



COLLEGE MANAGEMENT SYSTEM



Presenting by

1.N.Sasirekha
2.T.Sravanti

Guided by

Mrs.Pavithra

INTRODUCTION:

- The College management system is an automated version of manual Student Management System.
- It can handle all details about a student.
- In case of manual system they need a lot of time, manpower etc. Here almost all work is computerized. So the accuracy is maintained. Maintaining backup is very easy. It can do with in a few minutes.
- Our system has two type of accessing modes, administrator and user. Student management system is managed by an administrator. It is the job of the administrator to insert update and monitor the whole process.

DRAWBACKS OF MANUAL COLLEGE MANAGEMENT SYSTEM

- This process is so time-consuming.
- There is a threat to the record of the student and teachers, in this case, there might be a chance that a person makes entry on someone else record.
- There is no proper way of getting to know about the events and extra curriculum activities happening in colleges.
- It might be the case that student tries to bribe the teachers to avoid the long queue.

APPLICATION OF AUTOMATED COLLEGE MANAGEMENT SYSTEM

- A student can register himself directly on the college application for any notification of the event and for depositing fee and students can also post their queries and check their fee status.
- If the students are not aware of policies, then the admin will make him contact with this application and help them to complete the whole process.
- Admin will maintain the record and atomicity by using the proper table to save records so that no one can access it and data remain safe.
- Students need not bother about his/her fee deposit and without moving from their place or making so many phone calls they can make the fee payment.
- No need of worrying about status you can check that from your place

MODULES

- Admin Module
- Professor Module
- Student Module
- Course Module
- Payment Module

**“
SOFTWARE REQUIREMENTS
”**

BACKEND:

- Spring Core, Spring Boot, Rest API, Java Persistence API and Hibernate
- Postman

DATABASE:

- MySql

HARDWARE REQUIREMENTS

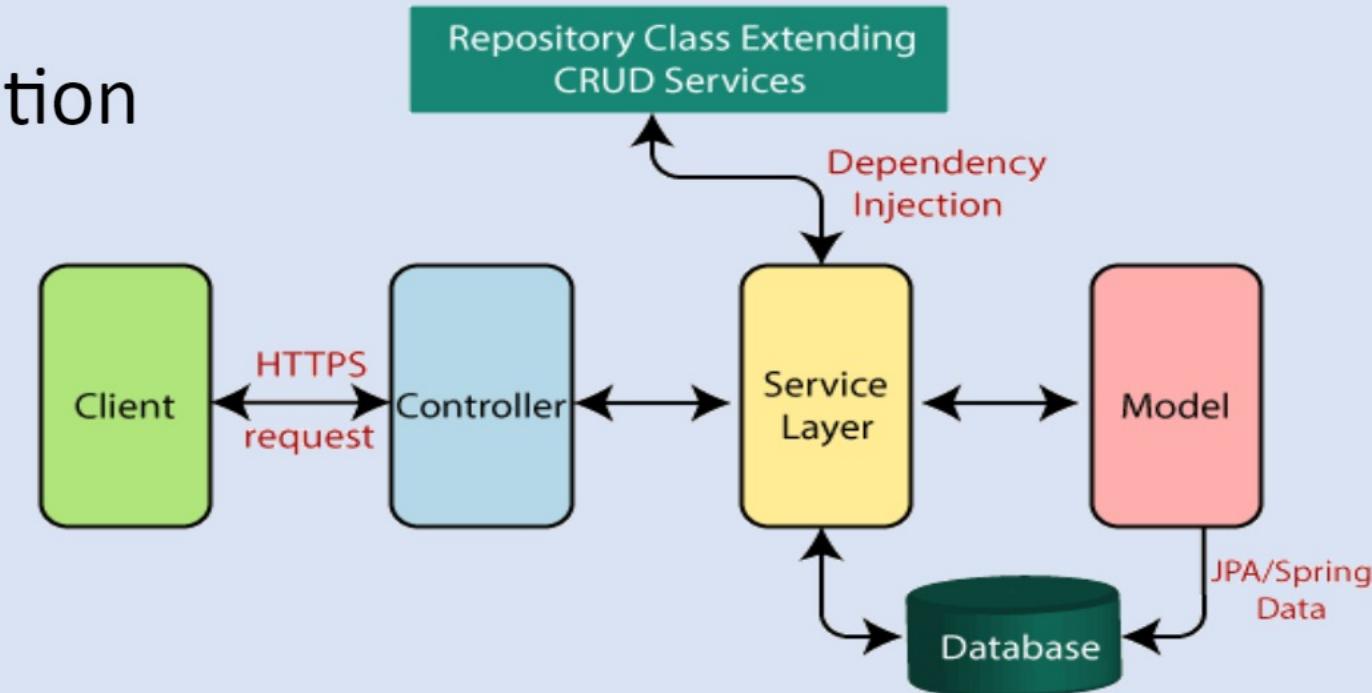
- Windows 10 or above.
- RAM 4GB or above.

WHAT IS SPRING BOOT?

