EX.NO:2 PASSPORT MANAGEMENT SYSTEM

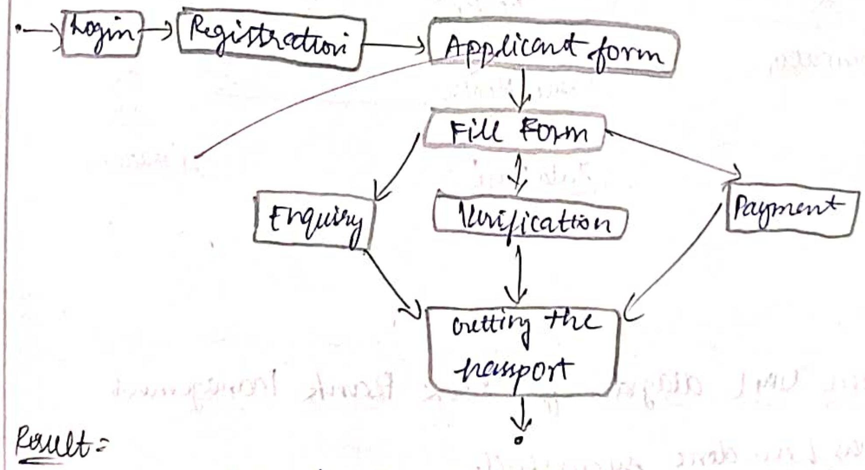
Aim:

To develop a system using UML for passport management.

Procedure Code:

```
#include <stdio.h>
#include <string.h>
typedef struct{
    char name [50];
    char address [100];
    char dob[20];
    int id;
}Applicant;
void submitApplicant(Applicant *applicant);
int verifyDetails (Applicant *applicant);
valid notifyApplicant(Applicant *applicant, int approval);
int main(){
  Applicant applicant;
  submitapplicant (Applicant);
  int approval = verify Details (Applicant);
  if(approval){
     printf("Issuing passport for applicant ID:%d\n",applicant,Id);
  }
void modify Applicant(Applicant *applicant, int approval)
{
 if(approval)
   {
     printf("Notification: Dear %s your passport has been issued".}
 else{
     printf("Verification");
```

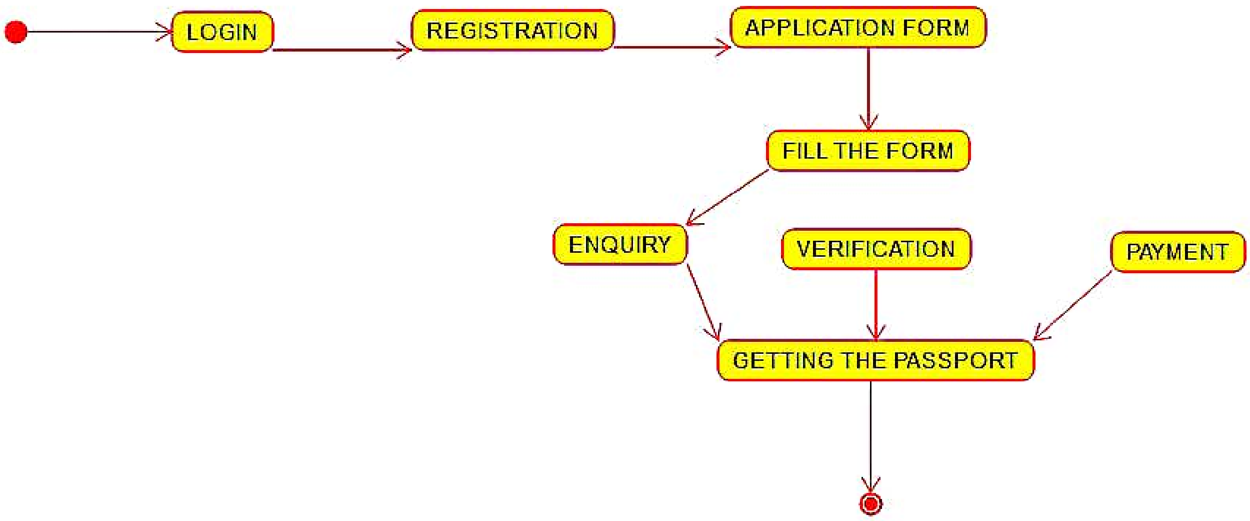
Diagram: Activity.



Result:

Thus, the UML diagram for passport management system is done successfully.

PASSPORT MANAGEMENT SYSTEM



EX:NO:3 Book Bank Management System.

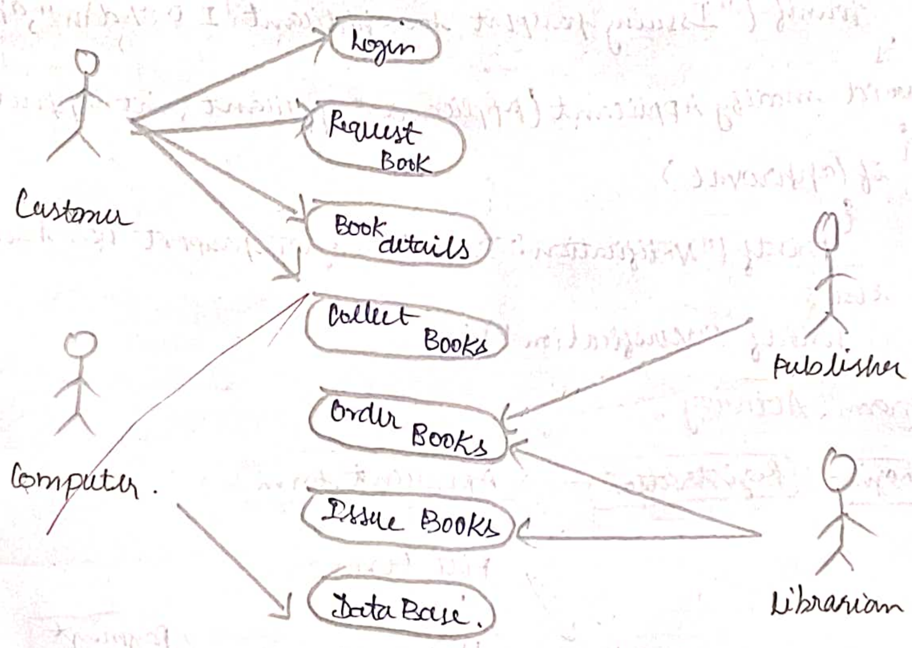
Aim:

To develop a system using UML for implementrying the Book Bank process.

