



Department of Software Engineering

Course Code: SE 133

Course Title: Software Development Capstone Project

Project Title:

Experience & Contact Diary

Submitted to: Nusrat Tasnim

Submitted by: Md Shamim Ahamed

Section: A

Student ID: 213-35-775

Date of submission: 12/09/2022

ABSTRACT

This project will be extremely helpful for those people who enjoy to note their daily special moment to a note and those who wants to find information of an unknow number. I chose this project as it focuses on some notable section including time save, saving nature by avoiding wood paper diary to keep healthy environment, they can also find spam calling or unknow number easily with this system.

This system is designed for all kind of users from different ages. It does not matter if someone from old generation or new generation they can note their experience and view it and enjoy those moments in future or they can note their experience for future generation to know which will never be lost. The reason this system is password protected so it certainly protects user privacy and create a good user experience

At current time spam call become part of our daily life. We can find those spam calls and their necessary basic information from this system.

Table of Contents

Chapter 1

| | |
|--|----|
| Introduction----- | 04 |
| 1.1 About the system----- | 04 |
| 1.2 Purpose----- | 04 |
| 1.3 Why this system us necessary?----- | 04 |

Chapter 2

| | |
|---------------------------------------|----|
| Features----- | 05 |
| 2.1 About the features of system----- | 05 |

Chapter 3

| | |
|---|----|
| Implementation----- | 06 |
| 3.1 C concepts we used for our project----- | 06 |

Chapter 4

| | |
|-----------------------------------|----|
| System Testing----- | 11 |
| 4.1 Introduction----- | 11 |
| 4.2 Input and desired output----- | 12 |
| 4.3 Report summery----- | 19 |

Chapter 5

| | |
|------------------------------------|----|
| Conclusion----- | 21 |
| 5.1 Good features----- | 21 |
| 5.2 Limitation of the system ----- | 21 |
| 5.3 Features Enhancement----- | 21 |

Chapter 6

| | |
|------------------|----|
| User manual----- | 22 |
|------------------|----|

Chapter-1

(Introduction)

1.1 About the system

The system aims to serve all kind of user from different ages. Every user can add their daily experiences, view those experiences, they can remove those experience if they feel like they do not need those, they can also edit those experience. Moreover, users can find spam call details and their information. With this service, a new whole unique environment will be created for diverse types of users.

1.2 Purpose

The prime purpose of this system is to introduce everyone to digital diary and contact system. The system will change the traditional way to note down every experience to wood paper diary and make it more digital. The system aims at keeping a good user-friendly environment.

1.3 Why this system is necessary

The importance of the system is undeniable. The system will have a positive impact on the users. With this system user can add, view, remove or change their daily experience and note those. Moreover, they can find spam calls. Which will remove their extra hassle to find unknow number.

Chapter-2

(Features)

2.1 About the feature of system

This project holds some notable features like Universal Contact, Add Experience just by creating an account to the system, View Experience, Edite Experience, Remove Experience. Now users do not need to waste wooden paper anymore to note down to their daily experience. They also do not need to spent their time and cash to find unknow number or spam callers.

This system is also producing the functionality to add contact number, view contact, search universal contact information and help others to an unknown number.

Chapter-3

(Implementation)

3.1 C concepts we used for our project

```
printf("\t\tPress '1' to Register\n\t\tPress '2' to Login\n\n");
printf("\n\nEnter your choice : ");
scanf("%d",&option);

gets(filename);
```

Figure-1 (input and output)

In Figure-1, we used **scanf()** and **printf()** to input and output command and its whole built in function is stored in **#include<stdio.h>** header file. Next **gets()** is used to input and its built-in function is stored in **#include<string.h>**.

```
printf("\tENTER NAME OF THE PERSON NEXT TO YOU:");

fflush(stdin);

gets(customer.name);

fflush(stdin);

printf("\tENTER PLACE:");

gets(customer.place);
```

Figure-2 (Input buffer)