❖ Java Spring Boot(Spring Boot) is a tool that makes developing web application and microservices with Spring Framework faster and easier through three core capabilities

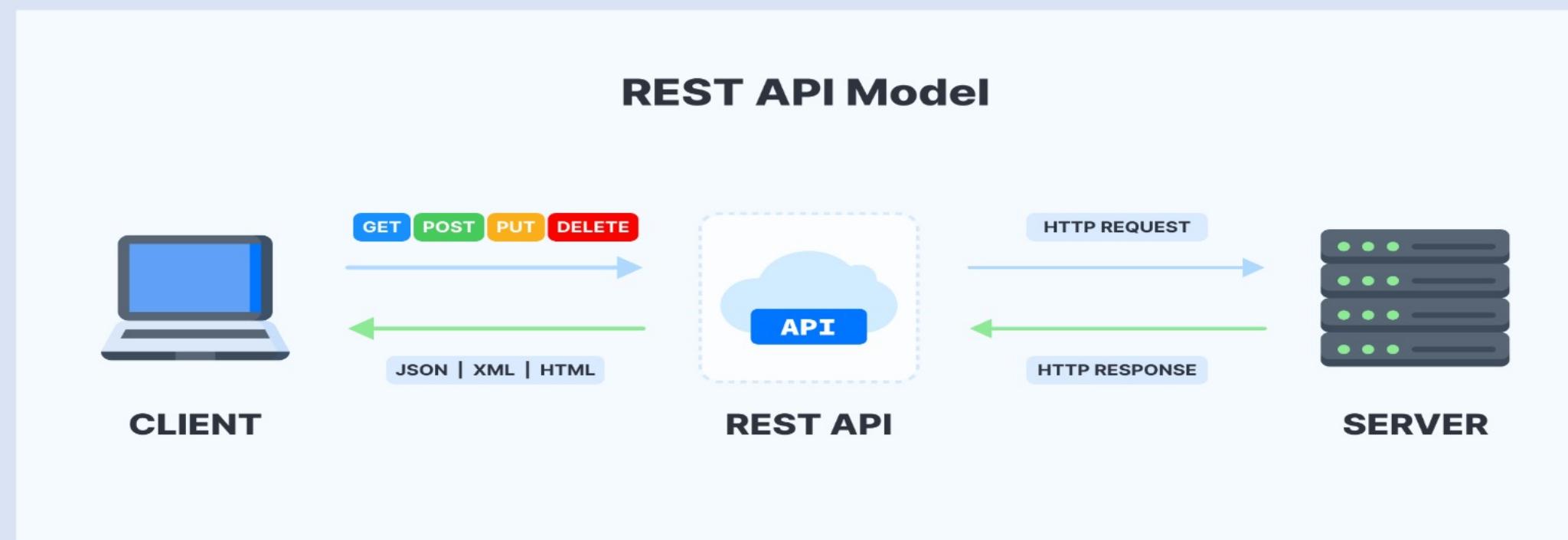
1. Auto Configuration
2. An Opinionated approach to Configuration
3. The ability to create standalone applications

Spring Boot flow architecture



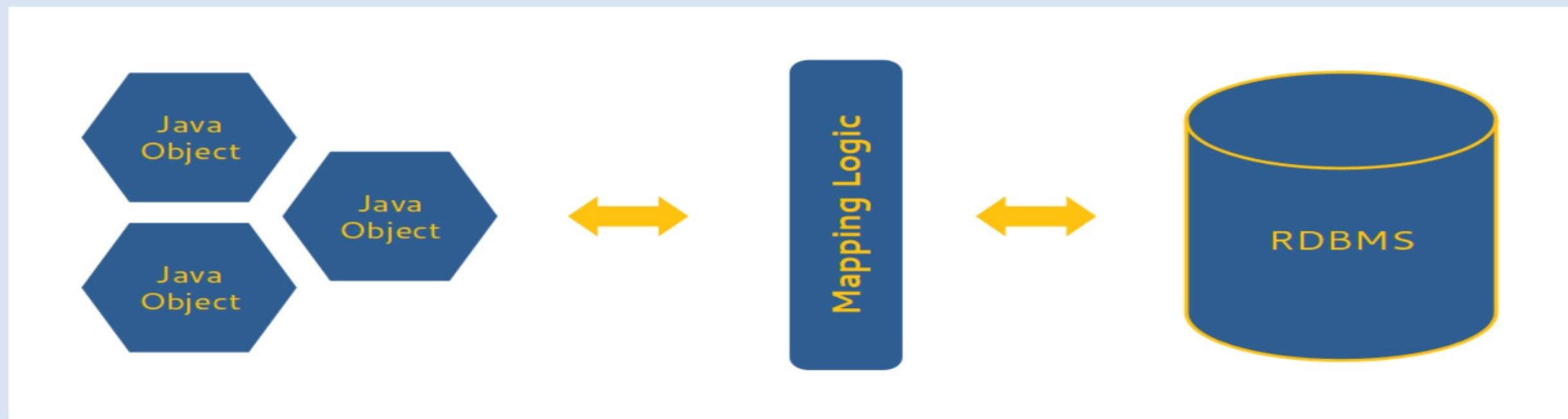
What is Rest API?

- Representational State Transfer (REST) is an architectural style that defines a set of constraints to be used for creating web services.
- REST API is a way of accessing web services in a simple and flexible way without having any processing.



