PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
100 Ft. Road, BSK III Stage, Bengaluru – 560 085

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Course Title: Problem Solving with C Laboratory		
Course code: UE19CS152		
Semester : II sem	**Section: A**	**Team Id: 2**
SRN: PES1UG19CS504	**Name: Sri Ramya Priya .V**	
SRN: PES1UG19CS204	**Name: K. K. Sivaani**	
SRN: PES1UG19EC197	**Name: Pooja CR**	
SRN:	**Name:**	

PROJECT REPORT

Problem Statement

To develop railway reservation system, using which, the user can login into the system, view the trains and the number of berths available for his/her journey, book his/her tickets and also cancel them if not required.

Description

Goals and objectives:

1. To develop a robust, error free and user friendly system of reservation in trains.
2. To enable the users to quickly find out the various trains and number of berths available for their journey.
3. To enable the users to book tickets.
4. To enable the users to cancel the booked tickets.

Problem to be addressed:

Train travellers face a lot of difficulty in knowing the various alternatives available for their journey. Also, the number of berths in various trains, in various classes and on various dates keep

changing on a real time basis. Travellers need the train information and berth information on a real time basis and they should be able to book/cancel tickets easily and quickly online.

Approaches and execution methods:

The system has the following modules:

1. Login module:

   This module first attempts to find whether the user is an existing user or a new one. It enables the existing users to login into the system with their credentials i.e. usernames and passwords. A csv file named usercredentials.csv has been created. This file contains a list of all usernames and passwords of all registered users. If the user is able to key in correct username and password, as available in the database i.e. usercredentials.csv file, the system logs in him/her. If the user is a new user, the module asks him/her to register, by entering new username and password, which get appended into the database. For the purpose of security, the user is asked to re-login using these credentials.

2. Booking module:

   This module gets the travel requirement of the users i.e. source station, destination station and date of travel. After obtaining these details, the program displays the list of available trains for that particular journey from the database file. The program displays the number of available berths in each class in the train (2A, 3A, SL). The user then can enter the class in which he/she wants to travel. The user then can select the train and enter the required passenger details such as name, age and gender. The program then, allots user his/her seats and displays the ticket with travel details and confirmation of reservation i.e. confirmed or waiting list. After this, the program subtracts that many number of berths from the database of that train in that class on that date, so that they are not available for the next user. The program can also handle the status of waiting list and also the combo cases, where a certain number of passengers are in confirmed status and a certain number of passengers (of the same ticket) are in waitlisted status.

3. Cancellation module:

   This module enables the user to cancel his/her reservation. After this, the program adds back that many number of berths to the database of that train in that class on that date, so that they are available for the next user.

C-concepts used

1. Structures:

We have used the following structures:

1. login – members used: userid, pwd
2. station – members used: stn1, stn2
3. pnr – members used: train_no, date, train, class, active_status etc..
4. user_booking- members used: ch, train_no, train_class, date, no_of_persons, first_name, last_name, person_age
5. booking_database- members used: berth_2A, berth_3A, berth_SL, train, date_of_travel

2. Arrays:

We have used arrays for:

1. berths in each train for each date. For the purpose of project, we have added 36 trains and 17 dates i.e. from for 3 classes (2AC, 3AC, SL) for each of those trains in the database. We have used 2 berths in 2AC, 3 berths in 3AC and 1 berth in Sleeper class as default available no. of berths in each train on each date. This has been intentionally done to demonstrate the capability of program to show waitlist status.
2. To store details of tickets booked i.e. PNR etc.

3. File Management:

We have created files for:

1. Storing usernames and passwords of registered users. The program gets details from the file to login the user. Also, new users can register and the details are appended into the file.
2. Storing the available trains between two stations. Upon entering the source and destination stations by the user, the available trains are displayed from the file.

4. Control Statements:

We have used control statements like: looping structures – (while, do-while, for), nested loops involving them and if-else statements.

5. Random function:

We have used it to generate a random 7-digit PNR number, while booking the ticket.

Learning Outcome

Before attempting the project, we had to first understand various features of railway reservation system. Based on these features, we had to conceive the appropriate system of design. Initially, we started with keeping all the data in arrays, but then we understood that as the data was huge it was not possible to put it in arrays. Also from a practical point of view, when a user uses this system he/she cannot be expected to keep all these data in the program itself. So, we started used csv files as database.

Hence, we thought it appropriate to use files as data structures for this particular system and hence we changed a part of the program using files where the data has been put into different files. We used fopen() command to read all the data from the files and display it to the user and also write the data into the files in append mode, so that all the data gets appended to these files which could be retrieved later.

However, we did not use only files. We used arrays also and kept a part of the data in arrays. For certain dynamic data like vacant berths, PNR numbers etc. we used arrays. We learnt that arrays are more suitable to handle dynamic data and files are more suitable to handle static and huge data.

Output Screenshots

5 screenshots of 5 different modules are given below:

Initial login for existing user:

```
Enter 0 if you are an existing user; Enter 1 to sign up as a new user:  0
Enter Username:  Ramya
Enter Password:  123
                Userid & Password matched!  Successfully logged in!
```

Registration of new user and login with the newly added credentials:

```
Enter 0 if you are an existing user; Enter 1 to sign up as a new user:  1
Choose username (Max 20 characters):  Manasi
Choose password(Max 10 characters):  15#&r
Enter 0 if you are an existing user; Enter 1 to sign up as a new user:  0
Enter Userid:  Manasi
Enter Password:  15#&r
                Userid & Password matched!  Successfully logged in!
Enter 0 to book tickets and 1 to cancel tickets:
```

Train and berth information to user based on user's requirement of route and date:

```
Enter 0 to book tickets and 1 to cancel tickets:  0
Enter source station:  bangalore
Enter destination station:  delhi
Trains between bangalore and delhi: 1. 17001 Rajdhani Express 2. 12017 Karnataka Express
                    Enter the train number to be booked:  17001
                Enter date of travel in dd/mm/yy format:  19/04/20
Berths available in train 17001 for date 19/04/20 in 2nd AC:2   , 3rd AC:3  , Sleeper:1
```

Selection of Class and Entry of details in Booking module:

```
Enter the class(2A for 2nd AC, 3A for 3rd AC, SL for Sleeper:  SL
Enter the number of berths:  4
Enter Name:  Pooja
Enter Age:  19
Enter Gender (Male/Female):  female
Enter Name:  Shivani
Enter Age:  19
Enter Gender (Male/Female):  female
Enter Name:  Ramya
Enter Age:  19
Enter Gender (Male/Female):  female
Enter Name:  Krithi
Enter Age:  29
Enter Gender (Male/Female):  female
Amount has been deducted from your account.     Ticket succesfully booked!!
```

Display of ticket after booking:

```
**********TICKET**********

PNR 7329434 Date: 21/04/20  Train: 17001    Class: SL

NAME            GENDER  AGE STATUS
Pooja           female  19  CONFIRMED
Shivani         female  19  Waiting List 1
Ramya           female  19  Waiting List 2
Krithi          female  29  Waiting List 3

**************************************************
```

Display of updating status that after booking of ticket (for our information):

```
Remaining ... berths in this train and date is -3

Adding.....7329434
Added 1 PNR to PNR ID 7329434
```

Cancellation module (details after cancellation message are for our information):

```
Enter 0 to book tickets and 1 to cancel tickets:  1
Enter the PNR number to cancel:  3911940

Cancellation Successful!
Earlier Available SL Births for train and Date were:  -1

Now Available SL Births for train and Date are:  1
```

Name and Signature of the Faculty