Code:

```
Bank. System = Bank-System()
customer 1= customer (1,"customer", "bmail.com")
agent 1= agent (100, "John")
bank-system.add-customer (customer)
bank-system.add_agent (agent1)
bank-system.record-interaction (interaction)
for interaction in customer, interaction{
    print(" Interaction Io: Interaction. interaction-id),
    agent (Interaction. agent. name]
int main(){
  char student Details[ ]= "John Doe, Student ID =12345;
send details to Central Computer (Student Details);
return0;
}
```

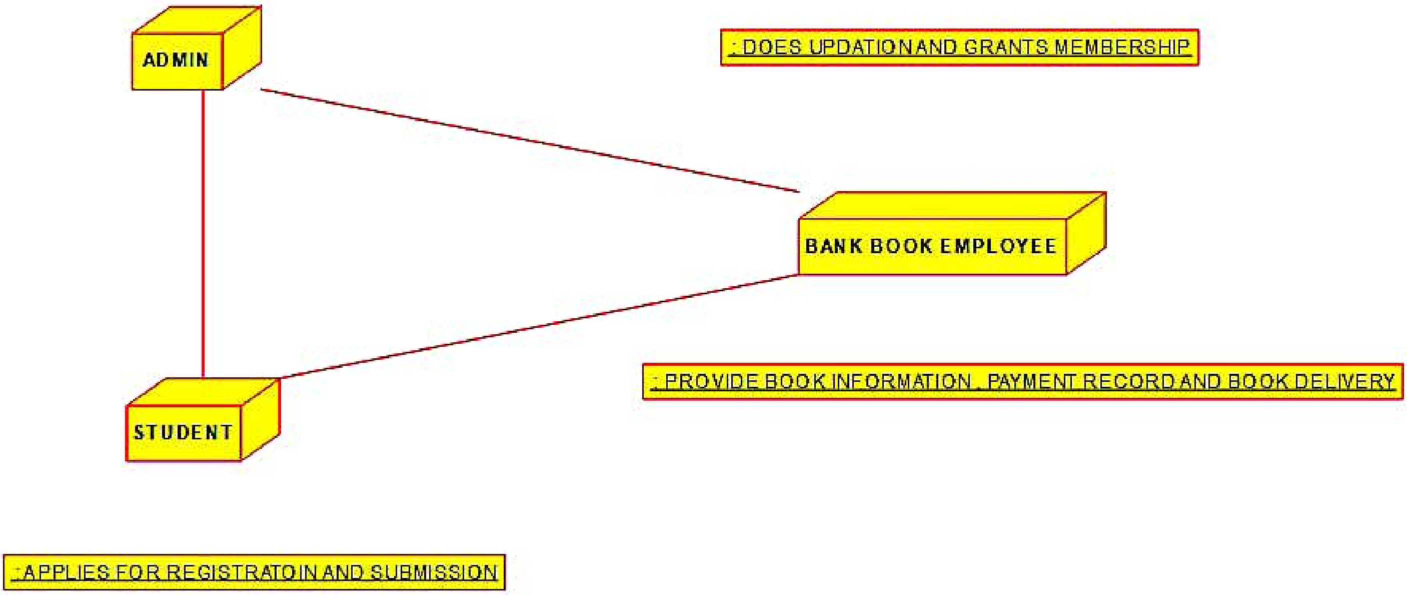
Diagram: Use Case



Result:

Thus, the UML diagram for Book Bank Management System has been done successfully.

BOOK BANK PROCESS


ADMIN
: DOES UPDATION AND GRANTS MEMBERSHIP
BANK BOOK EMPLOYEE
: PROVIDE BOOK INFORMATION , PAYMENT RECORD AND BOOK DELIVERY
STUDENT
:APPLIES FOR REGISTRATOIN AND SUBMISSION

EX: No: 4 Exam Registration Portal

Aim:

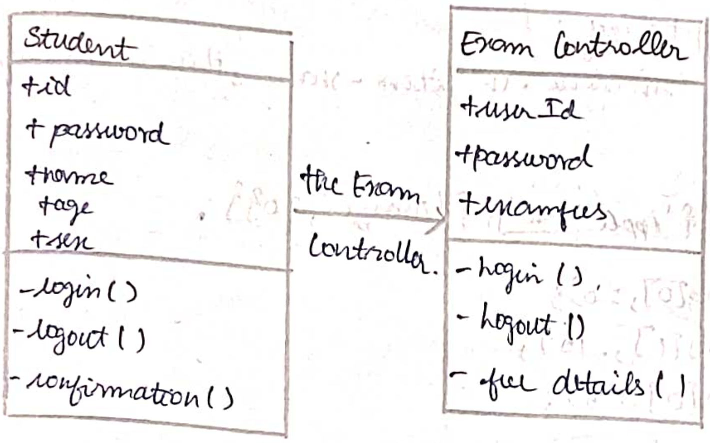
To dwelop a system using UML for implementing Exam Registration portal.

Code:

```
#include <stdio.h>
#include <string.h>
type def struct {
    char name [50];
    int registered ;
  } student ;
int main() {
  student   student1 = {"Alice", 0};
  student  student2 = {"Bob", 0};
  register = exam (Student);
  return 0;
}
```