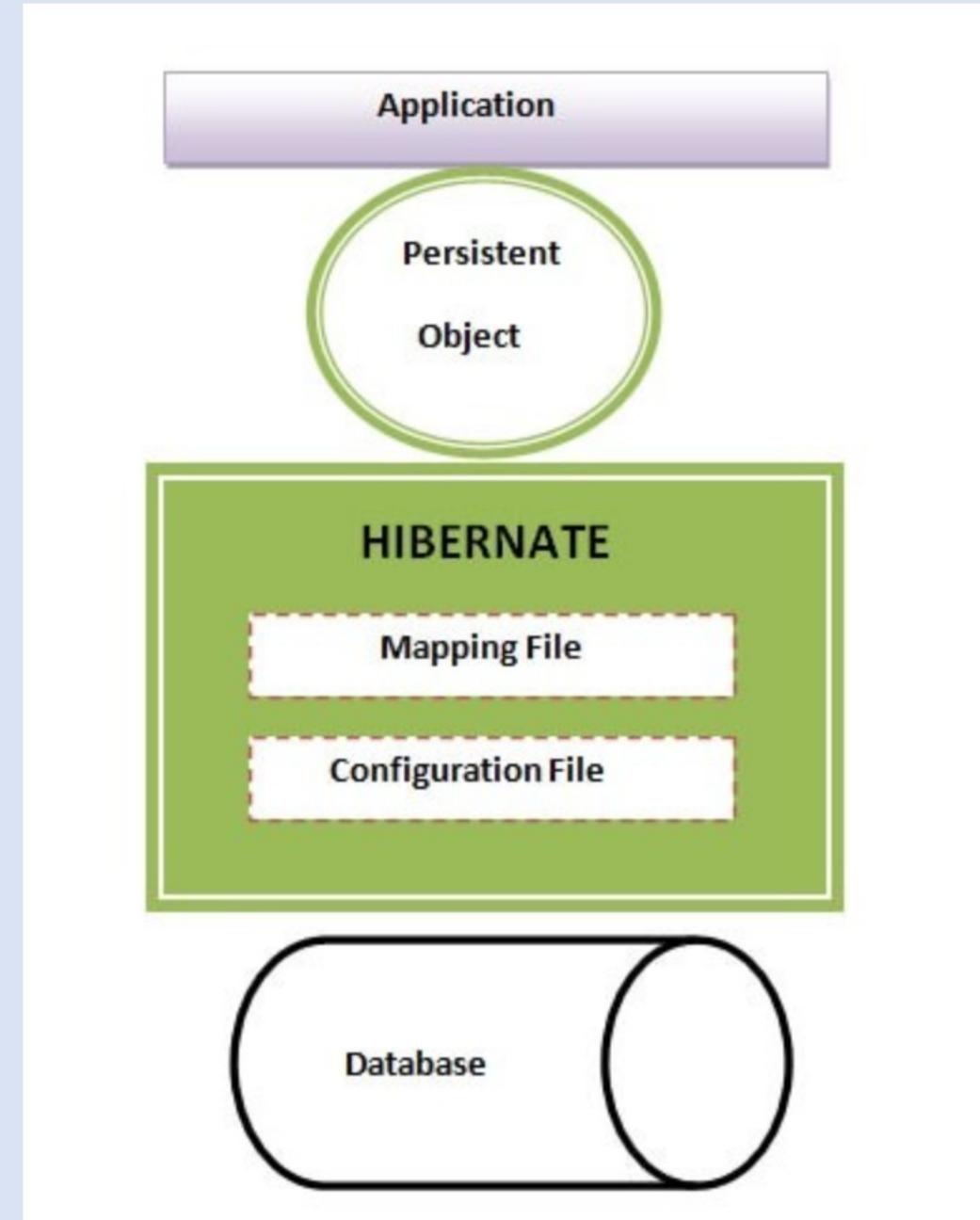
Java Persistence API ?

- A JPA (Java Persistence API) is a specification of Java which is used to access, manage, and persist data between Java object and relational database.
- It is considered as a standard approach for Object Relational Mapping.
- JPA can be seen as a bridge between object-oriented domain models and relational database systems.



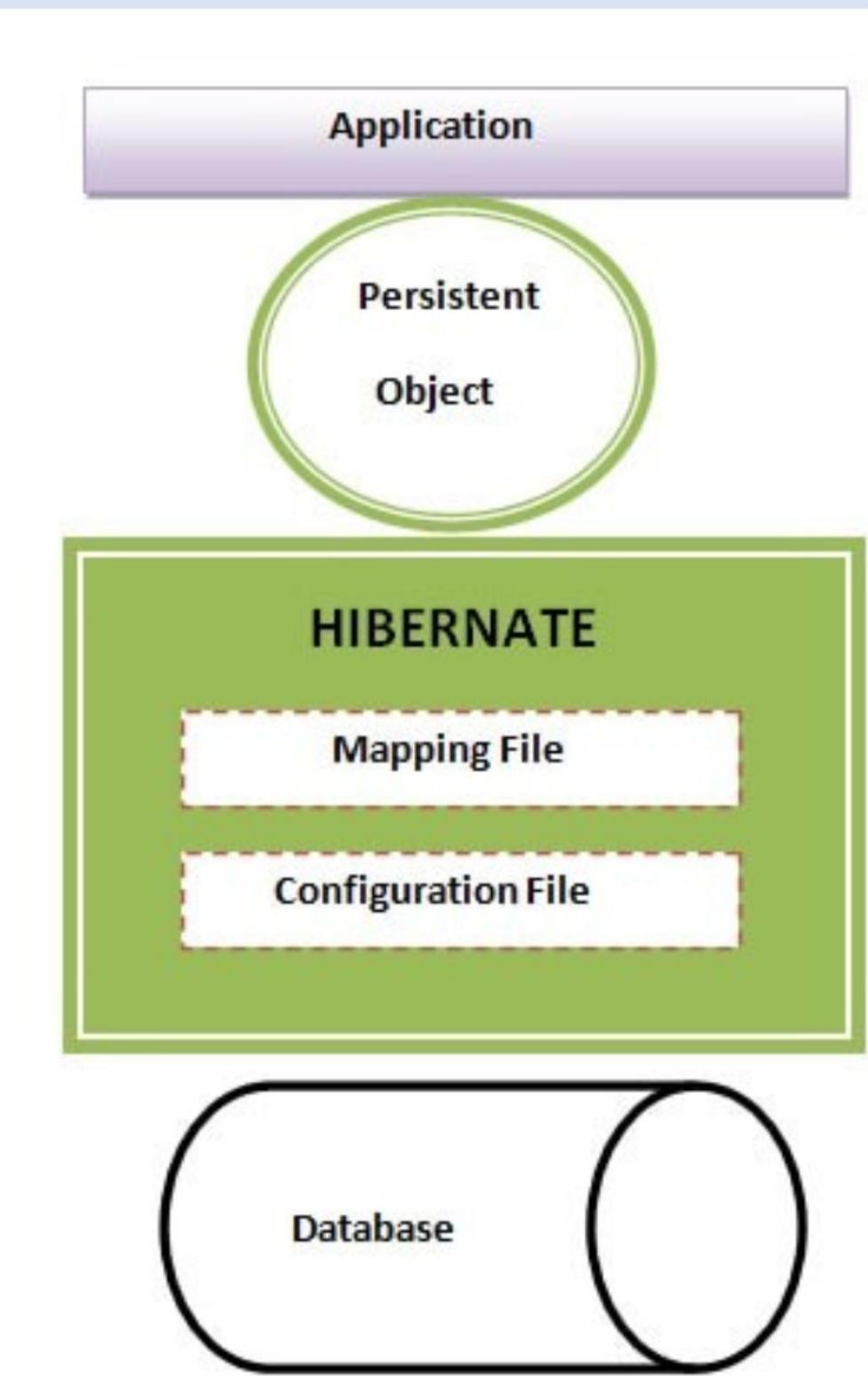
Hibernate ?

