STUDENT FEES MANAGEMENT

- Srijana Dey

WHAT IS A STUDENT FEES MANAGEMENT SYSTEM?

A fees management system is a software solution that makes the process of fee collection,segregration,and categorization simple and automated for easy integration with the integrated school /college platform.

TABLE OF CONTENTS

01

ABOUT THE PROJECT

Abstract of the Project.

02

MOTIVATION OF WORK

Inspiration of making the system.

03

SYSTEM REQUIREMENTS

Requirements of the System for the outcome of this project.

TABLE OF CONTENTS

04

SNEAK PEEK

Description of the Code.

05

OUTPUT

Results.

06

PROPOSED WORK

A proposal of the system capabilities.

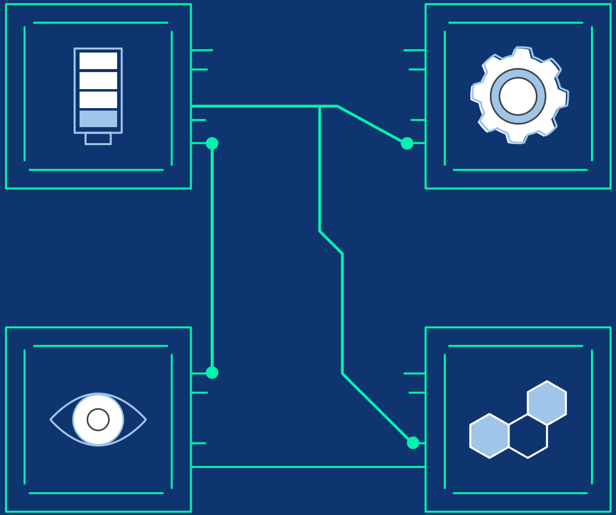
ABOUT THE PROJECT

ABSTRACT

Objective of the Project.

MODULES

Parts of the System.



KEYWORDS

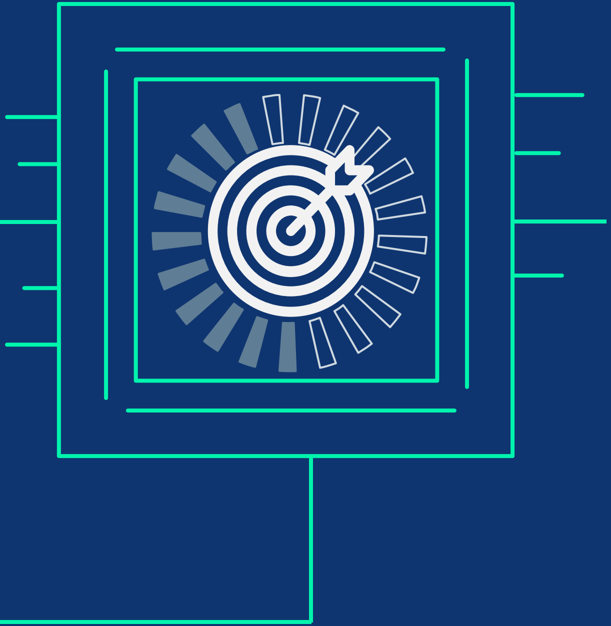
Subjects related to the Project.

PRE-REQUISITES

Languages.

ABSTRACT

This project Student Fees Management System has been developed on Python,Tkinter and MySQL and its a Python Web Application. The main aim of this project is to develop a GUI (Graphical User Interface) based software i.e. platform independent & user friendly and which can be fit into any college system.



