

EXAMINATION RESULT AUTOMATION AND ANALYSIS

MINI PROJECT REPORT

submitted by

SANAL DAVIS(EPAJECS067)

SHAHEER V.M.(EPAJECS068)

SHARATH K.S.(EPAJECS069)

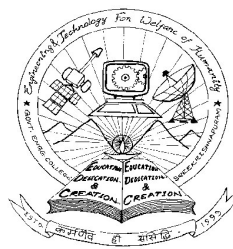
SHERLIN N.G.(EPAJECS055)

SIJO M.C.(EPAJECS070)

for the award of the degree

of

Bachelor of Technology



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GOVERNMENT ENGINEERING COLLEGE SREEKRISHNAPURAM

PALAKKAD-678633

JULY 2012

CERTIFICATE

This is to certify that the mini project report entitled **EXAMINATION RESULT AUTOMATION AND ANALYSIS** submitted by **SANAL DAVIS, SHAHEER V.M., SHARATH K.S., SHERLIN N.G.** and **SIJO M.C.** to the **Department Of Computer Science and Engineering, Government Engineering College, Sreekrishnapuram, Palakkad-678633**, in partial fulfilment of the requirement for the award of B.Tech Degree in Computer Science and Engineering is a bonafide record of the work carried out by them.

Prof. Reena Nair
Project Guide

Dr. P.C. Reghu Raj
Head of the Department

Ezudheen P
Project Co-ordinator

Place: Sreekrishnapuram

Date:

ACKNOWLEDGEMENT

First and foremost we wish to express our wholehearted indebtedness to God Almighty for his gracious constant care and blessings showered over us for the successful completion of the project.

The team of the project is deeply indebted to Dr. P.C Reghuraj , Head of Department of Computer Science and Engineering, Govt. Engineering College Sreekrishnapuram, for providing us all the required facilities for undertaking the project in a systematic way.

We also express our sincere gratitude to Mr Ezudheen P , Project co-ordinator, Department of Computer Science and Engineering, Govt. Engineering College Sreekrishnapuram, for his valuable co-ordination on our project

We are extremely grateful to Prof. Reena Nair, Department of Computer Science and Engineering, Govt. Engineering College Sreekrishnapuram, for their sincere guidance, inspiration throughout the project.

Gratitude is extended to all teaching and non teaching staffs of Department of Computer Science and Engineering, Govt. Engineering College Sreekrishnapuram for the sincere directions imparted and the cooperation in connection with the project.

We are also thankful to our parents for the support given in connection with the project. Gratitude is extended to all well-wishers and our friends who supported us to complete the project in time.

Table of Contents

List of Figures	v
Abstract	1
1 INTRODUCTION	2
2 PROBLEM DEFINITION	3
3 BACKGROUND STUDY	4
3.1 Literature Review	4
3.1.1 Result Automation	4
3.1.2 PDF Extraction	5
3.1.3 Result Analysis	6
4 DESIGN	7
4.1 Levels	7
4.1.1 User Level	7
4.1.2 Administrator Level	7
4.2 System Requirements	8
4.2.1 Software Requirements	8
4.3 Algorithm	9
4.3.1 Algorithm for automation	9
4.3.2 Algorithm for pdf extraction and storing	9
4.3.3 Algorithm for result analysis	10

5	IMPLEMENTATION	13
6	RESULT AND ANALYSIS	15
7	CONCLUSION AND FUTURE SCOPE	22
	Bibliography	23
	Appendix A	24

List of Figures

4.1	Block Diagram	11
4.2	Table 1: TABLES	11
4.3	Table 2: SEMBRYEAR	12
6.1	Starting Form	16
6.2	Main Entry Form	17
6.3	Register Number Form	18
6.4	Result Form	19
6.5	Table Form	20
6.6	Bar Chart Form	21

Abstract

We introduce a new project naming "EXAMINATION RESULT AUTOMATION AND ANALYSIS". In present scenario, each student queries the main university result database and will get the scoresheet in pdf format. If you want to get the entire result each student has to query themselves. Also, the analysis of entire result includes more manual effort, it is a tedious work. Thus the importance of the proposed system EXAMINATION RESULT AUTOMATION AND ANALYSIS. This system is mainly focuses on minimizing the manual effort. In other words, it aims on the automation of manual works. In this proposed system, the result of all the data will be retrieved from the university database and will be saved in a single file. The result can be analysed in different ways such as class-wise, subject-wise, and so on according to the users perspective.

Our main aim of this project is to reduce the manual work. We focus on the retrieval of results of students in a concise manner. The entire result of students can be retrieved by entering their first and last register number. Also, a bar chart is used to represent the percentage of success in each subject and analysis can be done based on it.