- Hibernate is an Object-Relational Mapping (ORM) tool which is used to save the state of Java object into the database.
- It is one of the most frequently used JPA implementation.
- It is defined in **org.hibernate** package.

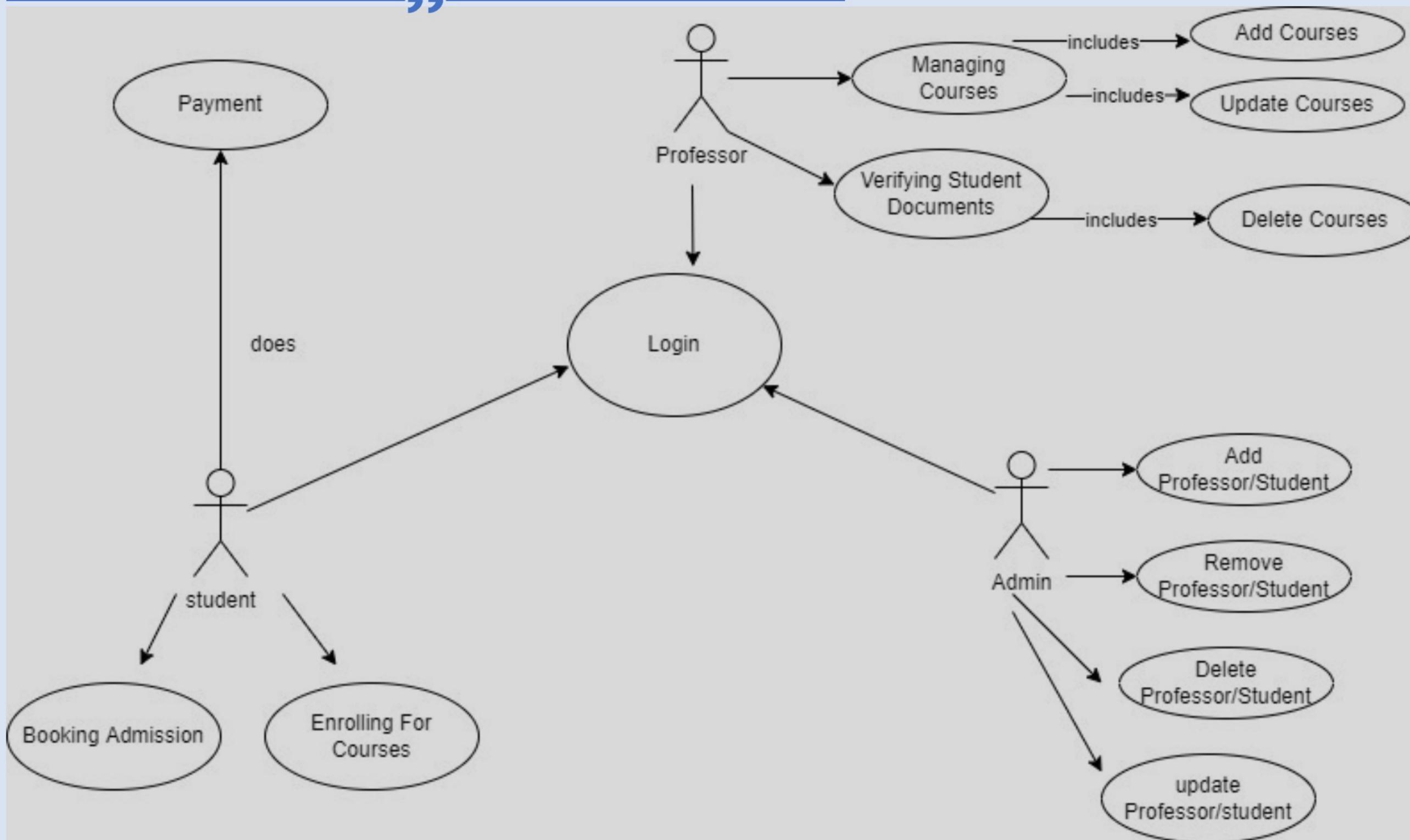


“ POSTMAN? ”