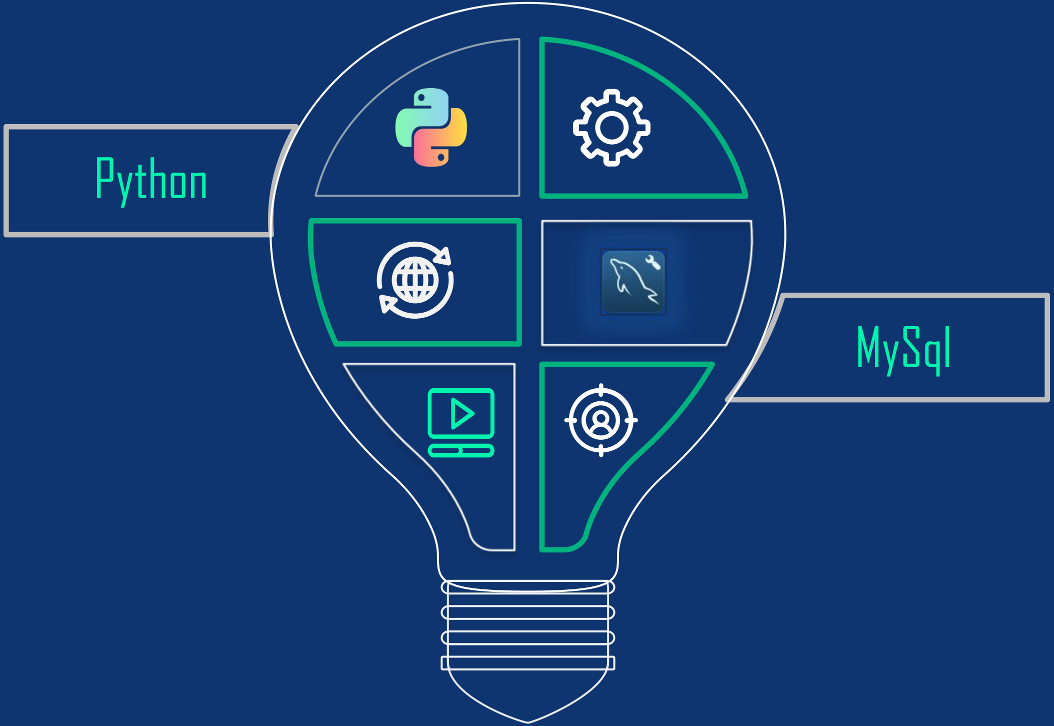
OBJECTIVE

- To reduce paperwork
- To make storage of information more efficient and secure.
- To have a friendly interface.
- To operate it easily and with minimum experience.

MODULES

1. Registration and Authentication
2. Administrator
3. Fee Payments
4. Reports

PRE-REQUISITES



Python

MySql

KEYWORDS

- Fee Management
- Python project
- MySQL
- Tkinter
- Fee Details
- Course Scheme
- Admin Login
- Student Details
- Dues
- Payment

SCOPE OF THE SYSTEM

- Specifically designed for a individual college/school.
- Inserting new student records are possible.
- It is based on desktop application.
- Not including fees other than academic like bus fees and etc.
- There are only limited number of modules for fees management.

MOTIVATION OF WORK

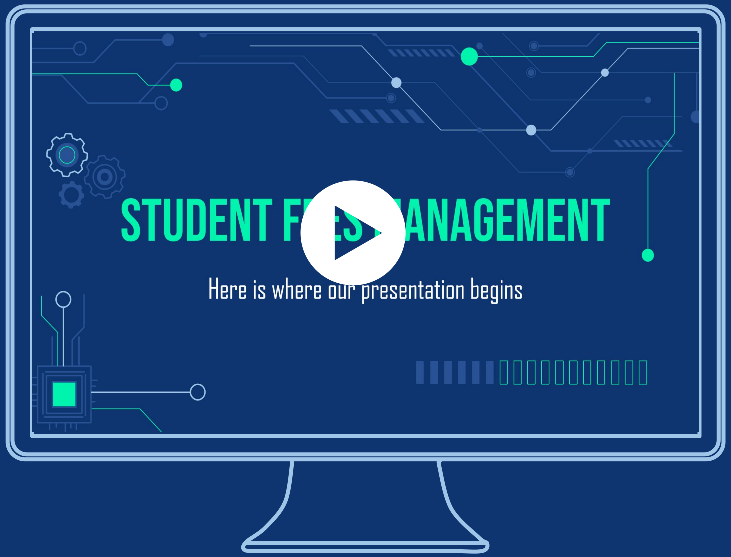
- In the existing system, colleges have to maually maintain information regarding to Fees deposited by the students.
- It is not properly capable to manage the student records with their fee details at asingle place.
- Managing collection of student fees, issuing fee receipts and fee register updation is a laborious manual process, leading to data inaccuracy and / or reconciliation.
- Preparing receipts manually everyday needs additional clerical staff.
- To generate due fees report is required a complete manual procedure, which involvesa lot of time and clerical staff manpower.
- Re-entry of fees receipts in accounting software separately leads to double manpower,cost and time.