CHAPTER 1

INTRODUCTION

The work for retrieving the results of entire students of a batch is getting complex. Each student has to query themselves to get their result. The result will be shown in an individual pdf. Therefore, it is harder to analyze the results of all the students based on user's perspective. This analysis is a time consuming one since it is a manual work, and each results are in seperate pdf.

Our project aims to reduce this manual work. We focus on the retrieval of results of students in a concise manner. The entire result of students can be retrieved by entering their first and last register number. Also,a bar chart is used to represent the percentage of success in each subject and analysis can be done based on it. This analysis is a time saving one and increases the efficiency of analysing.

In other words, it aims on the automation of manual works. In this proposed system, the entire result will be retrieved from the university database and will be shown in a single file. The result can be analysed in different ways such as class-wise, subject-wise, and so on according to the users perspective.

CHAPTER 2

PROBLEM DEFINITION

Retrieving the results from university and its analysis is a tedious manual work. Results of students are in separate pdf and their analysis takes much more time. So our objective is to develop a system that can retrieve results within a single query and an efficient form of analysis of results.

CHAPTER 3

BACKGROUND STUDY

3.1 Literature Review

3.1.1 Result Automation

Result automation means automatic retrieval of the results from the university database. The aim is to automatically retrieve the exam result without human interaction. For this we can use features of python. Following are some of the features of python.

Simple

Python is a simple and minimalistic language. Reading a good Python program feels almost like reading English, although very strict English! This pseudo-code nature of Python is one of its greatest strengths. It allows you to concentrate on the solution to the problem rather than the language itself.

Free and Open Source

Python is an example of a FLOSS (Free/Libr and Open Source Software). In simple terms, you can freely distribute copies of this software, read its source code, make changes to it, use pieces of it in new free programs, and that you know you can do these things. FLOSS is based on the concept of a community which shares knowledge. This is one of the reasons why Python is so good - it has been created and is constantly improved by a community who just want to see a better Python.

High-level Language

When you write programs in Python, you never need to bother about the low-level details such as managing the memory used by your program, etc.

Portable

Due to its open-source nature, Python has been ported (i.e. changed to make it work on) to many platforms. All your Python programs can work on any of these platforms without requiring any changes at all if you are careful enough to avoid any system-dependent features.

3.1.2 PDF Extraction

The purpose of pdf extraction is to retrieve each and every data from the pdf. Pdf has certain restrictions on operations of data on it. So, data is extracted from the pdf and is stored in a text file because operations on the textfile has no restrictions. This portion is the important part of the system. Following shows the sample code for pdf extraction.

```
writer = PdfFileWriter()
content=""
remoteFile = urlopen(Request(url1)).read()
memoryFile = StringIO(remoteFile)
pdfFile = PdfFileReader(memoryFile)
content += pdfFile.getPage(0).extractText() + ""
content = " ".join(content.replace(u"xa0", " ").strip().split())
f= open('test2.txt','w')
pdf1 = content.encode("ascii", "ignore")
```

```
f.write(pdf)
```

```
f.close()
```

Data is retrieved from the pdf based on the url. This data is stored onto the text file for further operations. This data is then used to enter into the database and for analysing purposes.

3.1.3 Result Analysis

Result analysis is the process of analysing the results based on user's perspective. In this system, we use bar chart for analysing. Bar chart is a way of analysing the data based on a certain result. It shows the percentage of each portion of data in the result using bars. At the top of each bar, it shows the percentage level. In this system, bar chart allows us to analyze the results. We can also analyse the results using SQL queries. SQL queries are used for manipulating the contents of the database. We can insert, update and delete the contents of database. In this system, SQL queries are used to analyze the results on various aspects. The TKinter package is used for implementing the barchart. From this the rectangle boxes are used to construct bar in the barchart.

Tkinter is the tool used to draw chart. Tkinter is a GUI (graphical user interface) widget set for Python.

.grid() command is used to display various interface to the form.

- A cell is the area at the intersection of one row and one column.
- The width of each column is the width of the widest cell in that column.
- The height of each row is the height of the largest cell in that row.
- For widgets that do not fill the entire cell, you can specify what happens to the extra space. You can either leave the extra space outside the widget, or stretch the widget to fit it, in either the horizontal or vertical dimension.
- You can combine multiple cells into one larger area, a process called spanning.

CHAPTER 4

DESIGN

4.1 Levels

Result analysis can be carried out in two levels.