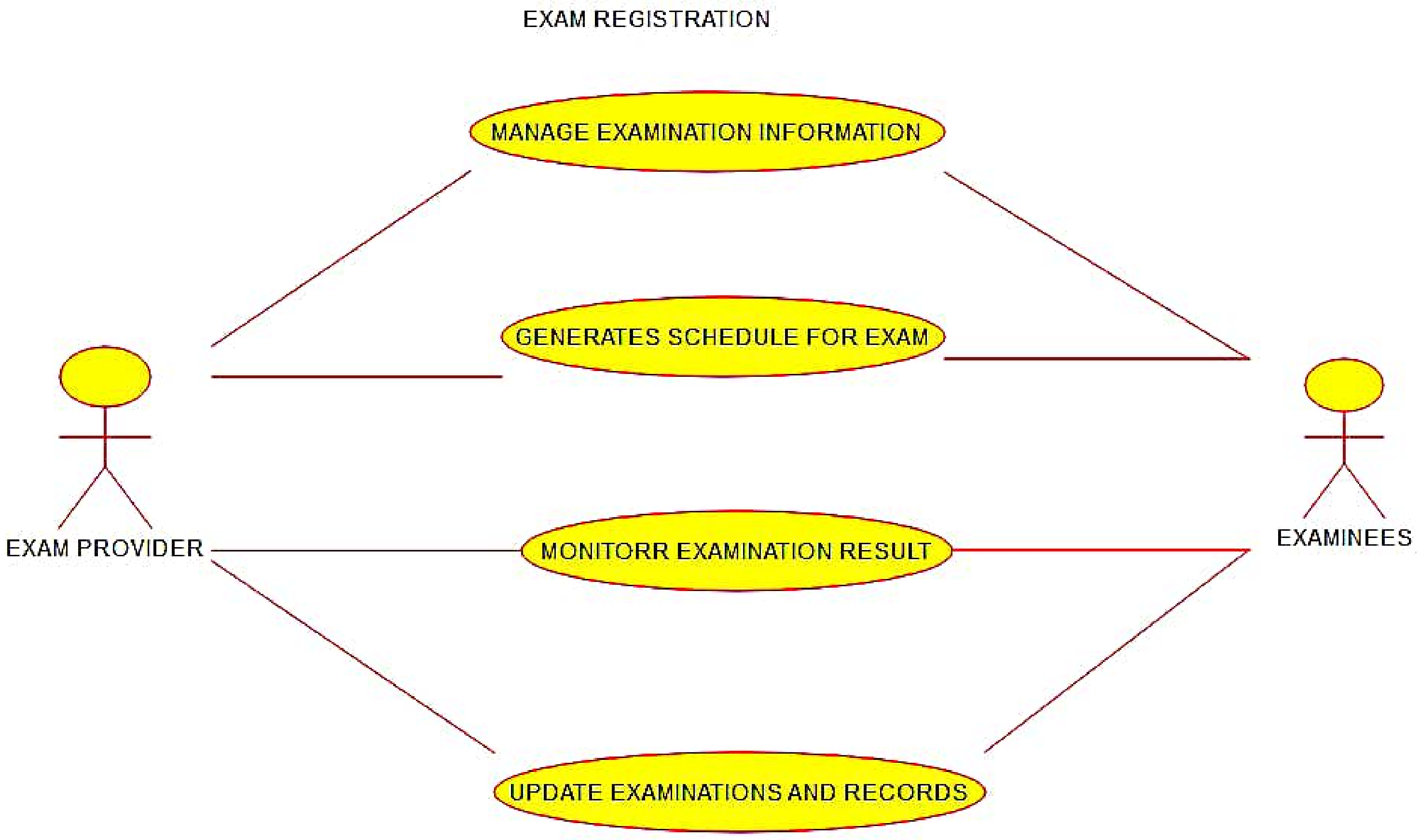
```
void issuetaken()
{printf("Office : Issue hallticket
                  to candiate");
printf("Office: Hall ticket issued
}                         successfully");
int main ()
{char candiatedetails[]="Jane
                      Joe, candiate
                      ID:67890]
 sendDetails To Control computer(
                  Candiate Details)
  return 0;
  }
```

Diagram: Class



Result:

The design for exam registration system is done successfully using the UML software.


EXAM REGISTRATION
MANAGE EXAMINATION INFORMATION
GENERATES SCHEDULE FOR EXAM
EXAM PROVIDER
MONITORR EXAMINATION RESULT
EXAMINEES
UPDATE EXAMINATIONS AND RECORDS

EX.NO: 5

Stock Maintenance System

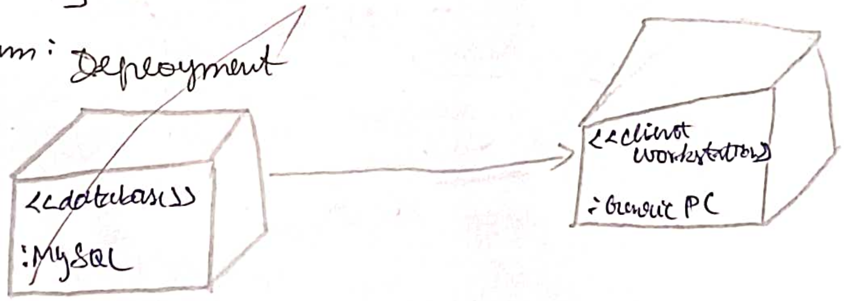
Aim:

To develop a system using UML for implementing Stock Maintenance System.

Code:

```
#include <stdio.h>
#include <string.h>
typedef struct{
    char name [50];
    int registered;
} student;
                    update_stock
void ~~registered_exam~~ (~~student *s~~){It
    printf("%s" %. registered.\n", s->name, s->registered ? "already" : "");
    s->registered = 1;
~~int main~~(){
  ~~student students []~~=
void update_stock (Item *item, int qty){
    item ->quantity += qty;
    printf("%s": %d in stock.\n", item->name, item -> quantity);
}
int main(){
  Item items[]={{'Apple', 50}, {'Orange', 30}};
  update_stock (&items[0], 20);
  update _stock (&items[1], -10);
  update _stock (&items[0], - 5);
  return 0; }
```

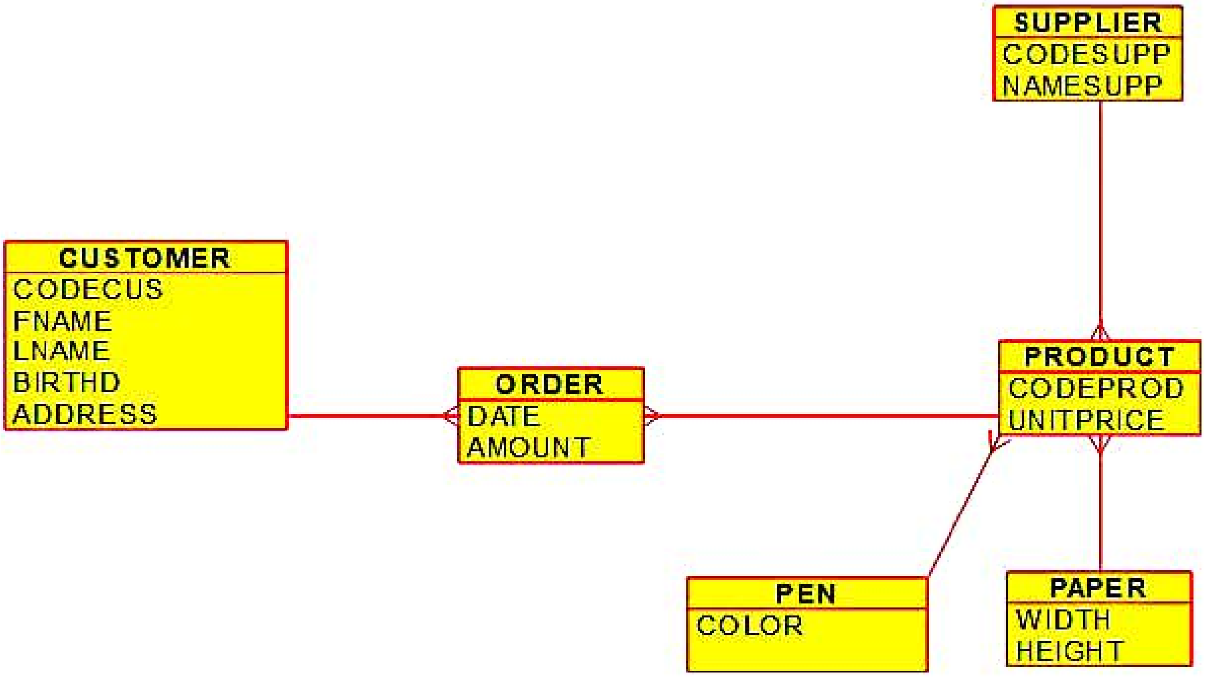
Diagram: Deployment



Result:

