. The software shall be compatible with the personal computing devices used by the company.
2. The system shall remain functional for as long as the Registrar Office decides to use it.

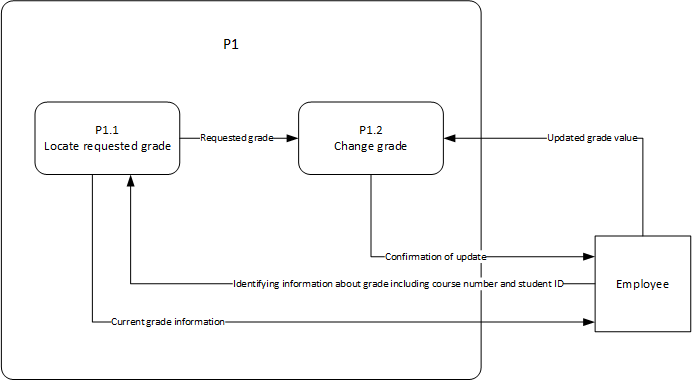
# Use Case Diagram

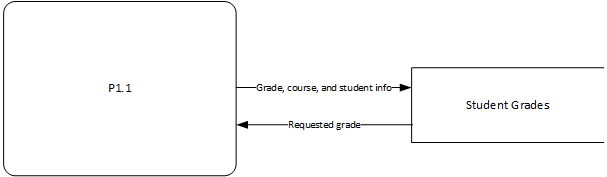


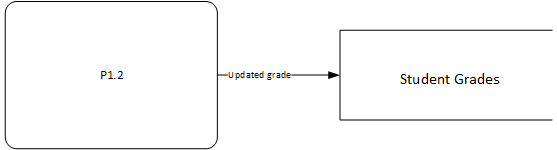
# Data Flow Diagrams

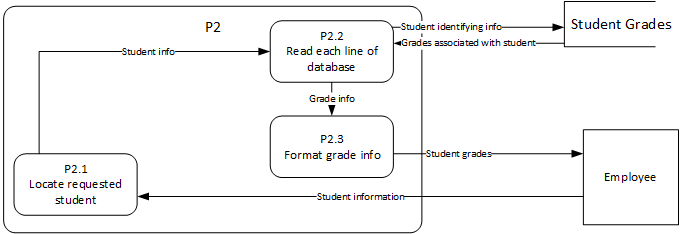


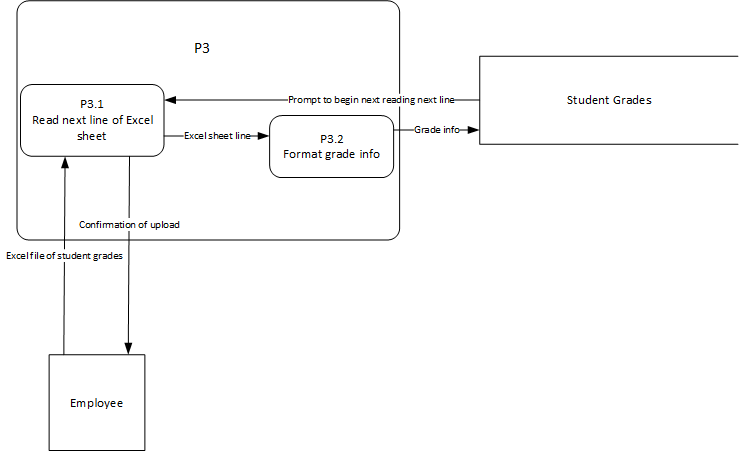


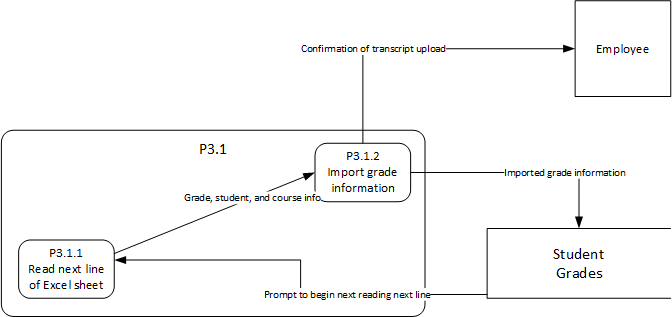


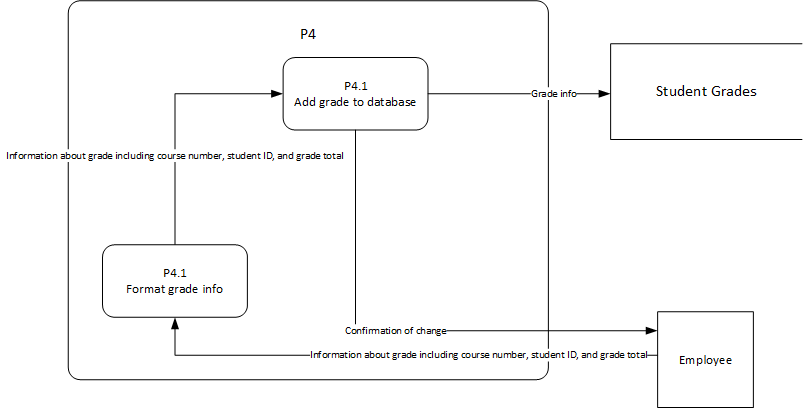










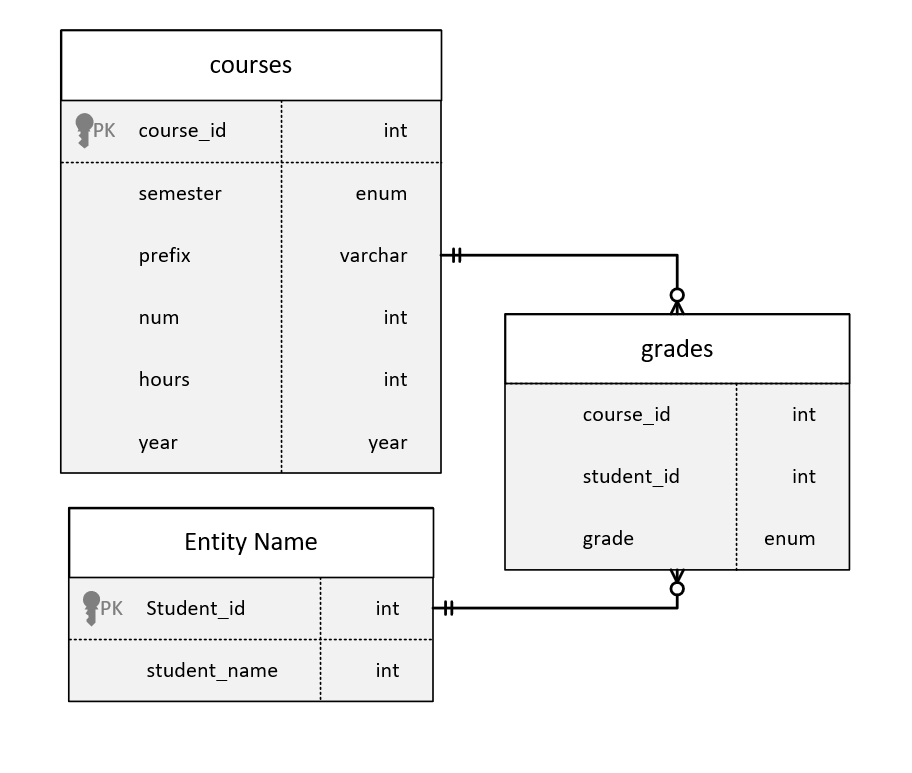


# Structure Flow Chart



# Database Design

## ER Diagram



## Table Schema

grades

|  |  |  |
| --- | --- | --- |
| student\_id | course\_id | grade |

students

|  |  |
| --- | --- |
| Student\_id | student\_name |

courses

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| course\_id | semester | prefix | num | hours | year |

# Algorithm Design

## Delete Grade

Function main:

If UserInput = delete grade

Call DeleteGrade

End Function main

Function DeleteGrade:

Initialize variable CurStudentID

Initialize variable CurCourseID

Set CurStudentID to value entered for student ID

Set CurCourseID to value entered for courseID

Search grades database for entry where student\_id = CurStudentID AND course\_id = CurCourseID

If matching entry is found:

Delete entry from database

else:

Return error message

If successfully deleted

Return confirmation message

else

return error message

## Add New Grade

Function main:

If UserInput = add grade

Call AddGrade

End Function main

Function AddGrade:

Initialize variable CurStudentID

Initialize variable CurCourseID

Initialize variable CurGrade

Set CurStudentID to value entered for student ID

Set CurCourseID to value entered for courseID

Set CurGrade to value entered for course grade

Create a new entry in grades database where student\_id = CurStudentID AND course\_id = CurCourseID AND grade = CurGrade

If successfully added

Return confirmation message

else

return error message

# Conclusion

The EKU Registrar office has requested a transcript management system that allows employees to easily upload, delete, change, and search for student transcripts. Our proposal is a graphical student record management system that allows employees to access and change student grades as well as import Excel files containing student grades. In this specific individual project, a non object oriented approach to solving this problem will be taken, and the previous diagrams and charts illustrate how this approach will be used to fulfill the needs of the customer.