4.1.1 User Level

Analysing is done based on the users aspects. Bar chart is used to analyse the results. User enters the required information to the correct field and proceed to the next stage. User enters mainly three inputs- the url of the university database where results are stored, the starting and ending register number of the batch of students. After that,the desired output will be stored in the database for analysis.

If the user enters the correct data, a corresponding table is created. Then, each students result is loaded into the database. Using bar chart,these contents are analysed. Bar chart represents the percentage of the success of students in each subject. User analysis can be done in following ways. It shows the subject in which the most students got passed. Also,it shows the subject inwhich most students failed.

4.1.2 Administrator Level

Analysing is done based on the administrator level. SQL query is used for analyzing. Administrator are different from normal users because they have high previlages. They can change the structure of the database. Queries are normally used to manipulate the database such as insertion,deletion,update etc.In this level, queries are

used to analyse the results. Administrators can get the results directly from the table itself rather than by executing the program codes

4.2 System Requirements

4.2.1 Software Requirements

Operating System: Linux Programming Environment:

- Python Language
- MySQL
- TKinter

Platform

The coding is done entirely in LINUX platform. We choose Ubuntu because it is more user friendly and its better performance in real time environment. Moreover, choosing an open source platform give us access to the immense support from various forums in open source community.

Programming Language

The source code is written in Python and is executed as Python filename.py. Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

PyPdf

A Pure-Python library built as a PDF toolkit. It is capable of: extracting document information (title, author, ...), splitting documents page by page, merging

documents page by page, cropping pages, merging multiple pages into a single page, encrypting and decrypting PDF files. By being Pure-Python, it should run on any Python platform without any dependencies on external libraries. It can also work entirely on StringIO objects rather than file streams, allowing for PDF manipulation in memory. It is therefore a useful tool for websites that manage or manipulate PDFs.

4.3 Algorithm

4.3.1 Algorithm for automation

Automation means automatic retrieval of the results from the university database. It takes less time to execute than normal ones.

Step 1: Enter the URL.

Step 2: Open the requested url and read the pdf.

Step 3: Store the result into a memory file.

Step 4: Using the PdfFileReader,download the contents into a file

4.3.2 Algorithm for pdf extraction and storing

Pdf extraction means extracting the contents from pdf. Pdf extraction is important because operations on it have some restrictions. So,extracting the contents to text files makes it easy to manipulate.

Step 1: Read the required pdf.

Step 2: Extract the details such as name, register number,grade etc using `extractText()`.

Step 3: Open a file in write mode.

Step 4: Store the extracted content into the file

Step 5: Read the grade and SGPA of each student.

Step 6: Store the contents into a database.

Step 7: Close the file.

4.3.3 Algorithm for result analysis

Result analysis means analysing the results based on some criteria. Bar chart is also used for result analysis which shows the percentage of each portion of data.

Step 1: Read the content.

Step 2: Display the content in a table format

Step 3: Draw the bar chart that shows the percentage of each subject.

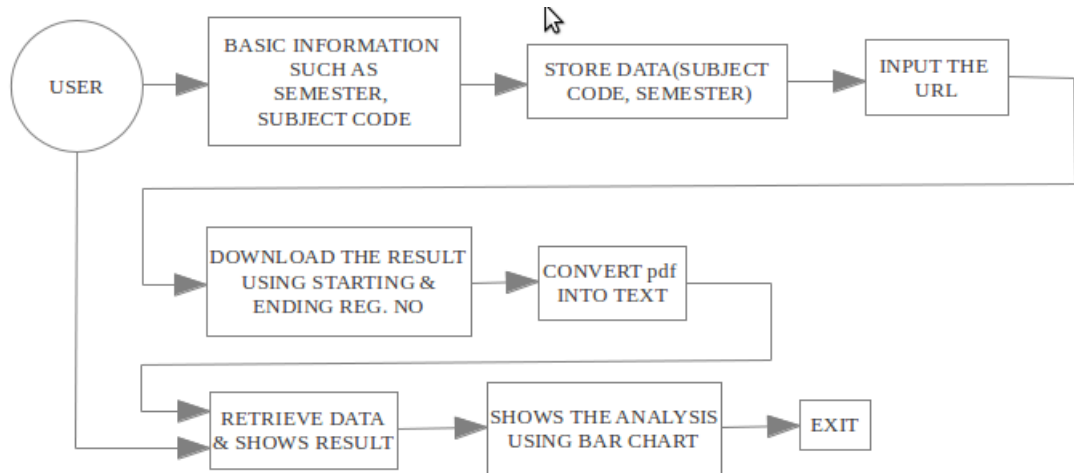


Figure 4.1: Block Diagram

FIELD_NAME	DATA_TYPE
NAME	VARCHAR(20)