Thus, the UML diagram for Stock Market System executed successfully.

STOCK MAINTENANCE SYSTEM



EX.NO: 6 Online Course Reservation System

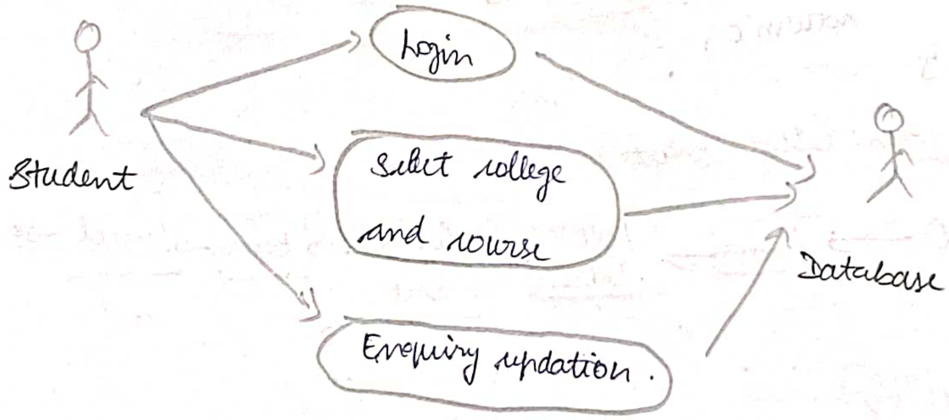
Aim:

To develop a system UML for implementing online course reservation system.

Code:

```
#include <stdio.h>
#include < string.h>
type def struct {
  char name [50];
  int registered;
 } Student;
void register_course (Student * s){
  printf ("%s, %registered.\n", s->name );
 s->registered = 1;
 }
int main(){
  Student Student [ ] = {{"Alice", 0}};
  register _course (&Student [0]);
 return 0;
 }
```

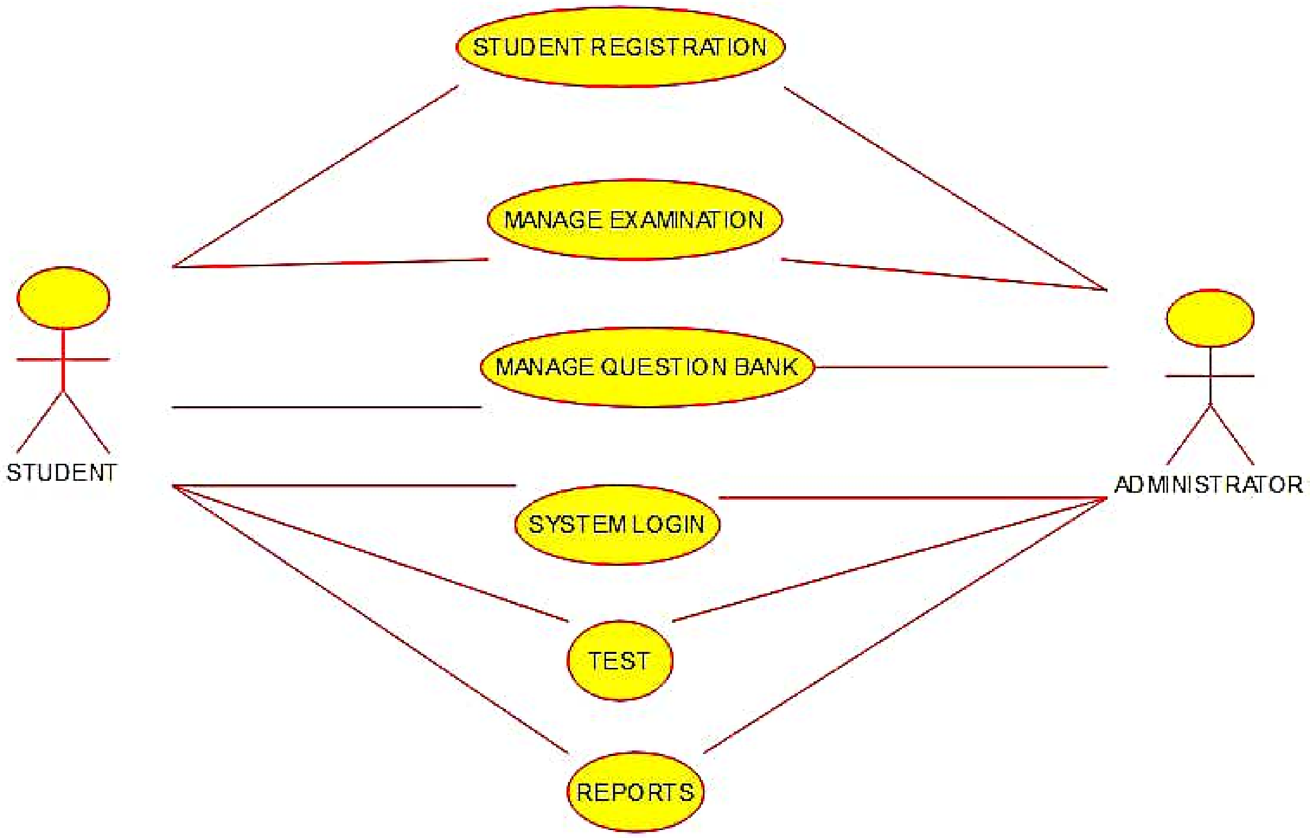
Diagram: Use case



Result:

Thus, the UML diagram for Online Course Reservation System successfully.