In Figure-2, **fflush(stdin)** is used to flush or clear input buffer. It is beneficial to clear the input buffer with **fflush(stdin)** before taking using input (here we use **gets()**) statement.

```

if(strcmp(username,l.username)==0 && strcmp(password,l.password)==0)
{
    printf("\nSuccessful Login\n");
}
else
{
    printf("\nIncorrect Login Details\nPlease enter the correct credentials\n");
}
}
getch();

system("cls");

```

Figure-3 (Compare two variable, pause system, clear screen)

In Figure-3, **strcmp(variable1, variable2)** is used to compare between two variables. If those two variables are same then it will result 0. Next **getch()** is used to get a character input from keyboard and it stored in **#include<conio.h>** header file. Last **system("cls")** is used to clear the whole output windows screen. This function is stored in **#include<stdlib.h>** header file.

```

if(option == 1)
{
    system("CLS");
    registration();
}

else if(option == 2)
{
    system("CLS");
    login();
}

```

Figure-4 (if..else if statement)

In Figure-4, we use **if ... else if** statement. The **if...else if** statement executes two different codes depending upon whether the test expression is true or false. Sometimes, a choice must be made from over 2 possibilities. The **if...else if** ladder allows us to check between multiple test expressions and execute different statements.

```
while(i<3){  
printf("%c",178);  
sleep(1);  
i++;  
}
```

Figure-5 (Loop)

In Figure-5, we use while loop for loading screen. The while loop evaluates the test expression inside the parentheses (). If test expression is **true**, statements inside the body of while loop are executed. Then, test Expression is evaluated again. The process goes on until test expression is evaluated false. If test expression is **false**, the loop ends.

Next **sleep(1)** is used for pause the output screen for 1 second.

```
struct experience_key{  
  
    char time[30];  
  
    char name[50];  
  
    char place[50];  
  
    char feeling[50];  
  
    char experience[500];  
  
}customer ;
```

Figure-6 (Arrays and structures)

In Figure-6, we use arrays like **char time[30]** and struct datatype **struct experience_key**. Here, **customer** is a variable of **struct experience_key** datatype.

An array is a variable that can store multiple value of same datatype in a single array and structures is used to store multiple values of different datatype (like integer, character together in same variable) in a single struct variable.

```
FILE *fp ;

gets(filename);

fp = fopen (filename, "ab+" ) ;//if file not exists then created append+binary mode

if ( fp == NULL )
{
    fp=fopen(filename,"wb+"); ///if file not exists then created new one.. else overwritten it in binary.
    if(fp==NULL)
    {
        printf("\nSYSTEM ERROR...");

        printf("\nPRESS ANY KEY TO EXIT ");

        int i=0;
        while(i<3){
            printf("%c",178);
            sleep(1);
            i++;
        }

        return ;
    }
}
```

Figure-7 (File Handling: File creating and Overwritten)

In Figure-7, we use file handling to create and overwritten a file. With **FILE *fp** we create a file pointer variable to open a file with **fopen()** command. Here **ab+** and **wb+** are specific mode with specific work process. If file not created in any case, then **fp** will be equal to **NULL**.

```
rewind(fp); // sets the file position to the beginning of the file of the given stream.

while(fread(&customer,sizeof(customer),1,fp)==1)
{
    if(strcmp(customer.time,time)==0)
    {
        printf("\n\tTHE EXPERIENCE ALREADY EXISTS.\n");
        choice=1;
    }
}
```

Figure-8 (File Handling: Linear search)

In Figure-8, we used File handling linear search line by line. With **rewind()** we set the file steam from the beginning and check 1 line each time. If the users search **time is** equal to stored time in inside file which is **customer.time** then we will show the searched item is found at the file. The searching will continue until End of file.