Figure 4.2: Table 1: TABLES

FIELD_NAME	DATA_TYPE
REGNO	VARCHAR(20)
EN09_701	VARCHAR(20)
EN09_702	VARCHAR(20)
EN09_703	VARCHAR(20)
EN09_704	VARCHAR(20)
EN09_705	VARCHAR(20)
EN09_706	VARCHAR(20)
EN09_707	VARCHAR(20)
EN09_708	VARCHAR(20)
SGPA	VARCHAR(20)

Figure 4.3: Table 2: SEMBRYEAR

CHAPTER 5

IMPLEMENTATION

The project was implemented using python as the language and Mysql as the database. This project consists of four forms. The initial form consists of three buttons such as MAINENTRY, GETSTATUS and EXIT. If we use MAINENTRY button, a new form will be displayed.

This form is used to enter the basic details of a result of the student like subject code, year, semester etc. If wrong details are entered, corresponding message will be displayed. If correct, a table is created with proper attributes. Also, a new window will be displayed. This form consists of mainly three fields. User has to enter the url of the university database, the starting and ending register number of students of a batch. If submit is clicked, a form will be displayed. This form displays the result of all students in tabular form. All results are extracted from separate pdf files and the results are in a table format. If we click next button, a form will be displayed.

This form is used for analysis purposes. While clicking the submit button, certain amount of data is extracted from the university database. These data is entered into the database with correct attributes value. Using the contents, a bar chart is generated and displayed. The bar chart shows the percentage of the success of students in each subject. This bar chart can be analysed based on users aspects. From the bar chart, the subject with highest win percentage and lowest win percentage are highlighted with different colors.

If we use GETSTATUS button, a new form will be displayed. This form contains three fields such as year, semester and branch. If we click submit button, then the form will be displayed. This form displays the results of all students in a table

format. When we click the next button, a form will appear that contains the bar chart. This is a way in which bar chart can be accessed directly. This can be used for analysis purposes. The main form contains an EXIT button which is used to exit from the project.

CHAPTER 6

RESULT AND ANALYSIS

The system for the automatic retrieval and analysis of results was implemented. This system decreases the manual effort. This can be used to analyse the results based on user's aspect. The entire results of the students of a batch can be displayed in a single table. Also, the percentage of success of students in each subject is displayed using bar chart. This system is a time saving one and is an efficient one.

6.1 USER INTERFACES FOR RESULT ANALYSIS

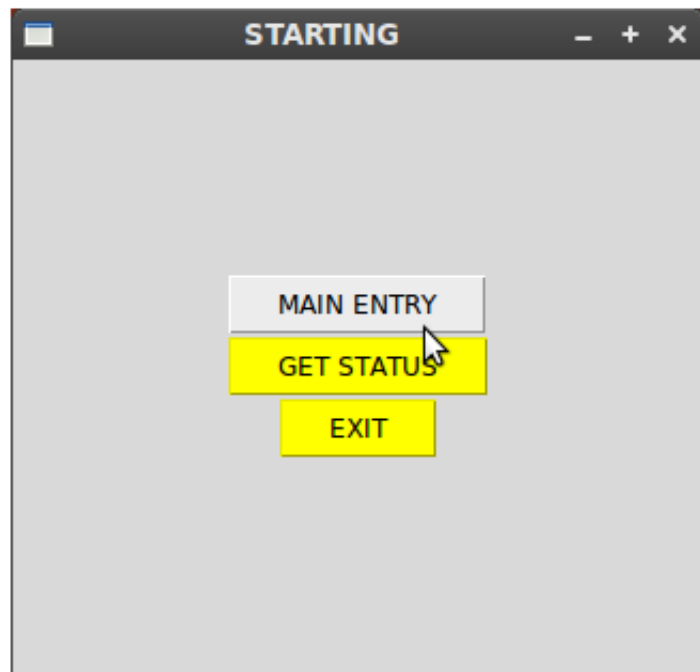


Figure 6.1: Starting Form

Sample application

Enter the branch	ec
Enter the year of admission	2009
Enter the month of examination	april
Enter the year of examination	2011
Enter the semester	s4
Enter the total number of subjects	8
Enter the code first subject	en09 401
Enter the code second subject	ec09 402
Enter the code third subject	ec09 403
Enter the code fourth subject	ec09 404
Enter the code fifth subject	ec09 405
Enter the code sixth subject	ec09 406
Enter the code seventh subject	ec09 407
Enter the code eight'th subject	ec09 408

NEXT

Figure 6.2: Main Entry Form

The image shows a web browser window with a dark title bar that reads "REGISTER NUMBER". The main content area has a light gray background. On the left, there is a text label "Enter the url(replace regno with %s)" followed by a long, empty white text input field. To the right of this field are two smaller, empty white text input fields. The first of these is labeled "Starting Regno" and the second is labeled "Ending Regno". Below the "Starting Regno" field is a "SUBMIT" button. A mouse cursor is visible over the "Starting Regno" field.

Figure 6.3: Register Number Form

GET RESULT

Enter the branch

cs

Enter the year of admission

2009

Enter the semester

s4

NEXT

Figure 6.4: Result Form

Sample application									
REGNO	EN09_401	EN09_402	CS09_403	CS09_404	CS09_405	CS09_406	CS09_407	CS09_408	SGPA
1	B	B	C	B	B	C	S	A	7.43
2	B	C	B	A	B	B	C	A	7.54
3	U	U	U	U	U	U	U	U	0*
4	C	B	C	B	C	D	A	C	6.75
6	D	B	B	C	A	C	C	A	7.07
7	U	D	C	U	B	U	U	A	3.43*
8	D	A	B	B	B	D	B	A	7.11
9	B	B	A	A	S	B	A	A	8.32
10	U	C	C	U	B	U	D	A	3.93*
11	C	A	S	S	S	B	A	S	8.88
12	U	D	U	U	C	U	U	D	1.91*
13	U	C	D	C	C	U	D	D	4.32*
14	D	D	B	U	B	C	B	B	5.79*
15	D	A	C	C	D	D	B	B	6.43
16	D	A	B	D	A	C	D	B	6.89
17	U	C	U	U	C	U	D	U	2.02*
18	U	A	D	C	U	U	D	B	3.75*
19	B	B	B	A	A	D	D	A	7.43
20	U	B	C	C	A	D	B	A	6.04*
21	U	U	U	U	U	U	U	D	0.39*
22	A	A	B	B	S	C	B	A	8.04
23	B	B	B	A	S	C	C	S	7.96
24	C	A	A	A	S	C	B	B	8
25	A	B	A	A	A	C	S	S	8.32
26	C	D	D	C	A	C	D	C	6.64

NEXT

Figure 6.5: Table Form



Figure 6.6: Bar Chart Form

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

This project automatically retrieves the results from university database by using a single query. Also, the entire result of the students can be shown in a single table. This project is a time saving one. It reduces the manual effort enormously. This project is very much helpful for the lectures for analysing the result on various aspects. It mainly aims on the automation of manual works. The result can be analysed in different ways such as class-wise, subject-wise, and so on according to the users perspective.

Bibliography

- [1] <http://www.daniweb.com/software-development/python/threads/167684/how-can-i-read-a-pdf-web-page>. [visited: March 2012]
- [2] <http://code.activestate.com/recipes/511465-pure-python-pdf-to-text-converter>. [visited: April 2012]
- [3] <http://code.activestate.com/recipes/577095-convert-pdf-to-plain-text/>. [visited: April 2012]
- [4] <http://stackoverflow.com/questions/6107195/python-pdf-to-text-convert>. [visited: April 2012]
- [5] <http://python4kids.wordpress.com/2011/07/18/tkinter-tinkering-graphical-user-interfaces/>. [visited: April 2012]
- [6] Copyright 2005 by Wiley Publishing, Inc., Indianapolis, Indiana: *Beginning Python*

Appendix A

Installation

- pyPdf
- MySQLdb
- tkinter

Installation of Python

- Install Python by clicking on the Python dmg we downloaded. After installation, a python2.7 command should now be available on the terminal.
- Download setuptools tar file here: <http://pypi.python.org/pypi/setuptools>. Untar it, cd to the untar-ed directory and type in
- With that a pip-2.7 command should be available

Installation of MYSQL

- MySQLdb is the python support bindings for MySQL. Not that the name would lead you to believe that.
- Its sourceforge page calls it <http://sourceforge.net/projects/mysql-python/> which makes more sense. you need setuptools, which you usually already have:
- `sudo aptitude install python-setuptools` You need MySQL-devel to compile, but its not called that, its called: `libmysql++-dev` on Ubuntu
- `sudo apt-get install libmysql++-dev` download MySQLdb itself from: <http://sourceforge.net/projects/mysql-python/> the version you download will be more recent
- `tar xzf MySQL-python-1.2.3.tar.gz`

- `cd MySQL-python-1.2.3`
- `python setup.py build`
- `sudo python setup.py install` if it worked then test: open a python shell (also change directories out of the compile dir you were just in to avoid confusing yourself with the just compiled version sitting in your path)
- `import MySQLdb`