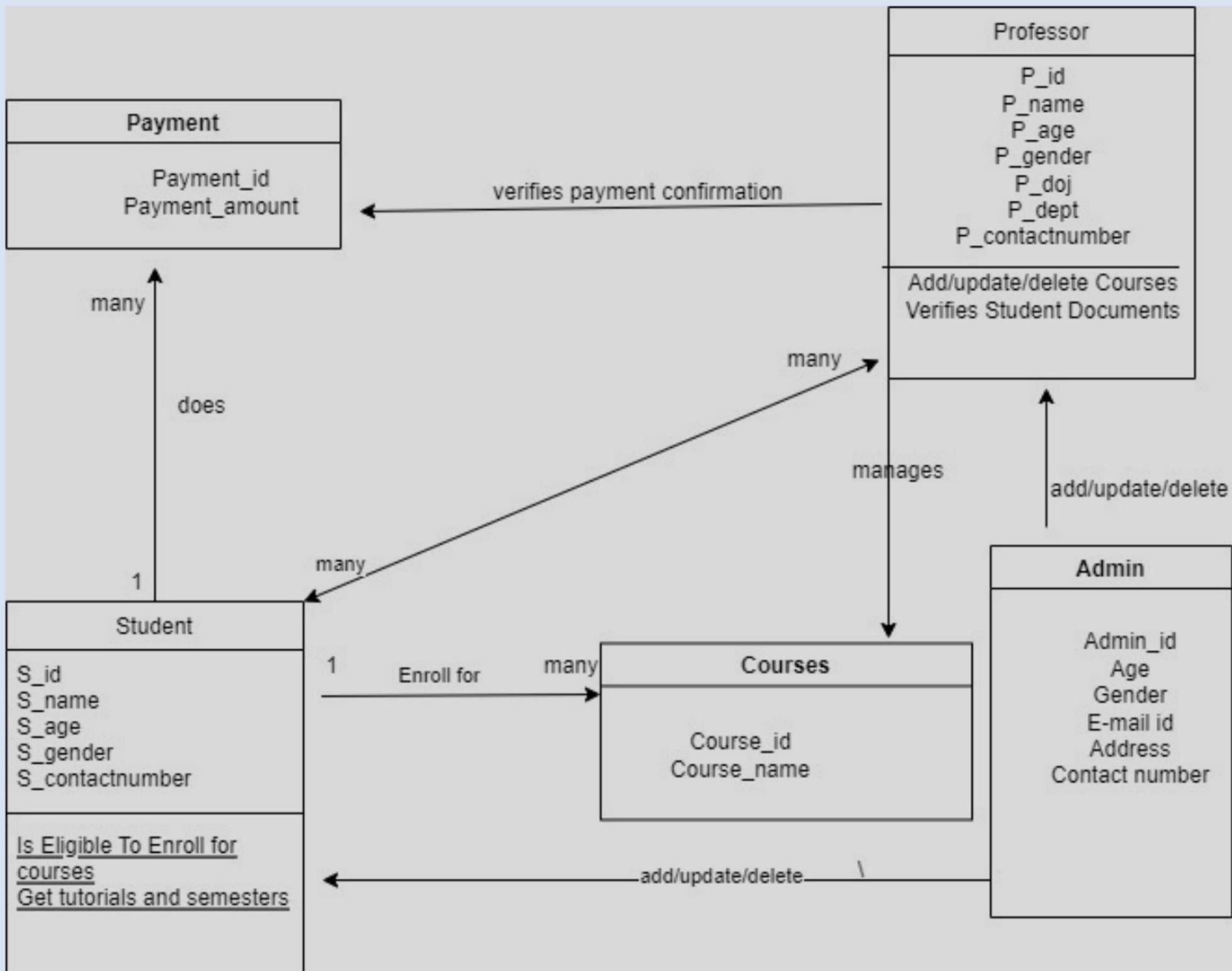
- Postman is an application used for API testing.
- It is an HTTP client that tests HTTP requests, utilizing a graphical user interface, through which we obtain different types of responses that need to be subsequently validated.
- Postman is an application that allows us to test APIs utilizing a graphical user interface.



