OOPS PROJECT

ACADEMIC REPORT MANAGEMENT SYSTEM

Pasnoori Nymika(2018BCS-032)
Laudya Mukesh Kumar(2018BCS-028)
Ryakam Achyuta Kumar(2018BCS-042)

TABLE OF CONTENTS :

1.Introduction

1.1 Overview of the Project

Academic Report Management as the name indicates will be used for managing student reports along with their personal information, their respective subjects and also professors' Information. Accepting correct data will result in generating the desired output in correct format without any delay. So this system has been enabled with validation data checkup while entering the data and before saving it to the particular file. Students will also able to check their data and modify their basic information as per their requirements. There will be only admin who will have to authority to make changes and modification while using this system. Admin will able to perform activities like, adding student records, modifying and deleting existing records, allowing authority to access particular section of student data by different departments. To perform admin work, users have to select admin panel and provide their id and password to access this section.

1.2 Objectives of the Project

The objectives of the project is to :-

- To maintain and update Student Reports
- Adding Student Records, Modifying and Deleting existing Records
- To display the Professors' Information and their respective Subjects
- To Display Students' personal Information like Name, admission number, stream, date of birth, contact number, etc.
- To provide an easy, interactive and secured interface by providing password to login for teachers.

1.3 Need for the Project

- As each institution has various departments which will use student records in order to carry out their task. Now, integrating task and sharing data among department can be easily achieved.
- It has been difficult for students to search for their respective reports. So this maintains students' database so it will be easy for students.
- It has been difficult for the management and the teachers to maintain and update student reports and teachers' Information.
- Sometimes students may modify their respective reports. So, this is a secured interface which only enables the teachers with their password to add or modify the records.

1.4. Overview of existing systems and technologies

Main technologies associated with Academic Report Management are

- OOPS concept (C++)
- File Handling (database)
- Lucid chart (Class diagram)

1.5 Scope of the project :

Main actors of this system :

- Teachers or Professors
- Students
- Others

Teachers :

- Add student records
- Modify student records
- Delete a student record
- Display all the records
- Search a respective student record by Admission number
- Search a student record by name
- Display sorted Student records by Percentage

Students:

- Search their respective report
- Display sorted Student records by Percentage.
- Display all reports.

Others :

- Display all the reports
- Display sorted Student records by Percentage.

2.Feasibility Study

2.1 Financial Feasibility

No cost will be charged for making Academic Report Management Project.

All you need to have is :

- C++ compiler
- Lucid Chart account to make a class diagram

Free version of Lucid chart is available which is more than enough for making a good class diagram.

2.2 Technical Feasibility

The main technologies that are associated with library management are

• C++ Language

• C++ compiler

• Class Diagram using Lucid Chart

Each of the technologies and programs are freely available and the technical skills required are easily manageable

So from these Its clearly understood that the project is technically feasible.

2.3 Resource and Time Feasibility

Resources that are needed for Academic Report Management are :

- Programming Device [Laptop]
- Programming Tools [Dev C++ and Lucid Chart are freely available online]
- Programming Individuals

So It is clear that the project is resource feasible.

2.4 Risk Feasibility:

Risk Feasibility can be discussed under several contexts.

Risk associated with size of program:

Same program can be reused for multiple records.So,the reusability of the program is high and size of the program is decreased.

Risk associated with runtime:

The tasks are clearly identified before the implementation phase.

Business Impact Risks:

- Effect of this management system on colleges:

Academic Report Management System can be implemented in college academic report or can be integrated into existing system for an easy , interactive and secured interface.

- This system supports many number of users simultaneously.
- Number of other systems with which this system must be interoperable.
  This system can be integrated with current university Fee validation system with slight modifications. Doing so will add a significant value to both systems.

Customer related risks:

Academic Report Management is a general type of system (not designed just for a single college). Before implementing the system in an educational institute, there will be some basic modifications required.

Development Related Risks:

Are tools for analysis and design available?

- C++(Programming Language)
- Lucid Chart(Class Diagram)

Are compilers or code generators available and appropriate for the system to be built?

- Dev C++ Compiler is available freely online.

Does the environment make use of a database or repository?

- This is a database oriented system that will use File Handling in C++

Technology risks:

Is the technology to be built new?

- All the technologies are very well established and old enough (but not obsolete).

Do the system requirements demand the creation of new algorithms, input or output technology?

- Academic Report System will have several algorithms to generate reports of the students and modify them.

2.5 Social/Legal Feasibility:-

- Academic Report Management uses freely available development tools, and provide the system as an open source system. Only the maintenance cost will be charged from potential customers.
- C++ libraries that are used in this system are free open source libraries.

Since this new system eliminates the effort to make statistical distributions, it will have a great impact in a university system.

3.Considerations:

Performance:

Academic Report Management performance will not be affected by the number of users

Dev C++Compiler is fast enough for better performance.

Response time: Yet to be tested

Processing time: Yet to be tested

Throughput: Yet to be tested

Query and Reporting time: Yet to be tested

Security:

Security measures are provided in many aspects in this system.

Teacher Authentication:

Teacher will have to authenticate using the password provided with.

Usability and Ease of Use:

The interfaces are designed to make it easy for any user to get familiar with the system.

Availability:

System will be available throughout the 24 hours. Mean time to failure and mean time to repair will be decided to increase the availability.

FUNCTIONAL REQUIREMENTS:

Classes Defined:

Student
Prof
Subject
Date
Admsnno
BCS
IMT
IMG

Class Student:

This class contains information about all the current existing students' personal information like admission number, name, father's name, mother's name, date of birth, contact number and their academic report with marks of the respective stream's subjects.
It contains functions to enter the information by the user, and to display the information.

+enter():
Takes input of all
+display():
Display Personal Information.
friend void add():
Add Student Records.
friend void dispall():
Display all Student Reports.
friend void del():
Delete Student Records
friend void modify():
Modify Student Records
friend void search():
Search students' reports by Roll Number.
friend void search_student():
Search students' reports by Name.
friend void sortper():
Sort students' reports by their percentage.

Class Prof:

This class contains information about all the current existing professors' personal information like Professor name and Professor ID.

Prof():

Contructor

Class Subject:

This class contains information about all the current existing Subjects for respective streams along with their professor details.

Subject():

Constructor

Class Date:

This class contains the format in which date must be entered.

Date():

Constructor

Class Admsno:

This Class contains the format in which admission number must be entered.

Admsno():

Constructor

Class bcs:

This Class is for students of BCS Stream.

entermarks:

Enter the marks of respective subjects.

displaymarks:

Display and Calculate total and percentage of all Subjects.

Class img:

This Class is for students of IMG Stream.

entermarks:

Enter the marks of respective subjects.

displaymarks:

Display and Calculate total and percentage of all Subjects.

Class imt:

This Class is for students of IMT Stream.

entermarks:

Enter the marks of respective subjects.

displaymarks:

Display and Calculate total and percentage of all Subjects.

Class Diagram: