**A MINI PROJECT REPORT**  
**ON**  
**STUDENT MANAGEMENT**

Submitted to Mumbai University  
In the partial fulfillment of the requirement for the award of the degree of  
**Bachelor of Engineering**

In  
**COMPUTER ENGINEERING**

By  
**Mrs. Khan Heena**  
**(16CO04)**

Under the guidance of  
**MR. AMER SYED**  
**Assistant Professor**



‑‑

**Department of Computer Engineering**  
**Anjuman-I-Islam Kalsekar Technical Campus**

**Affiliated to Mumbai University**

KHANDA GOAN, NEW PANVEL, NAVI MUMBAI, MAHARASHTRA

2016-2017

**Department of Computer Engineering**  
**Anjuman-I-Islam Kalsekar Technical Campus**

**Affiliated to Mumbai University**

KHANDA GOAN, NEW PANVEL, NAVI MUMBAI, MAHARASHTRA

2017-2018



**DECLARATION BY THE CANDIDATE**

the mini project report entitled **“STUDENT MANGEMENT”**, is a record of bonafide work carried out by me and the results embodied in this project have not been reproduced or copied from any source. The results of this project report have not been submitted to any other University or Institute for the award of any other Degree or Diploma.

**KHAN HEENA (16COO4)**

**Department of Computer Engineering**  
**Anjuman-I-Islam Kalsekar Technical Campus**

**Affiliated to Mumbai University**

KHANDA GOAN, NEW PANVEL, NAVI MUMBAI, MAHARASHTRA

2016-2017



**CERTIFICATE**

This is to certify that the project report entitled **“STUDENT MANGEMENT”**, submitted by **Mrs. <KHAN HEENA>** , bearing **Roll. No.: <16CO04>** in the partial fulfillment of the requirements for the award of the degree of **Bachelor of Computer Engineering** is a record of bonafide work carried out by him.

**Course Owner**

**(Assit. Prof. AMER SYED)**

**INDEX**

**CONTENTS**

CH**APTER 1: INTRODUCTION**

* 1. Introduction…………………………………………….1
  2. Objective………………………………..……< >
  3. Problem Statement……..……….....................< >

**CHAPTER 2: SYSTEM IMPLEMENTATION**

2.1 scheduling algorithm…………………………..< >

2.2 Code…………………..…..……….…….……< >

**CHAPTER 4: RESULTS**

4.1 Output…...……………………………..< >

**CHAPTER 5: CONCLUSION**

5.1 Conclusion….………………………………...< >

**REFERENCES**...………………………………………………< >

1.1

**Introduction**

For every colleges important task for administration department is to manage student information details in a procedure oriented manner with latest updates for every year which need to be available for easy access. In order to provide this service we designed a simple Students information management system project which has various modules which is helpful for administration to efficiently manage student’s details.

**PROBLEM STATEMENT**

One of the most important problems in operating systems designing is CPU Scheduling and challenge in this field is

to build a program to achieve proper scheduling. In case of priority scheduling algorithm when similar priority jobs

arrive than FCFS is used and the averagewaiting and turnaround time relatively higher. The process that arrives first is executed first, no matter how long it takes the CPU. So in this case if long burst time processes execute earlier

then other process will remain in waiting queue for a long time.

**OBJECTIVE**

The objective of this paper is:

 To Analysis of SJF priority based and FCFS priority based scheduling algorithm.

 To reduce the Average waiting time and Average Turnaround time of CPU.

**SCHEDULING ALGORITHM**

***1. First Come First Serve (FCFS)***



Process that requests the CPU first is allocated the CPU first

•Also called FIFO

*Non-preemptive*

•Used in batch systems

*Implementation*

• FIFO queues

• A new process enters the tail of the queue

• The scheduler selects next process to run from the head of the queue

***2.Shortest Remaining Proc. Time***

Algorithm

• Job with least remaining time to completion runs

• So, a new job that is shorter than remainder of running job preempts it

*Advantages*

• Similar to non-preemptive SJF

•Provably minimal average wait time

***.*** Moving shorter job before longer job improves waiting time of short

*Starvation again*

• A long job keeps getting preempted by shorter ones

• Example

**.**Process A with CPU time of 1 hour arrives at

time 0

***3.Round-robin***

One of the oldest, simplest, most commonly used scheduling algorithms

Select process/thread from ready queue in a round-robin fashion (i.e., take turns)

*Problems*

• Might want some jobs to have greater share

• Context switch overhead

**Code:**

import pymysql

connection=pymysql.connect('localhost','root','root','student)

cursor=connection.cursor()

def menu():

user1="admin"

pass1="admin"

user2=input("ENTER THE USERNAME:")

pass2=input("ENTER THE PASSWORD:")

if(user2 == user1 and pass2 == pass1):

print("-----------!!!!!!WELCOME-ACCESS GRANTED!!!!!!----------")

while(1):

c=int(input("""