SYSTEM REQUIREMENTS

- OS: Windows 7 with SP1; Recommended: Windows 10
- CPU: Intel or AMD processor with 64-bit support; Recommended: 2.8 GHz or faster processor
- GPU: nVidia GeForce GTX 1050 or equivalent; Recommended: nVidia GeForce GTX 1660 or Quadro T1000
- Disk Storage: 4 GB of free disk space
- Monitor Resolution: 1280x800; Recommended: 1920x1080

```
import mysql.connector as mysql

mydb = mysql.connect(host = "localhost", user = "rootsd", passwd = "1234",
database = "stfeeman")
mycursor = mydb.cursor(buffered=True)
mth = ('JANUARY', 'FEBRUARY', 'MARCH', 'APRIL', 'MAY', 'JUNE', 'JULY',
'AUGUST', 'SEPTEMBER', 'OCTOBER', 'NOVEMBER', 'DECEMBER')

def main():
    while 1:
        print("\n\n--------------------------------------")
        print("|          :  WELCOME  :          |")
        print("--------------------------------------")
        print("\n          1. LOGIN\n          2. SIGNUP\n          3. EXIT")
        choice = input("\nOption: ")

        if choice == "1":
            login()
        elif choice == "2":
            signup()
        elif choice == "3":
            exit(0)
        else:
            print("\nInvalid option!")
```

SNEAK PEEK

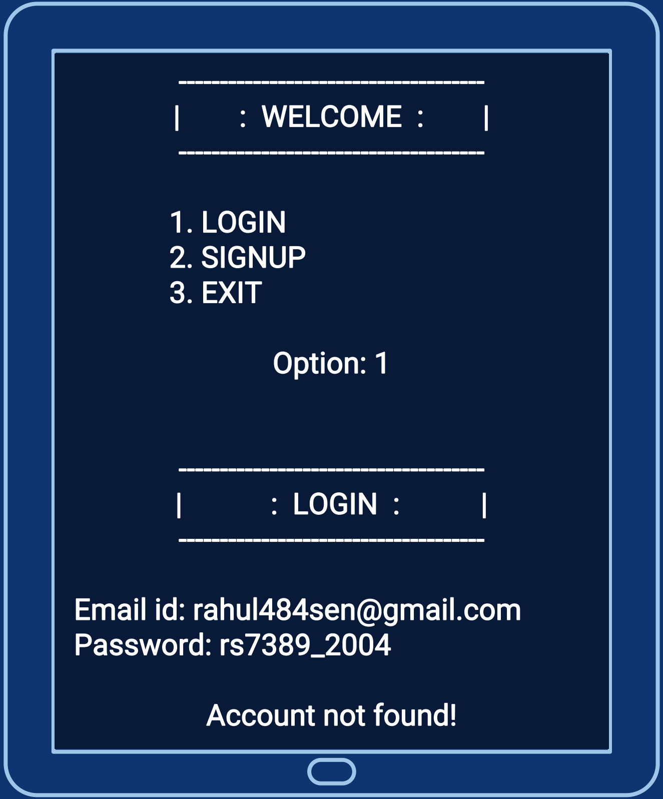


```
def login():
    print("\n\n------------------------------------")
    print("|          :  LOGIN  :           |")
    print("------------------------------------")
    email = input("\nEmail id: ")
    password = input("Password: ")
    mycursor.execute("SELECT * FROM student_db")
    data = mycursor.fetchall()

    if len(data) == 0:
       print("\nAccount not found!")
       main()
    else:
        for row in data:
            if row[4] == email:
                if row[6] == password:
                    print("\nLogged in successfully!")
                    name, adno = row[1], row[5]
                    student(name, adno)
                else:
                    print("\nInvalid email or password!")
                    login()
            else:
                if row[0] == len(data):
                    print("\nAccount not found!")
                    main()
```

```
def signup():
    print("\n\n------------------------------------")
    print("|          :  SIGNUP  :          |")
    print("------------------------------------")
    name = input("\nFullname: ")
    dob = input("D.O.B.(DD/MM/YYYY): ")
    contacts = input("Phone no.: ")
    email = input("Email id: ")
    adno = input("Admission no.: ")
    password = input("Create a strong password: ")

    get_val = (name, dob, contacts, email, adno, password)
    mycursor.execute("INSERT INTO student_db (name, dob, ph,
email_id, adm_no, password) VALUES (%s, %s, %s, %s, %s, %s)",
get_val)
    mydb.commit()
    print("\nAccount created successfully!")
    main()
```

```
def student(name, adno):
    print("\n\n------------------------------------")
    print("|       :  PAYMENT STATUS  :        |")
    print("------------------------------------")
    month = (input("\n\nEnter the month for which you want to pay the fees: 
")).upper()
    while month not in mth:
        print("\n\nError! Month not found.")
        month = (input("\nEnter the month for which you want to pay the 
fees: ")).upper()
    pay_check(name, adno, month)

def pay_check(name, adno, month):

    mycursor.execute("SELECT * FROM payment_db WHERE adm_no = 
%s AND month = %s", (adno, month))
    data = mycursor.fetchall()

    if len(data) == 0:
        print("\n\nPayment status: Not available")
        while 1:
            print("\n\nDo you want to:\n1. Proceed to payment\n2. Logout")
            ch = input("\nOption: ")
            if ch == "1":
                print("\n\nPROCEEDING. . .")
                print("\n\n------------------------------------")
                print("|          :  PAYMENT  :          |")
                print("------------------------------------")
                amt = input("\n\nEnter the amount to be paid: Rs. ")
                #Payment procedure
                print("\nPayment procedure. . .\n\n\nPayment successful!")
                print("\n\n--------------RECEIPT---------------")
                print(f"\n\n      * * * {month} FEES * * *")
                print("\n        Name:", name)
                print("        Admission no.:", adno)
                print("        Amount paid: Rs.", amt)
                get_val = (name, adno, amt, month)
                mycursor.execute("INSERT INTO payment_db (name, adm_no, 
amt, pay_stat, month) VALUES (%s, %s, %s, 'Paid', %s)", get_val)
                mydb.commit()
                main()
            elif ch == "2":
                main()
            else:
                print("Invalid option!")
    else:
        print("\n\nPayment status: Available")
        print(f"\n\n      * * * {month} FEES * * *")
        print("\n        Name:", name)
        print("        Admission no.:", adno)
        print("        Amount paid: Rs.", data[0][3])
        main()

main()
```