ONLINE COURSE REGISTRATION SYSTEM



Ex:No:7 E- Ticketing System

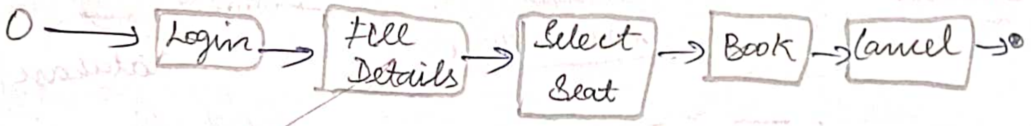
Aim:

To develop UML for implementing E-ticketing system.

Code:

```
#include <stdio.h>
typedef struct {
    int ticket_id;
     int booked; }
   Ticket;
   void book_ticket (Ticket *t){
          if (!t -> booked){
               t ->booked = 1;
              printf ("Tickett Booked");
              }
        else {
          printf("Already Booked");
         }
    }
 int main(){
          book_ticket (tickets);
          return 0;
    }
```

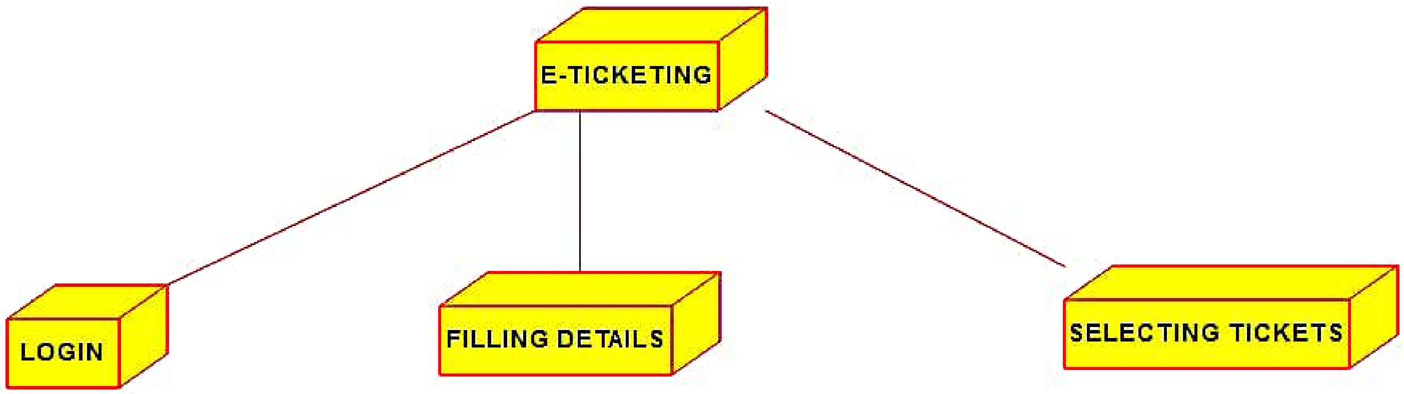
Diagram: (State chart)



Result:

Thus, the UML for implementing E-ticketing system has executed successfully.

E-TICKETING SYSTEM



Ex: No: 8 Credit Card Processing System

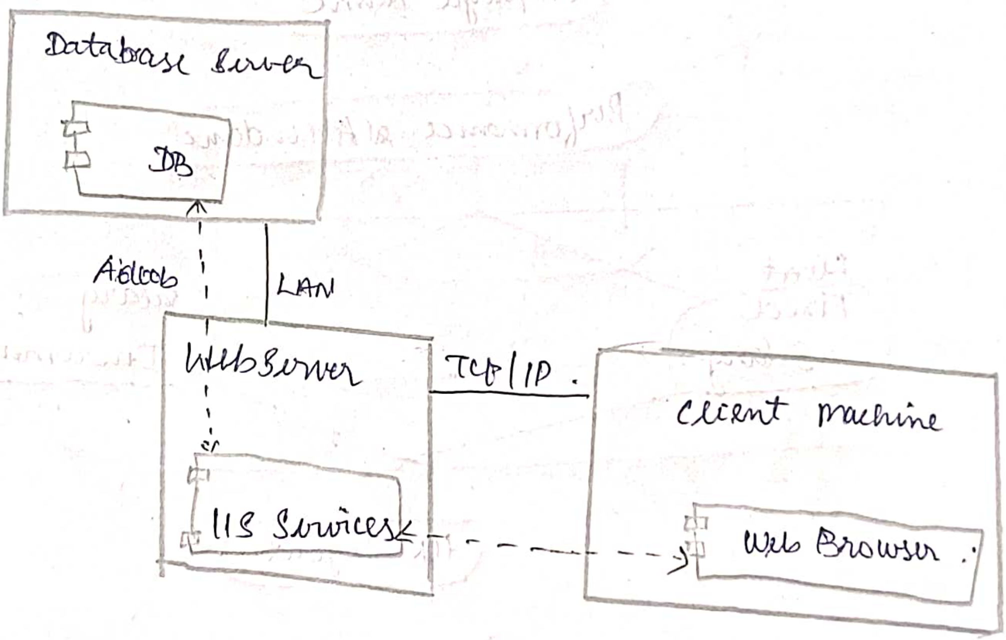
Aim:

To develop a system using UML for implementing credit card processing system.

Code:

```
#include <stdio.h>
typedef struct{
    char card_number [20];
   float balance;}
   Credit card;
  void procedure.payment (Credit Card *card, float amount);
int main ()
{
   Credit Card card = {"12.34567890", 100.0};
  process.payment (& card = 30.0);
  return 0;
 }
```