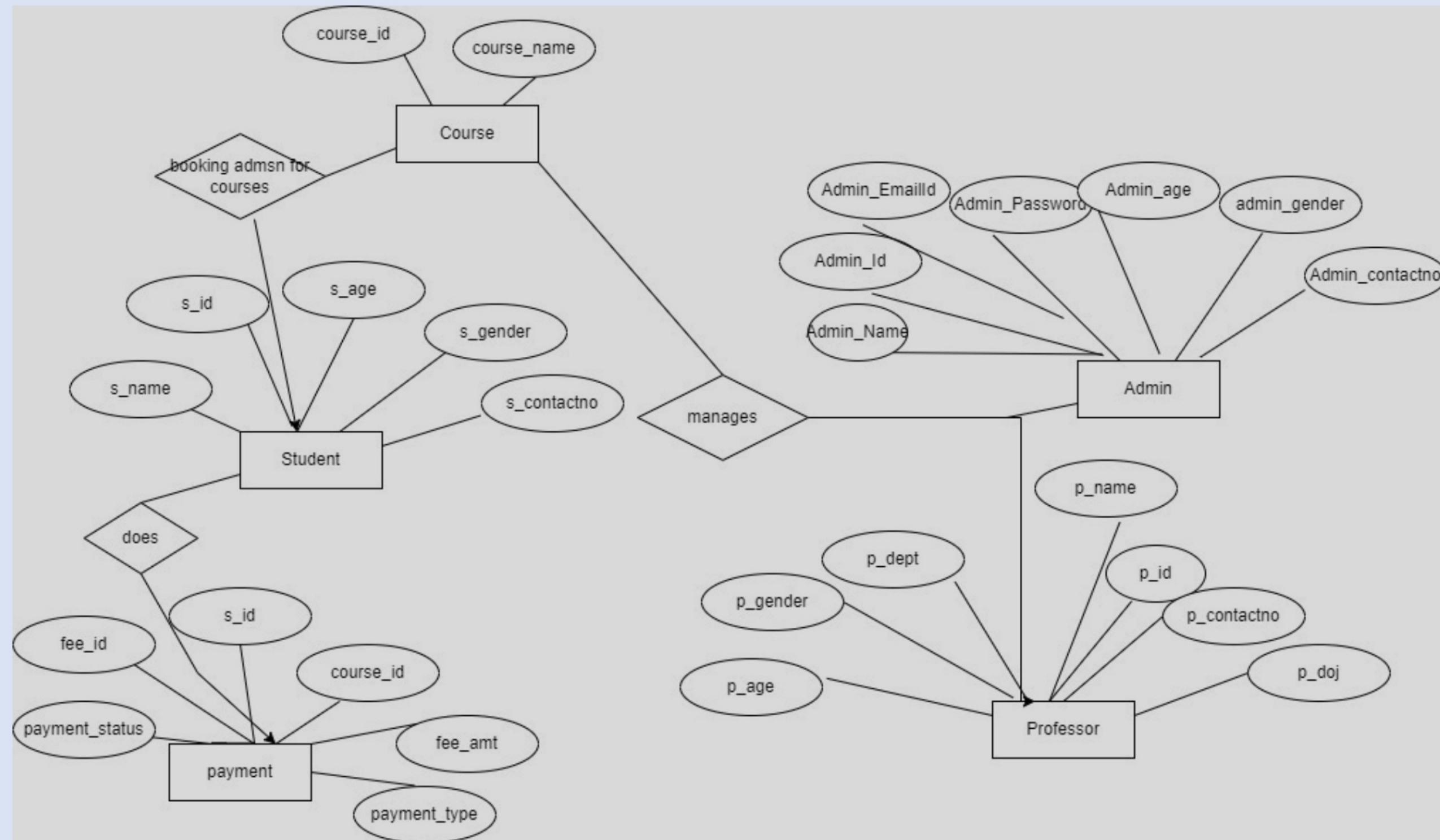
USE CASE DIAGRAM:

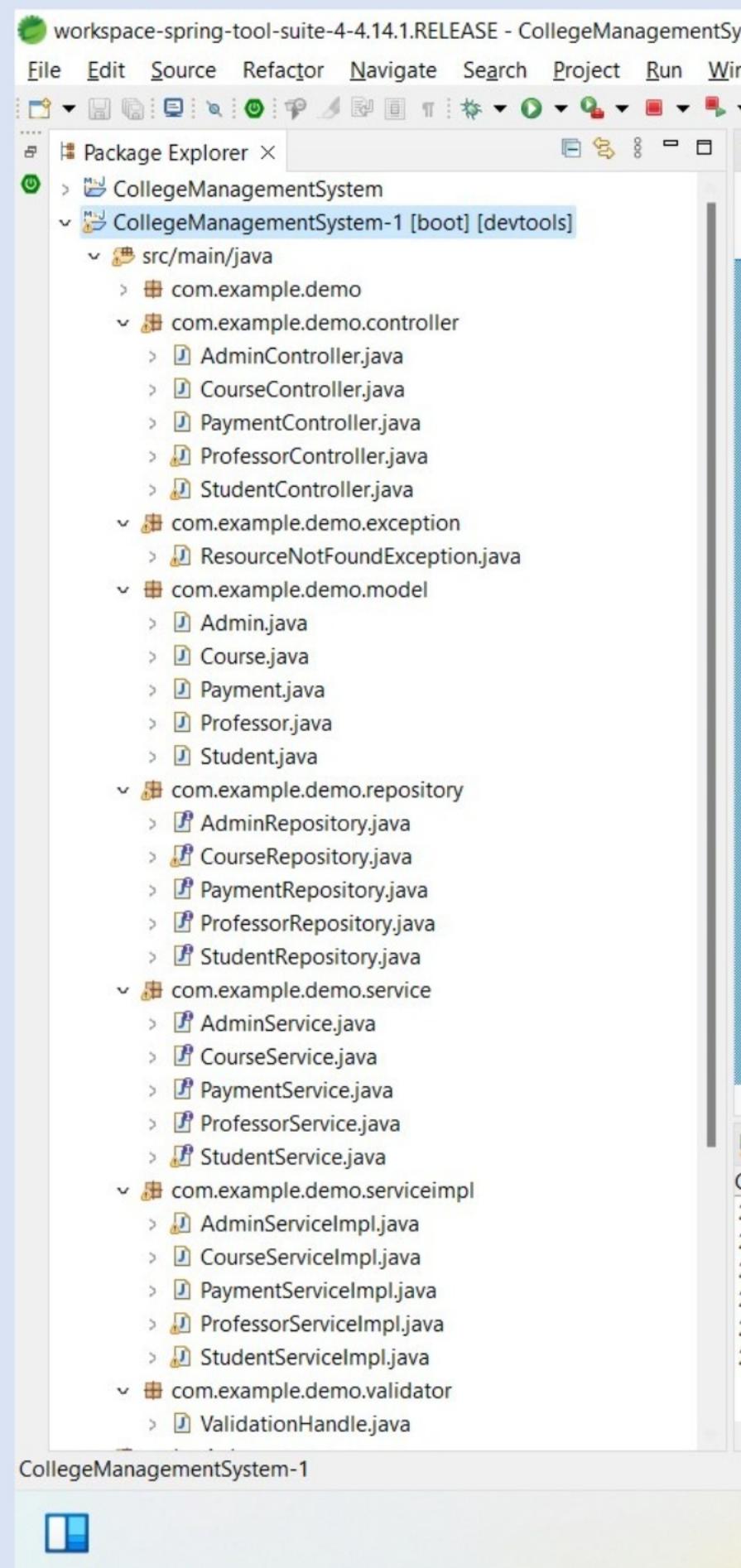


CLASS DIAGRAM:

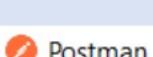


ER DIAGRAM





CollegeManagementSystem-1



File Edit View Help

Home Workspaces API Network Explore

Search

+ Invite



Upgrade

My Workspace

New Import

< Overview Profess POST Save/ POST Login/ GET GetAllA GET Get/ PUT Update/ DEL Delete/ POST SaveF POST S > + ... No Environment



Collections

+



...

CollegeManagementSystem

AdminManagement

POST SaveAdmin

POST LoginAdmin

GET GetAllAdmins

GET GetAdminById

PUT UpdateAdmin

DEL DeleteAdmin

ProfessorManagement

POST SaveProfessor

POST LoginProfessor

GET GetAllProfessors

GET GetProfessorById

PUT UpdateProfessor

DEL DeleteProfessor

StudentManagement

POST SaveStudent

GET LoginStudent

GET GetAllStudents

GET GetStudentById

GET UpdateStudent

DEL DeleteStudent

CourseManagement

GET AddCourse

CollegeManagementSystem / AdminManagement / SaveAdmin

POST http://localhost:8089/api/admin/register

Params Authorization Headers (9) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   ...
3   "adminEmailId": "sasinarayanan458@gmail.com",
4   ...
5   "adminName": "sasirekha",
6   ...
7   "adminPassword": "Sasi@1234",
8   ...
9   "age": 25,
10  ...
11  "gender": "female",
12  ...
13  "contactNumber": "8778123091"
```

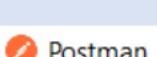
Body Cookies Headers (4) Test Results

🌐 Status: 400 Bad Request Time: 128 ms Size: 189 B Save Response

Pretty Raw Preview Visualize Text

1 OOPS!!! duplicate entry of email got detected

OUTPUT
SNAPSHOT-1



File Edit View Help

Home Workspaces API Network Explore

Search

Invite



Upgrade

My Workspace

New Import

< Overview Profess POST Save/ POST Login GET GetAllA GET Get/ PUT Update DEL Delete/ POST SaveF POST S > + ... No Environment



Collections

+



...

CollegeManagementSystem

AdminManagement

POST SaveAdmin

POST LoginAdmin

GET GetAllAdmins

GET GetAdminById

PUT UpdateAdmin

DEL DeleteAdmin

ProfessorManagement

POST SaveProfessor

POST LoginProfessor

GET GetAllProfessors

GET GetProfessorById

PUT UpdateProfessor

DEL DeleteProfessor

StudentManagement

POST SaveStudent

GET LoginStudent

GET GetAllStudents

GET GetStudentById

GET UpdateStudent

DEL DeleteStudent

CourseManagement

GET AddCourse

CollegeManagementSystem / AdminManagement / LoginAdmin

POST http://localhost:8089/api/admin/login

Params Authorization Headers (9) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {  
2   "adminEmailId": "rekhanarayanan458@gmail.com",  
3   "adminPassword": "Sasi@1234"  
4 }
```

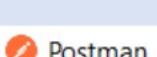
Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

```
1 {  
2   "adminId": 1000,  
3   "adminName": "sasirekha",  
4   "adminEmailId": "rekhanarayanan458@gmail.com",  
5   "adminPassword": "Sasi@1234",  
6   "age": 25,  
7   "gender": "female",  
8   "contactNumber": "8778123091"  
9 }
```

Status: 200 OK Time: 454 ms Size: 333 B Save Response

OUTPUT
SNAPSHOT-2



File Edit View Help

Home Workspaces API Network Explore

Search

Invite



Upgrade

My Workspace

New Import

< Overview Profess POST Save/ POST Login GET GetAll/ GET Get/ PUT Update DEL Delete/ POST SaveF POST S > + ... No Environment



Collections

+



...

CollegeManagementSystem

AdminManagement

POST SaveAdmin

POST LoginAdmin

GET GetAllAdmins

GET GetAdminById

PUT UpdateAdmin

DEL DeleteAdmin

ProfessorManagement

POST SaveProfessor

POST LoginProfessor

GET GetAllProfessors

GET GetProfessorById

PUT UpdateProfessor

DEL DeleteProfessor

StudentManagement

POST SaveStudent

GET LoginStudent

GET GetAllStudents

GET GetStudentById

GET UpdateStudent

DEL DeleteStudent

CourseManagement

GET AddCourse

CollegeManagementSystem / AdminManagement / GetAllAdmins

GET http://localhost:8089/api/admin

Params Authorization Headers (6) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

This request does not have a body

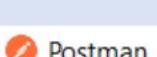
Body Cookies Headers (5) Test Results

Status: 200 OK Time: 145 ms Size: 504 B Save Response

Pretty Raw Preview Visualize JSON

```
1  [
2    {
3      "adminId": 1000,
4      "adminName": "sasirekha",
5      "adminEmailId": "rekhanarayanan458@gmail.com",
6      "adminPassword": "Sasi@1234",
7      "age": 25,
8      "gender": "female",
9      "contactNumber": "8778123091"
10     },
11    {
12      "adminId": 1001,
```

OUTPUT
SNAPSHOT-3



File Edit View Help

Home Workspaces API Network Explore

Search

Invite



Upgrade

My Workspace

New Import

< Overview Profess POST Save/ POST Login GET GetAllA GET Get... PUT Update DEL Delete/ POST SaveF POST S > + ... No Environment <>