OUTPUT

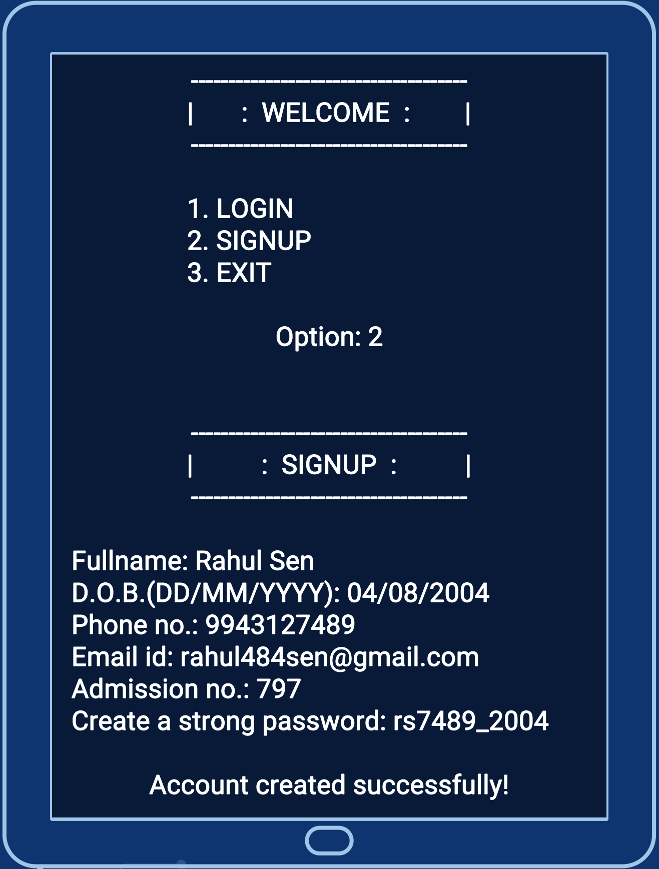


```
   -----------------------------------------
   |     :  WELCOME  :        |
   -----------------------------------------

   1. LOGIN
   2. SIGNUP
   3. EXIT

               Option: 1


   -----------------------------------------
   |           :  LOGIN  :           |
   -----------------------------------------

Email id: rahul484sen@gmail.com
Password: rs7389_2004

          Account not found!
```