1:ENTER THE STUDENT DETAILS

2:DELETE DETAIL

3:SEARCH DETAIL

4:EXIT

PLEASE ENTER YOUR CHOICE: """))

if(c is 1):

add\_detail()

elif(c is 2):

delete()

elif(c is 3):

search()

elif(c is 4):

return

else:

print("please enter valid number")

menu()

else:

print("Sorry can't access")

def add\_detail():

name=input("ENTER THE NAME: ")

gender=input("F OR M: ")

age=input("ENTER YOUR AGE: ")

clas=input("ENTER YOUR CLASS: ")

mob\_no=input("ENTER THE MOBILE NUMBER: ")

marks=input("ENTER YOUR MARKS: ")

email\_no=input("ENTER YOUR EMAIL: ")

sql = "INSERT INTO detail (NAME,GENDER,AGE,CLASS,MOB\_no,MARKS,EMAIL\_NO) VALUES (%s,%s,%s,%s,%s,%s,%s)"

cursor.execute(sql,(name, gender, age, clas, mob\_no, marks, email\_no))

connection.commit()

cursor.execute("select \* from detail")

data= cursor.fetchall()

print(data)

def search():

name\_s=input("ENTER THE NAME YOU WANT TO SEARCH : ")

cursor.execute("select \* from detail where NAME=%s",(name\_s))

data =cursor.fetchone()

if(data == None):

print('Sorry, Name not found')

else:

print("Id :",data[0])

print("Name : ",data[1])

print("Gender :",data[2])

print("Class :",data[3])

print("Age :",data[4])

print("Mob\_no :",data[5])

print("Marks :",data[6])

print("Email :",data[7])

def delete():

name\_d=input("ENTER THE NAME YOU WANT TO Delete : ")

cursor.execute("delete from detail where NAME=%s",(name\_d))

connection.commit()

print('Sucessfully deleted')

menu()

**Output:**

ENTER THE USERNAME:admin

ENTER THE PASSWORD:admin

-----------!!!!!!WELCOME-ACCESS GRANTED!!!!!!----------

1:ENTER THE STUDENT DETAILS

2:DELETE DETAIL

3:SEARCH DETAIL

4:EXIT

PLEASE ENTER YOUR CHOICE: 1

ENTER THE NAME: ilaf

F OR M: f

ENTER YOUR AGE: 20

ENTER YOUR CLASS: 2

ENTER THE MOBILE NUMBER: 65241389

ENTER YOUR MARKS: 70

ENTER YOUR EMAIL: ilaf@gmail.com

((1, 'muskan', 'f', '2', '19', '123346', '90', None), (2, 'heena', 'f', '2', '19', '123346', '50', None), (3, 'mariyam', 'f', '2', '19', '5876', '20', 'hgk'), (4, 'ilaf', 'f', '2', '20', '65241389', '70', 'ilaf@gmail.com'))

1:ENTER THE STUDENT DETAILS

2:DELETE DETAIL

3:SEARCH DETAIL

4:EXIT

PLEASE ENTER YOUR CHOICE: 2

ENTER THE NAME YOU WANT TO Delete : muskan

Sucessfully deleted

1:ENTER THE STUDENT DETAILS

2:DELETE DETAIL

3:SEARCH DETAIL

4:EXIT

PLEASE ENTER YOUR CHOICE: 3

ENTER THE NAME YOU WANT TO SEARCH : muskan

Sorry, Name not found

1:ENTER THE STUDENT DETAILS

2:DELETE DETAIL

3:SEARCH DETAIL

4:EXIT

PLEASE ENTER YOUR CHOICE: 2

ENTER THE NAME YOU WANT TO Delete : heena

Sucessfully deleted

1:ENTER THE STUDENT DETAILS

2:DELETE DETAIL

3:SEARCH DETAIL

4:EXIT

PLEASE ENTER YOUR CHOICE: 3

ENTER THE NAME YOU WANT TO SEARCH : mariyam

Id : 3

Name : mariyam

Gender : f

Class : 2

Age : 19

Mob\_no : 5876

Marks : 20

Email : hgk

1:ENTER THE STUDENT DETAILS

2:DELETE DETAIL

3:SEARCH DETAIL

4:EXIT

PLEASE ENTER YOUR CHOICE: 3

ENTER THE NAME YOU WANT TO SEARCH : ilaf

Id : 4

Name : ilaf

Gender : f

Class : 2

Age : 20

Mob\_no : 65241389

Marks : 70

Email : ilaf@gmail.com

1:ENTER THE STUDENT DETAILS

2:DELETE DETAIL

3:SEARCH DETAIL

4:EXIT

PLEASE ENTER YOUR CHOICE: 1

ENTER THE NAME: muskan

F OR M: f

ENTER YOUR AGE: 19

ENTER YOUR CLASS: 2

ENTER THE MOBILE NUMBER: 9876532

ENTER YOUR MARKS: 80

ENTER YOUR EMAIL: muskan@gmail.com

((3, 'mariyam', 'f', '2', '19', '5876', '20', 'hgk'), (4, 'ilaf', 'f', '2', '20', '65241389', '70', 'ilaf@gmail.com'), (5, 'muskan', 'f', '2', '19', '9876532', '80', 'muskan@gmail.com'))

1:ENTER THE STUDENT DETAILS

2:DELETE DETAIL

3:SEARCH DETAIL

4:EXIT

PLEASE ENTER YOUR CHOICE: 4

mariyam@mariyam:~/project$ python3 pro.py

ENTER THE USERNAME:mariyam

ENTER THE PASSWORD:mariyam

Sorry can't access

**CONCLUSION**

**REFERENCES**

[1]. Milenkovic,M., Operating System Concepts and Design,

McGraw Hill, International Edition, 1992.

[2].Silberschatz,A. and P.B. Galvin,1997. Operating System concepts,Fifth Edition, John Wiley & Sons, Inc.,