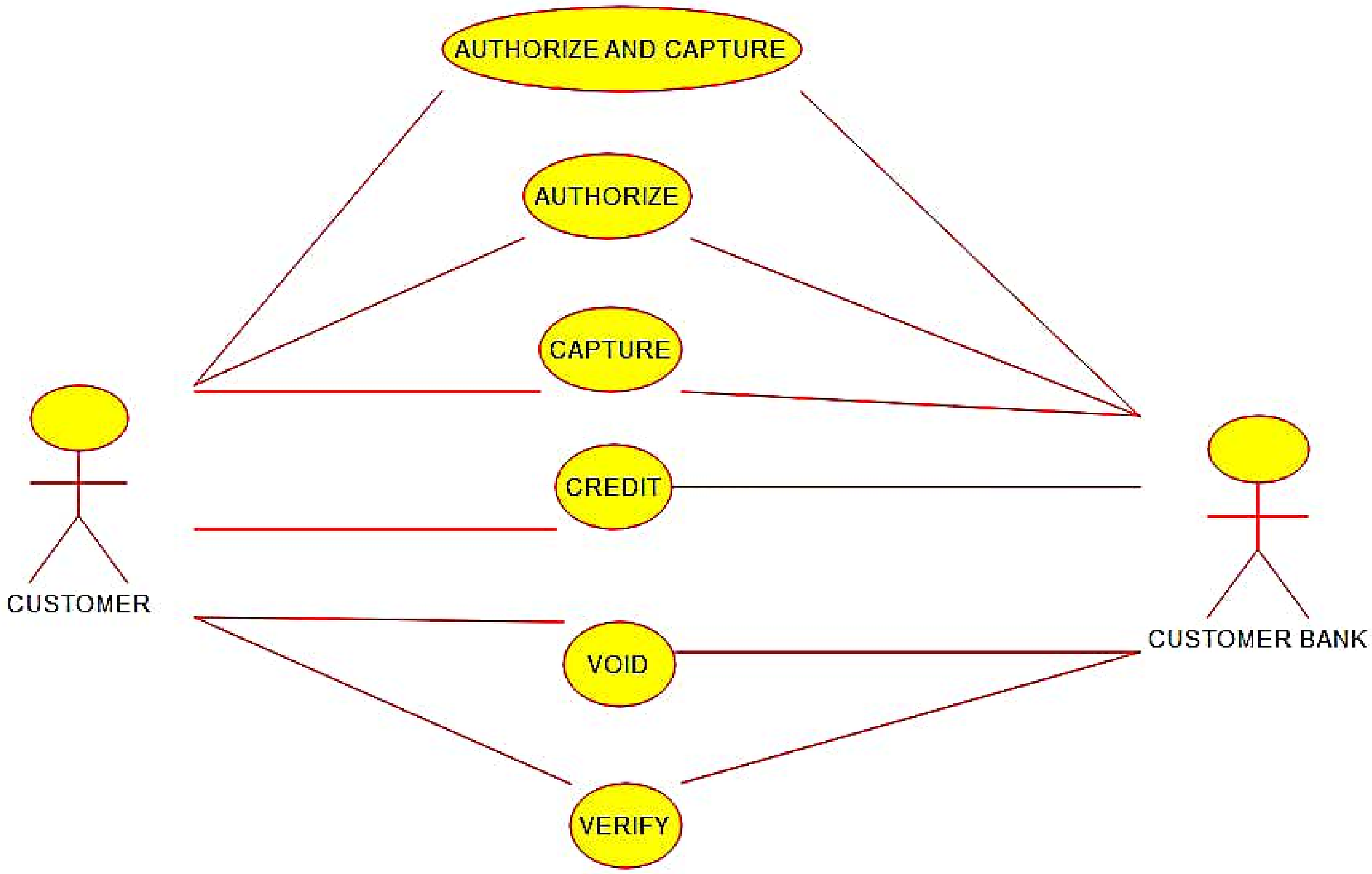
Diagram: (component)



Result:

Thus, the UML for implementing credit card processing system has executed successfully.

CREDIT CARD PROCESSING SYSTEM



EX.NO:9 Software Personal Management System.

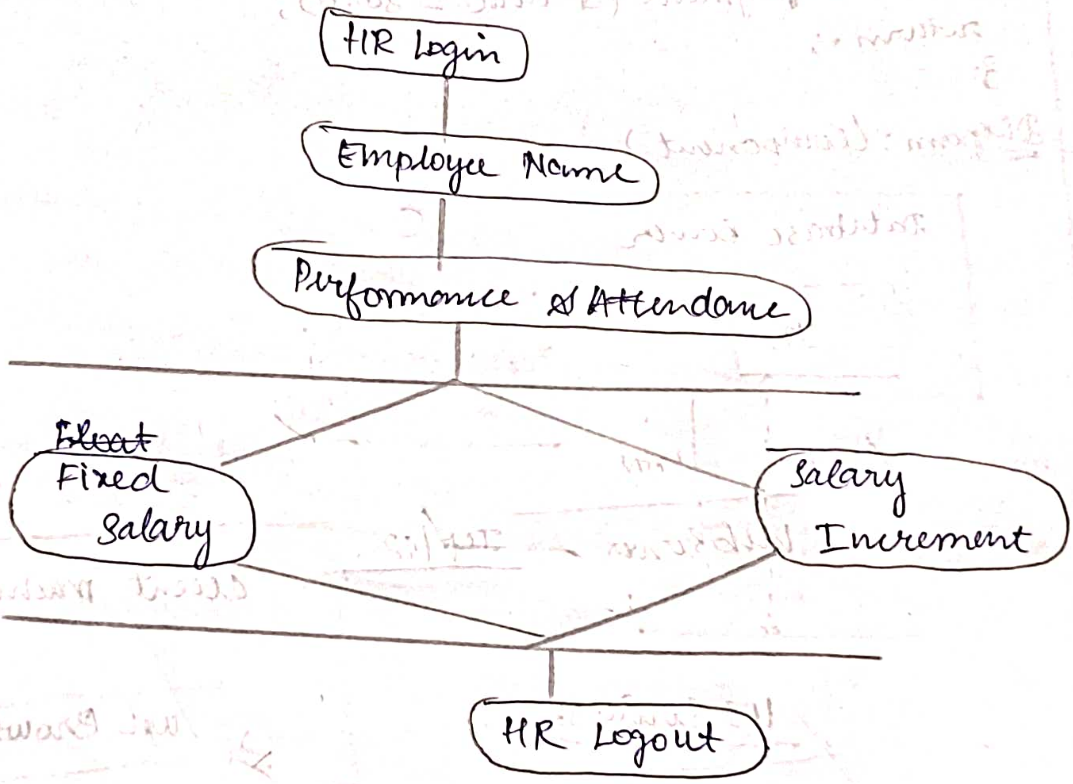
Aim:

To develop a system using UML for software personal management system.