DATABASE

Sl.	Name	DOB	Ph	Email_id	Adm_no	**Password**
1	Arjun Karmakar	08/09/2003	1234567809	arjun89@gmail.com	275	**arjun_k8903**
2	Piyali Dutta	21/09/2002	8752358805	dutta216piyali@gmail.com	184	**PiyaliD21**

OUTPUT

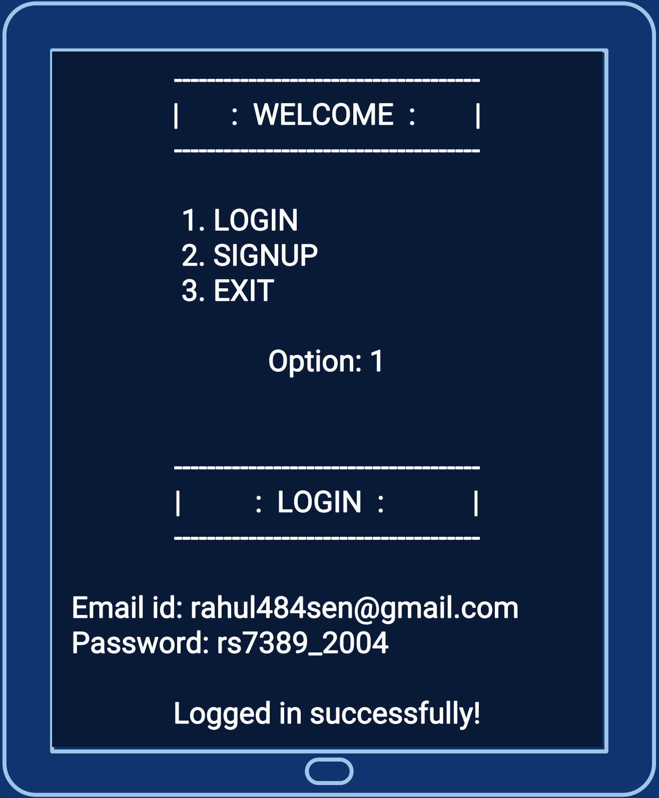


```
----------------------------------------
|      :  WELCOME  :        |
----------------------------------------

1. LOGIN
2. SIGNUP
3. EXIT

            Option: 2


----------------------------------------
|        :  SIGNUP  :         |
----------------------------------------

Fullname: Rahul Sen
D.O.B.(DD/MM/YYYY): 04/08/2004
Phone no.: 9943127489
Email id: rahul484sen@gmail.com
Admission no.: 797
Create a strong password: rs7489_2004

         Account created successfully!
```

DATABASE

Sl.	Name	DOB	Ph	Email_id	Adm_no	Password
1	Arjun Karmakar	08/09/2003	1234567809	arjun89@gmail.com	275	arjun_k8903
2	Piyali Dutta	21/09/2002	8752358805	dutta216piyali@gmail.com	184	PiyaliD21
3	Rahul Sen	04/08/2004	9943127489	rahul484sen@gmail.com	797	rs7489_2004