Collections

+

CollegeManagementSystem

AdminManagement

POST SaveAdmin

POST LoginAdmin

GET GetAllAdmins

GET GetAdminById

PUT UpdateAdmin

DEL DeleteAdmin

ProfessorManagement

POST SaveProfessor

POST LoginProfessor

GET GetAllProfessors

GET GetProfessorById

PUT UpdateProfessor

DEL DeleteProfessor

StudentManagement

POST SaveStudent

GET LoginStudent

GET GetAllStudents

GET GetStudentById

GET UpdateStudent

DEL DeleteStudent

CourseManagement

GET AddCourse

CollegeManagementSystem / AdminManagement / GetAdminById

GET http://localhost:8089/api/admin/10000

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Send

Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (4) Test Results

Status: 400 Bad Request Time: 48 ms Size: 181 B Save Response

Pretty Raw Preview Visualize Text

1 Admin not found with AdminId : '10000'

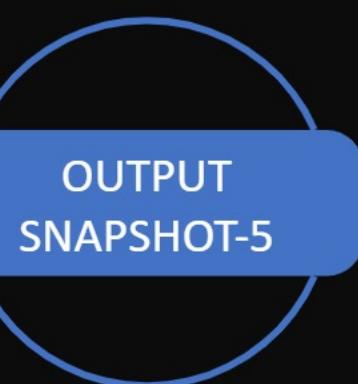
OUTPUT
SNAPSHOT-4

```
mysql> select*from student_table;
+-----+-----+-----+-----+-----+-----+
| s_id | age | contact_number | gender | email_id           | s_name      | pass_word |
+-----+-----+-----+-----+-----+-----+
| 501  | 24  | 9860004183   | female | vidya123@gmail.com | vidya       | Vidya@123  |
| 552  | 25  | 8660006783   | female | sasirekha500@gmail.com | sasirekha  | Sasi@123  |
+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

```
mysql> select*from course_table;
+-----+-----+-----+-----+
| course_id | course_fee | course_name | s_id |
+-----+-----+-----+-----+
| 5051      | 2500       | python      | 552  |
| 5052      | 5000       | data science | 552  |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

```
mysql> select*from payment_table;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| payment_id | card_number | cvv | exp_year | name_on_card | payment_date          | payment_status | s_id | total_payment | upi | course_id |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 7000       | 656354354653614 | 712 | 2028    | arav        | 2022-07-03 23:12:17.225000 | paid          | 552  | 2500          | NULL | 5051 |
| 7001       | 656354354653614 | 632 | 2023    | sravs       | 2022-07-03 23:22:00.408000 | paid          | 552  | 5000          | NULL | 5052 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

```
mysql> ■
```



```
mysql> select*from admin_table;
```

admin_id	email_id	admin_name	pass_word	age	contact_number	gender
1000	sravanthi123@gmail.com	sravanthi	Sravs@123	23	7989708734	female
1051	sasirekha123@gmail.com	sravanthi	Sasi@123	23	7989708734	female
1151	sasirekha456@gmail.com	sasirekha	Sasi@123	23	7989708734	female
1152	sravanthi1112@gmail.com	sravanthi	Sravs@1113	23	7989706789	female

4 rows in set (0.00 sec)

```
mysql> select*from professor_table;
```

p_id	email_id	pass_word	age	contact_number	gender	p_dept	p_doj	p_name
2000	bhanuprakash123@gmail.com	Bhanu@123	43	7666000843	male	computer science	2010/10/05	bhanuprakash
2001	prakash123@gmail.com	Prakash@123	45	7666000678	male	mechanical	2020/10/05	prakash

2 rows in set (0.00 sec)

```
mysql> select*from student_table;
```

s_id	age	contact_number	gender	email_id	s_name	pass_word
501	25	8660006783	female	sasirekha123@gmail.com	sasirekha	Sasi@123
552	25	8660006783	female	sasirekha500@gmail.com	sasirekha	Sasi@123
603	25	8660006783	female	sasirekha501@gmail.com	sasirekha	Sasi@123

3 rows in set (0.00 sec)

```
mysql> select*from student_table;
```

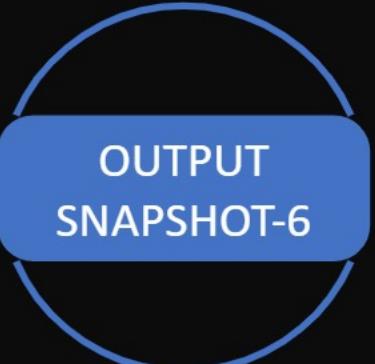
s_id	age	contact_number	gender	email_id	s_name	pass_word
501	24	9860004183	female	vidya123@gmail.com	vidya	Vidya@123
552	25	8660006783	female	sasirekha500@gmail.com	sasirekha	Sasi@123

2 rows in set (0.00 sec)

```
mysql> select*from course_table;
```

course_id	course_fee	course_name	s_id
5051	2500	python	552
5052	5000	data science	552

2 rows in set (0.00 sec)



Future Scope

- We can enhance this system by including more features like online examination module to conduct online examination.

- Further,The Faculty can upload the study materials of their lectures on this site and students who had missed those classes can view those study materials for better understanding.

CONCLUSION

- This paper presents a method for increasing information requested by students with the use of automated College Management System.In this ,instead of direct Contacting with faculty the student can directly checks the Courses and Fees Details from the System if the student is registered in CMS.

- CMS helps educational institute to do regular activities accurately, fastly and reliably.

THANK YOU!!

“
REFERENCE
”

- <https://www.w3schools.com/java/>

- <https://www.javatpoint.com/spring-boot-tutorial>