Code:

```
#include <stdio.h>
int main(){
    personel staff[] = {{"Ram","develop"}};
    Updated_position(&staff[0]="Senior Developer};
  return 0;
}
```

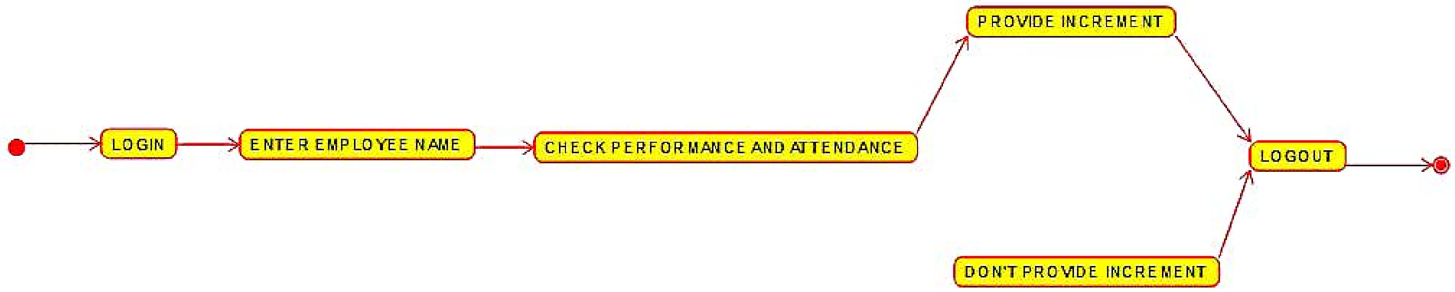
Diagram: (Activity):



Result:

Thus, the UML diagram for Software Personal Management System.

SOFTWARE PANEL MANAGEMENT SYSTEM



Ex:No:10 E-Book Management System.

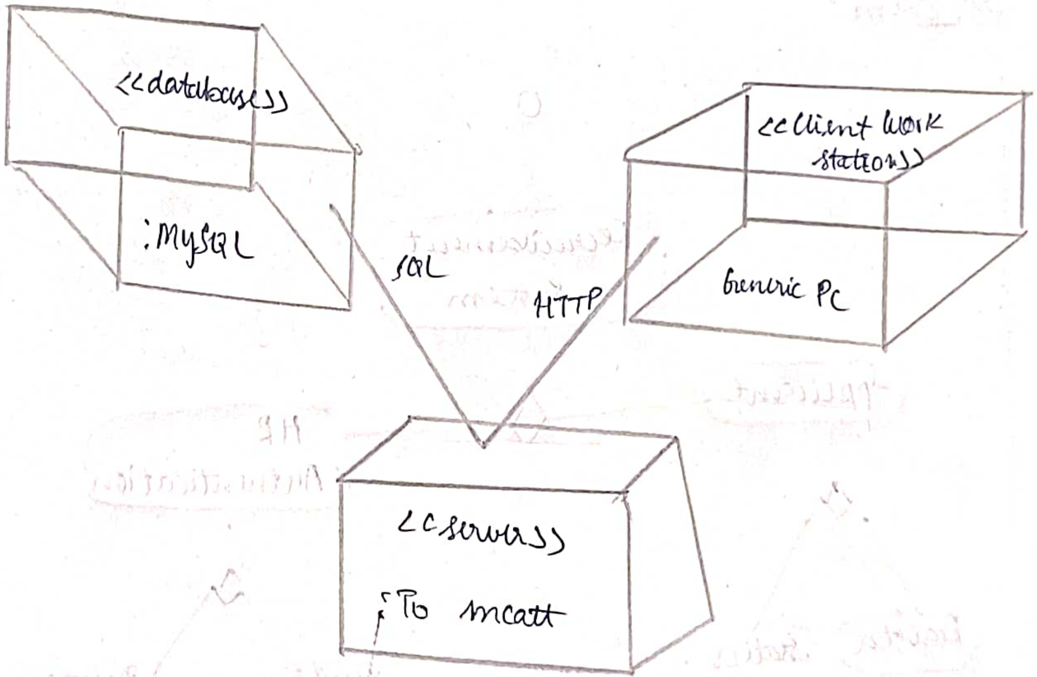
Aim:

To develop a UML for E-Book management system

Code:

```
#include <stdio.h>
typedef struct {
    char title[50];
    int available;
}EBook;
int main(){
  EBook books[] = {{"OOAD",1},{"AI",1}};
  borrow_book(&books[0]);
  return 0;
}
```

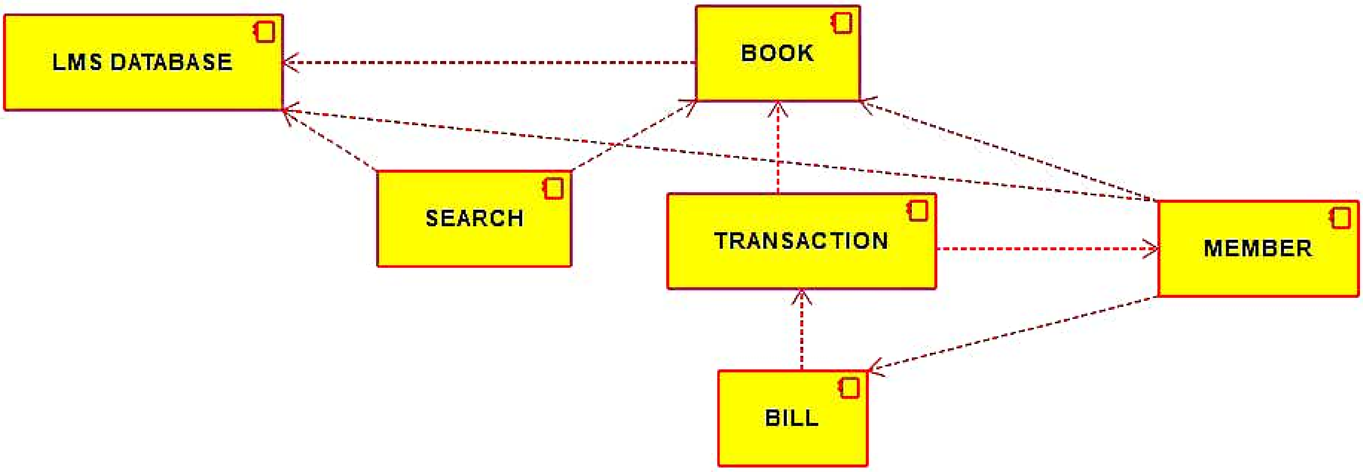
Diagram:



Result:

Thus, the UML for E-Book management system has implemented successfully.

E-BOOK MANAGEMENT SYSTEM



EX:NO:11 Recruitment System.