OUTPUT

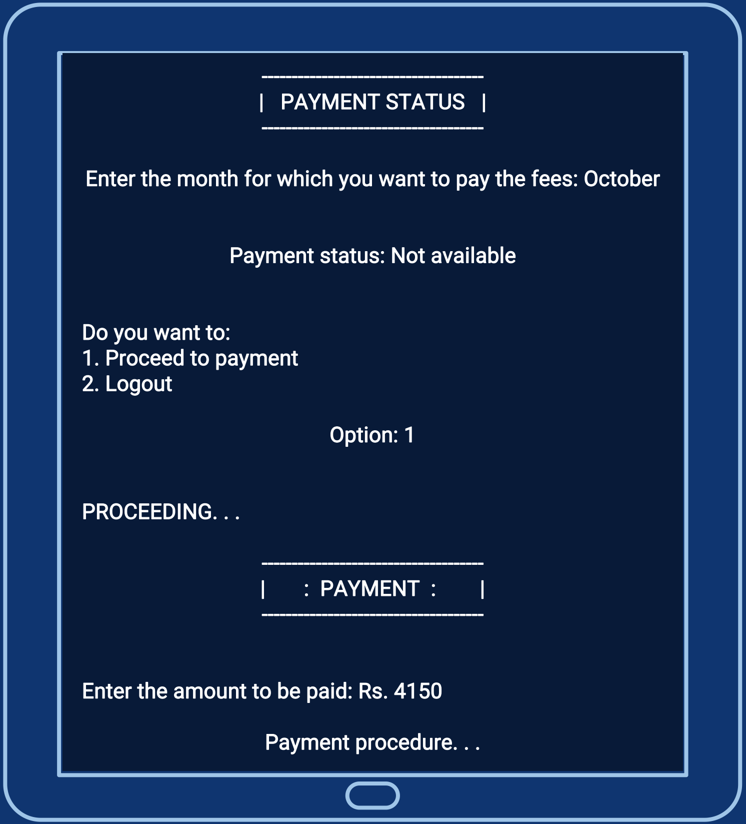


```
-----------------------------------
|     : WELCOME :        |
-----------------------------------

  1. LOGIN
  2. SIGNUP
  3. EXIT

          Option: 1


-----------------------------------
|       : LOGIN :          |
-----------------------------------

Email id: rahul484sen@gmail.com
Password: rs7389_2004

          Logged in successfully!
```

DATABASE

Sl.	Name	DOB	Ph	Email_id	Adm_no	Password
1	Arjun Karmakar	08/09/2003	1234567809	arjun89@gmail.com	275	arjun_k8903
2	Piyali Dutta	21/09/2002	8752358805	dutta216piyali@gmail.com	184	PiyaliD21
3	Rahul Sen	04/09/2004	9943127489	rahul484sen@gmail.com	797	rs7489_2004

OUTPUT

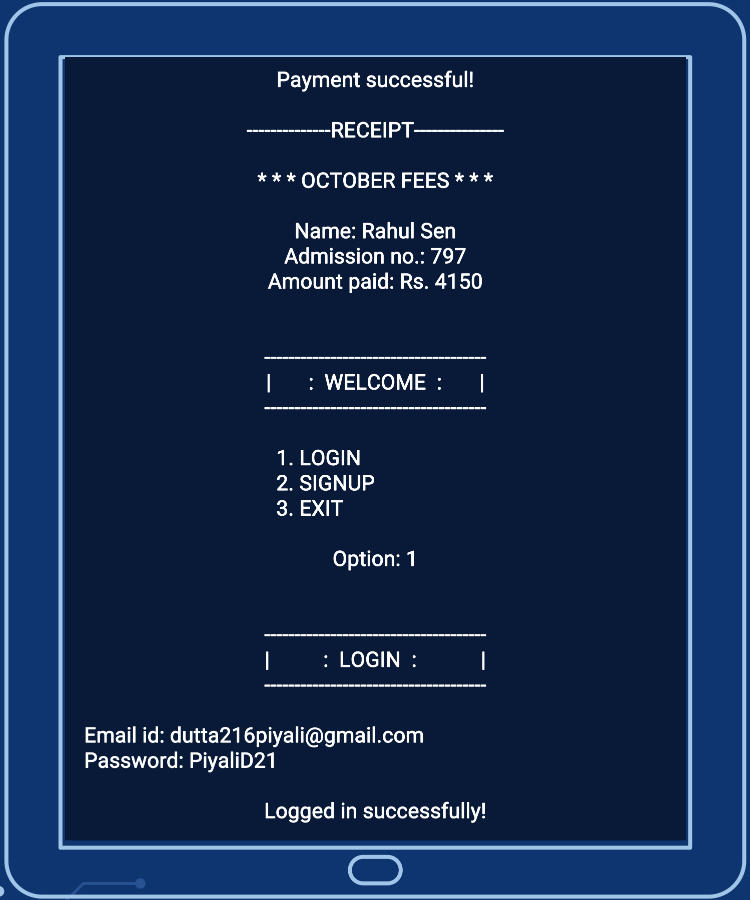


```
              ----------------------------------------
              |  PAYMENT STATUS  |
              ----------------------------------------

  Enter the month for which you want to pay the fees: October


                   Payment status: Not available


 Do you want to:
 1. Proceed to payment
 2. Logout

                          Option: 1


 PROCEEDING. . .


              ----------------------------------------
              |      :  PAYMENT  :       |
              ----------------------------------------


 Enter the amount to be paid: Rs. 4150

                   Payment procedure. . .
```

