Category shld be shown after user select 1 as option; why before and why hardcoded category in main method

No input validation seen

```
In main method, No service layer method called to add details into map

List<TicketCategory>listTicketCategory(); : not implemented

Service class ok minus validatin methods

No test cases seen


Wy    public static String raiseNewticket(TicketBean ticketbean) {


TicketNumber1573logged sucessfully at2019-04-22
enter your category

why ask user to enter category again


not understood the flow of  your application
```

why hardcoded category in main method


No input validation seen

Service class impl : no validation methods seen

Delegation seen

Why ask ticket status from user ?

Ticket category ( how to know which value to enter; not shown to user )

How to know what to enter as priority ?

Why dao and service layer are returning false after adding details into map


```
TicketNumber 85 Logged Successfully at 2019-04-22 2019-04-22T01:50:09
```

No projects found to import ; looks like copy case

177395_Ticket_Management_System

Category hardcoded; method given in paper not used

No need of while loop after accepting priority from user

No input validation methods seen

Before adding ticket into map, all ticket attirbutes are null

Why return 0 from dao and service methods after adding into map ?

No junit

```
Enter Priority:   1:low 2:Medium    3:High
1
Ticket Number 1554 logged Successfully at 2019-04-23

Ticket No: 0
Ticket Description: null
Ticket Category Id: null
Ticket Priority: null
Ticket Status: null

Ticket No: 1001
Ticket Description: Problem with accessing software
Ticket Category Id: 1
Ticket Priority: Medium
Ticket Status: new

Ticket No: 1002
Ticket Description: Problem with accessing mail
Ticket Category Id: 2
Ticket Priority: High
Ticket Status: old
```

But overall good

C:\Users\kaviaror\OneDrive - Capgemini\team related\mohammed\SolutionsModule1TR6\Solutions Module 1 TR6\Ticket\177438_saitejaswi\17438_saitejaswi

No projects to import . how come ??

```java
if (ticketLog.put(ticketNo, ticketBean) != null)
            return true;
        else
            throw new TicketException("Item could not be raised");
```

exception is raised every single time

```java
public List<TicketCategory> listTicketCategory() { not implementd
```

hard coded values for category

no test case

```
Select Ticket Category from below list
1 : Software Installation

2 : MailBox creation

3 : Network issues

1
Software installed sucessfully
Enter your Name:
asdfasd
Select Ticket Category from below list
1 : Software Installation

2 : MailBox creation

3 : Network issues

1
Software installed sucessfully
Enter your Name:
```

Flow not right

Not accepting other required values from user

Below logic was not needed

```
        case 1:
            soffwareINstall();
                break;
            case 2:
                mailboxCreation();
                break;
            case 3:
                networkIssues();
```

input validation methods written in service layer but flownot right

if name does not match pattern, then no msg displayed and also the flow is wrong


loops not used correctly . program notexecuting as expectd

```
public int raiseATicket() {
        // TODO Auto-generated method stub
        return 0;
    }
```

No test case

Test case written but completely wrong


service.raiseNewTicket(tb,ticket); : no msg displayed after this

return type not considered

program does not run within loop

cannt execute fetch functionality

much better code than others

Unnecessary while loop

Call not delegated to service layer

Service class has only empty methods ; no implementation logic written

Same with dao layer

Only dummy msg shown that ticket logged

Compilation errors

Ui class not written as expected

Service layer method returning false/ null

Junit empty

Retest

Junit missing

Ui not as expected

Service layer /dao layer no logic written in methods

No code seen to enter data into map

Retest

```
        if (ticketLog.put(ticketNo, ticketBean) != null)
                return true;
```

above coding is not right at all

exception will always get thrown

```
public List<TicketCategory> listTicketCategory() {
        return null;
    }
```

Below is from service layer

```java
public List<TicketCategory> listTicketCategory() {
        // TODO Auto-generated method stub
        return null;
    }
```

No validation methods seen

Test cases written but giving compilation erors

Ui layer is ok but not delegating to service layer

i.e no logic seen for adding ticket into the hashmap

No projects found to import . looks like copy case to me

No code seen to put ticket into map

All logic in main method only

```java
public boolean raiseNewTicket(TicketBean ticketBean) {
        String category = null;
        String tcno = null;
        Util.raiseNewTicket(tcno,category);
        return false;
    }

    @Override
    public List<TicketCategory> listTicketCategory() {
        Util.listTicketCategory();
        return null;
    }
```

Did not understnad why return false  or return null

Test case not written

```java
        case 2:
```

```java
                    System.out.println("priority is medium");
                    System.out.println("Enter the ticket status:");
                    String status1=sc.next();
                    TicketBean tic1=new TicketBean();
                    int
ticketnumber1=(int)(Math.round(Math.random()*1024));
                    LocalDate ld1=LocalDate.now();
                    LocalTime lt1=LocalTime.now();
                    System.out.println("Ticket number"+"
"+ticketnumber1+" "+"logged sucessfully at"+" "+ld1+" "+lt1);

                    break;
              case 3:
                    System.out.println("priority is high");
                    System.out.println("Enter the ticket status:");
                    String status2=sc.next();
                    TicketBean tic2=new TicketBean();
                    int
ticketnumber2=(int)(Math.round(Math.random()*1024));
                    LocalDate ld2=LocalDate.now();
                    LocalTime lt2=LocalTime.now();
                    System.out.println("Ticket number"+"
"+ticketnumber2+" "+"logged sucessfully at"+" "+ld2+" "+lt2);

                    break;
```

repetitive coding seen

call not delegataed to service layer at all

```java
public boolean raiseNewTicket(TicketBean ticketBean) {
          ticketdao.raiseNewTicket(ticketBean);
          return false;
      }
```

Return from Methods of service layer not right

Dao methods not right

```java
public boolean raiseNewTicket(TicketBean ticketBean) {
      int max=500;
      int min=0;
      int range=(max-min);
    int ticketNo=(int)(Math.random()*range);


      return true;

}

      public static List<TicketCategory> listTicketcategory(){
            ticketcategory.get(ticketcategory);
            return null;
```

compilation errors

compilation errors

```
        if (ticketLog.put(ticketNo, ticketBean) != null)
                return true;
            else
                throw new TicketException("Ticket could not be raised");
        }

        @Override
        public List<TicketCategory> listTicketCategory() {
            return null;
        }
```

Test case empty

No validation methods seen

```
else if(categoryChoice == 2){
                    Random idr=new Random();
                    int idrd1=idr.nextInt();
                    ticket.setTicketCategoryId("tc002");
                    ticket.setTicketCategoryName("mailbox creation");
                    ticket = new TicketCategory(ticket.getTicketCategoryId(),
                                ticket.getTicketCategoryName());

            ticket=new TicketCategory(idrd1+"tc002", "mailbox creation");

            }else if (categoryChoice == 3) {
                    Random idr=new Random();
                    int idrd3=idr.nextInt();
                    ticket.setTicketCategoryId("tc003");
                    ticket.setTicketCategoryName("mailbox issues");
                    ticket = new TicketCategory(ticket.getTicketCategoryId(),
                        ticket.getTicketCategoryName());

                     ticket=new TicketCategory(idrd3+"tc003", "mailbox
issues");
                    }

Repetitive coding seen
```

```java
    @Override
    public boolean raiseNewTicket(TicketBean ticketBean) {
        Util.raiseNewTicket(ticketBean);
        return true;
    }

    @Override
    public List<TicketCategory> listTicketCategory() {
        List<TicketCategory> list = Util.listTicketCategory();

        return list;
    }
```

Service layer good