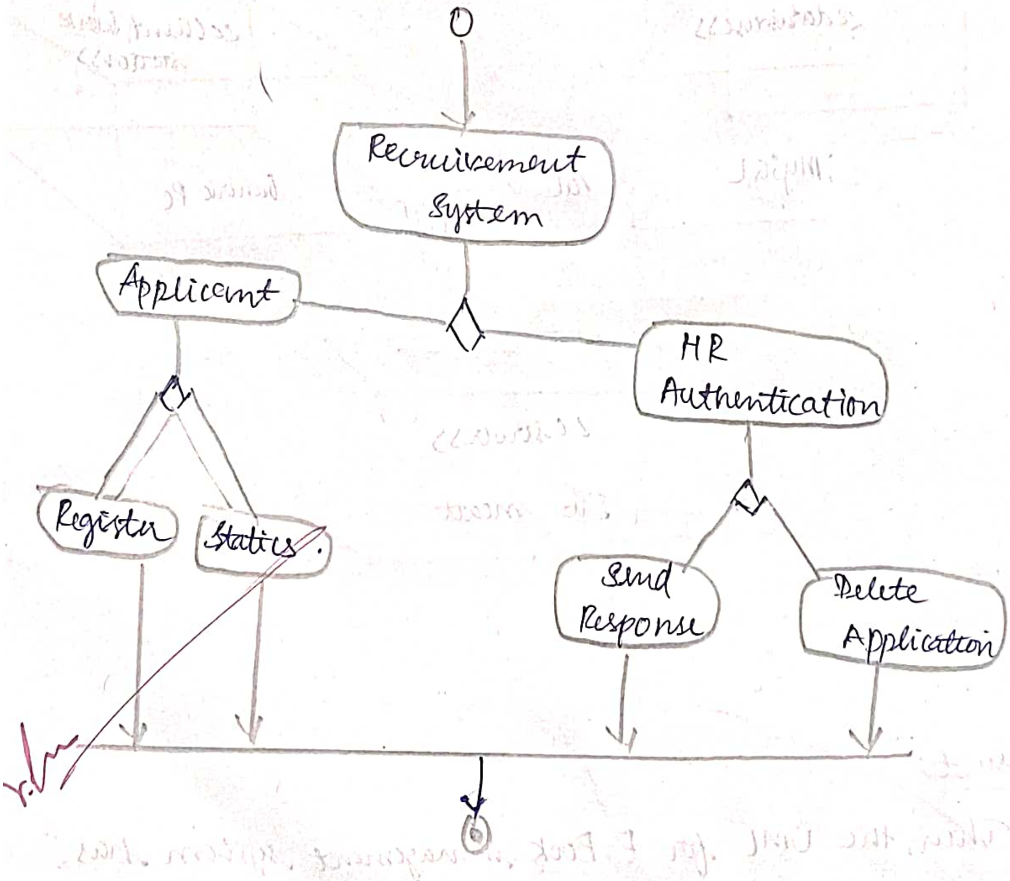
Aim:

To develop a system using UML of software personal management recruitment system.

Code:

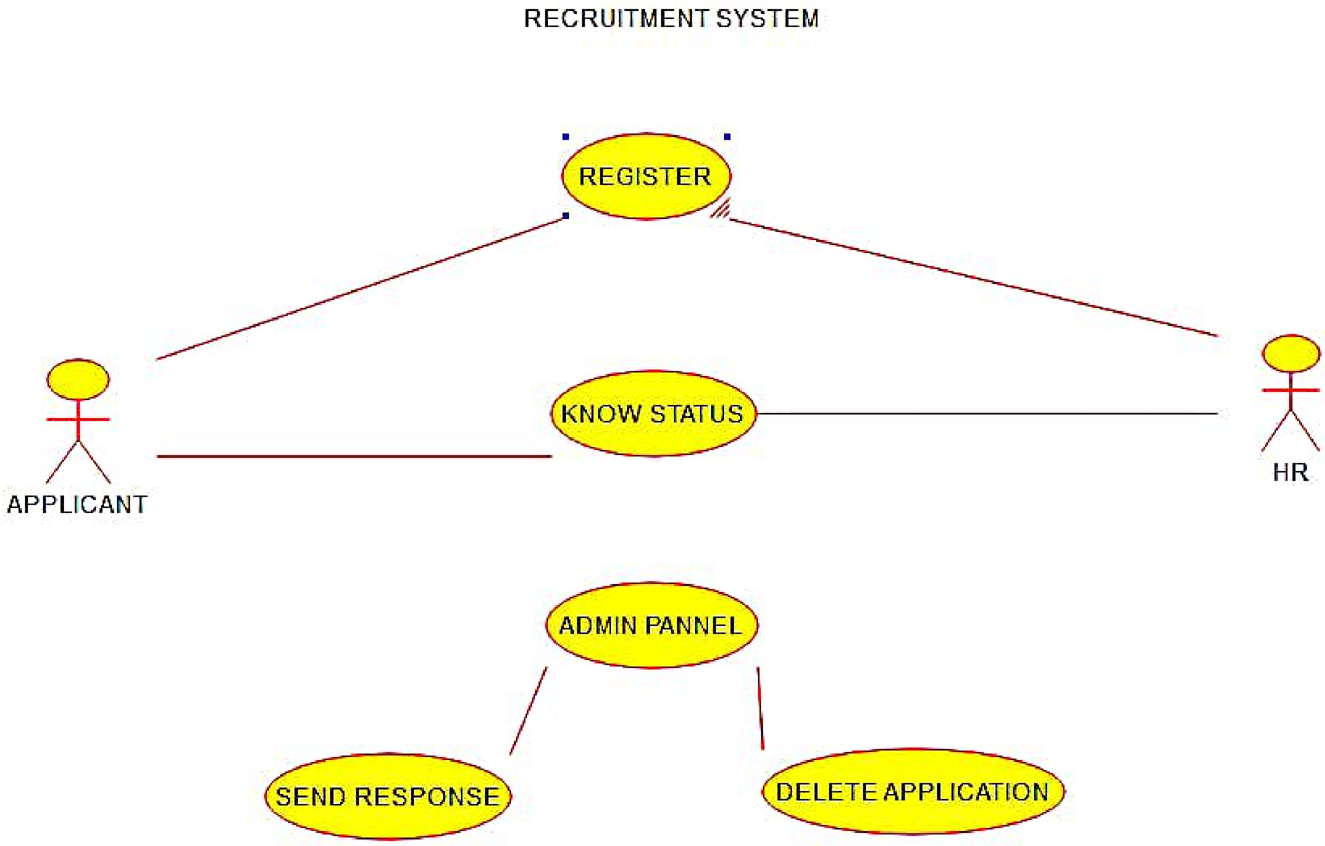
```
#include <stdio.h>
typedef struct {
   char name [50];
   int hired;
   } Candidate;
int main(){
   Candidate candidates [] = {{"Ram", 0}};
   Process_candidates ();
   return 0;
 }
```

Diagram:



Result:

Thus, the program of UML executed diagram and verified.


RECRUITMENT SYSTEM
REGISTER
KNOW STATUS
APPLICANT
HR
ADMIN PANNEL
SEND RESPONSE
DELETE APPLICATION