DATABASE

Sl.	Name	Adm_no	Amt	Pay_stat	Month
1	Arjun Karmakar	275	4300	Paid	OCTOBER
2	Piyali Dutta	184	4500	Paid	OCTOBER

OUTPUT

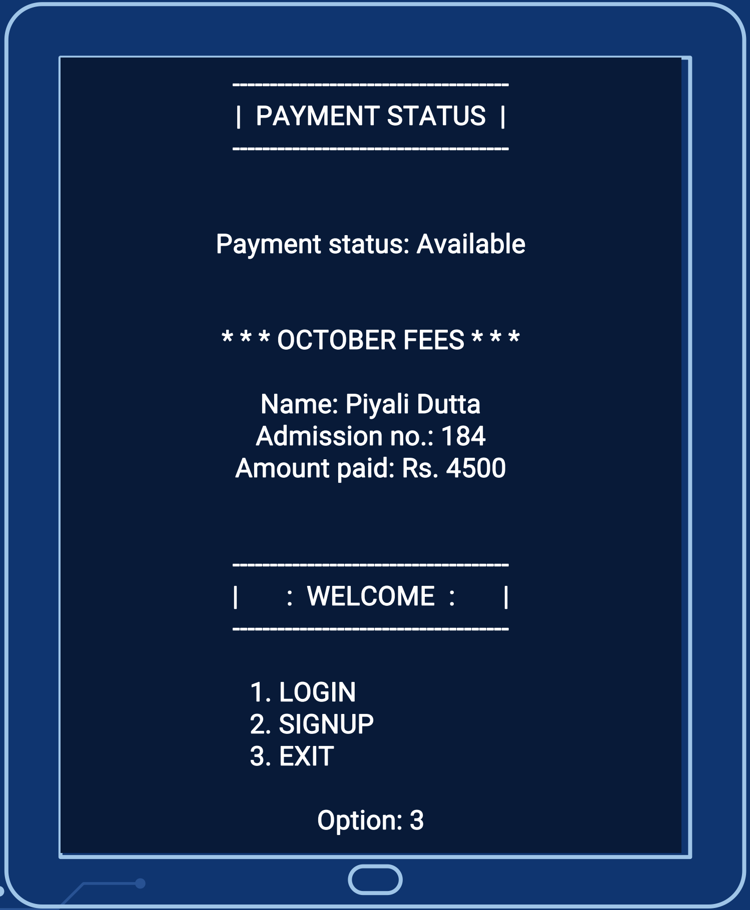


```
                    Payment successful!

         -------------RECEIPT--------------

              * * * OCTOBER FEES * * *

                  Name: Rahul Sen
                Admission no.: 797
              Amount paid: Rs. 4150


            ------------------------------------
            |      :  WELCOME  :         |
            ------------------------------------


              1. LOGIN
              2. SIGNUP
              3. EXIT

                      Option: 1


            ------------------------------------
            |        :  LOGIN  :          |
            ------------------------------------

Email id: dutta216piyali@gmail.com
Password: PiyaliD21

                 Logged in successfully!
```

DATABASE

Sl.	Name	Adm_no	Amt	Pay_stat	Month
1	Arjun Karmakar	275	4300	Paid	OCTOBER
2	Piyali Dutta	184	4500	Paid	OCTOBER
3	Rahul Sen	797	4150	Paid	OCTOBER

OUTPUT



```
-----------------------------------------
|  PAYMENT STATUS  |
-----------------------------------------


Payment status: Available


* * * OCTOBER FEES * * *

Name: Piyali Dutta
Admission no.: 184
Amount paid: Rs. 4500


-----------------------------------------
|      :  WELCOME  :      |
-----------------------------------------

   1. LOGIN
   2. SIGNUP
   3. EXIT

        Option: 3
```

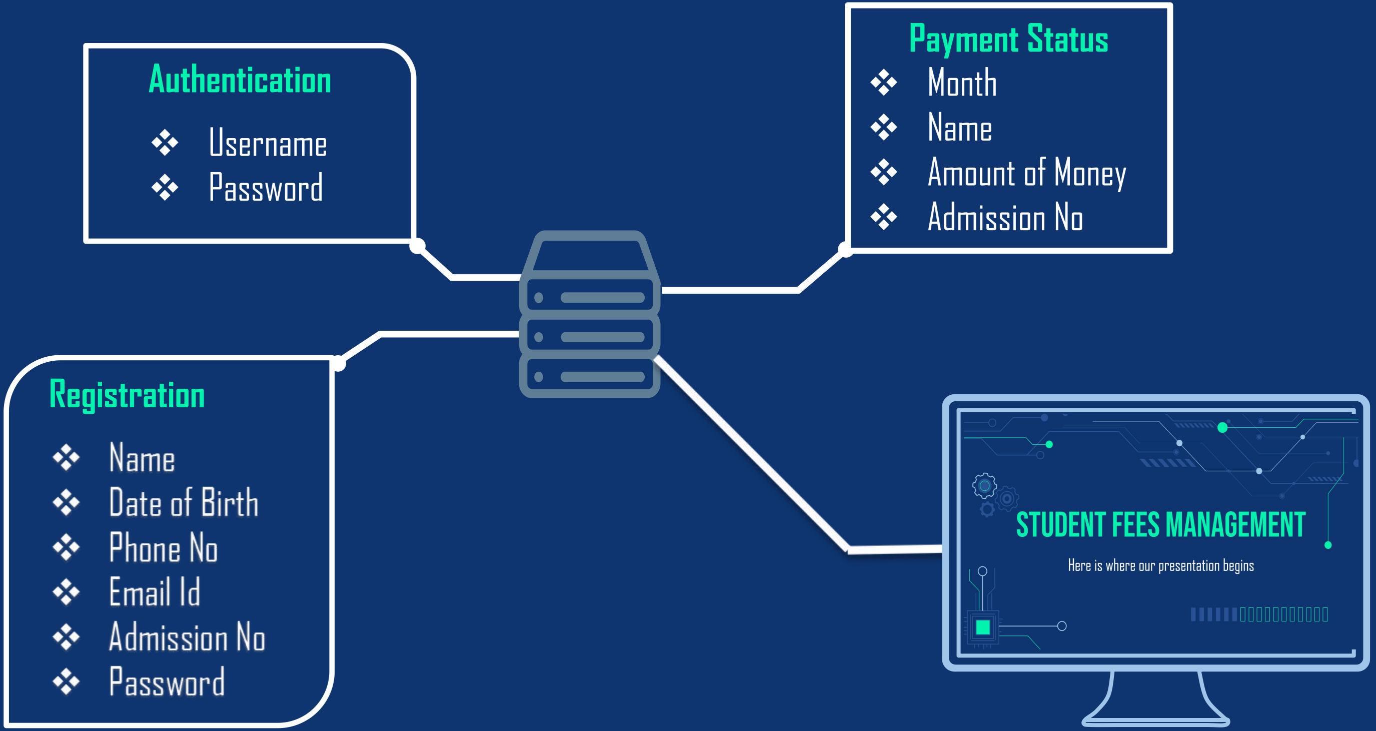
DATABASE

Sl.	Name	Adm_no	Amt	Pay_stat	Month
1	Arjun Karmakar	275	4300	Paid	OCTOBER
2	Piyali Dutta	184	4500	Paid	OCTOBER
3	Rahul Sen	797	4150	Paid	OCTOBER

PROPOSED WORK

- ❑ The fees management System is a desktop system aimed to maintaining students records and their fees details.
- ❑ It also generate records like I.e. Feed Paid, dues, and etc.
- ❑ The system requires small amount of time to generate reports needed to manage the fees of the student.
- ❑ Managing collection of school fees, issuing fee receipts and fee register updation is done with the help of software resulting in highly accurate data.
- ❑ Software provides facility to print receipts automizing office work.
- ❑ This Software provides facility to generate due fees report easily and at any point of time.
- ❑ Software directly enter fees receipts to the accounts of the school.

ZERO FLOW DIAGRAM



Authentication

- ❖ Username
- ❖ Password

Payment Status

- ❖ Month
- ❖ Name
- ❖ Amount of Money
- ❖ Admission No

Registration

- ❖ Name
- ❖ Date of Birth
- ❖ Phone No
- ❖ Email Id
- ❖ Admission No
- ❖ Password

STUDENT FEES MANAGEMENT

Here is where our presentation begins

THANKS!