Chapter-3

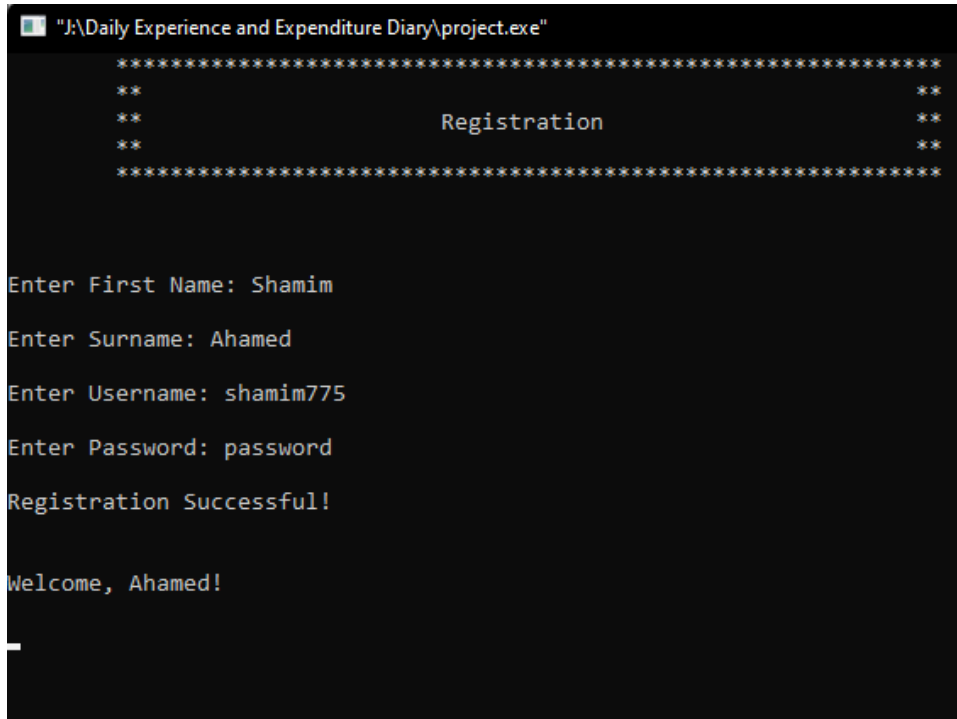
(System testing)

4.1 Introduction

In system testing, the complete system will be tested and the report will be presented.

By system testing, the client will be able to know if his demands have been fulfilled or not. Regardless of how competent software developers and engineers may be, the possibility of glitches and bugs is always present in untested software. The worst thing about bugs and glitches is that it shows software is not secure. Especially when it comes to software that is meant for organizations, errors or loopholes can lead to vulnerability. Testing will lead to better functioning of the product as hidden errors will be exposed and fixed. If the functionality of the software is affected by the change of devices, it can count towards a negative user experience. Testing cuts such errors in the performance while adding to the compatibility and adaptability of the software.

4.2 Input and desired output



```
"J:\Daily Experience and Expenditure Diary\project.exe"

*****
**                                     **
**                               Registration                               **
**                                     **
*****

Enter First Name: Shamim
Enter Surname: Ahamed
Enter Username: shamim775
Enter Password: password
Registration Successful!
Welcome, Ahamed!
```

Figure –9.1 (Registration)

(a) Registration system:

Here we aimed to keep the Registration system as simple as possible. For this reason, we took the user first name, surname, username and user password only. If user did not fill-up the information properly, they would not be able to create account successfully.

```
*****
**                                     **
**                               Login  **
**                                     **
*****

Username:  shamim775
Password: password_
```

Figure –9.2 (Login)

(b)Login System:

If user already registered to the system, they can login directly by supplying their username and password. If user supply wrong password, username or did not register the system before they would not be able to login to the system.

```
*****
**                                     **
**                               ADD Menu                               **
**                                     **
*****

ENTER USERNAME & DATE OF YOUR EXPERIENCE[username yyyy-mm-dd]:shamim775 19Nov2022

ENTER TIME:[hh:mm]:12Am
ENTER NAME OF THE PERSON NEXT TO YOU:None
ENTER PLACE:Dhaka
ENTER YOUR FEELING:Good
EXPERIENCE:Awesome

YOUR EXPERIENCE IS ADDED...

ADD ANOTHER EXPERIENCE...(Y/N)
```

Figure- 9.3 (ADD Menu)

(c) Add menu system:

User can simply add experience if they are logged in to the system. But before they enter experience, they need to enter their user's name and specific date when they want to store the experience. If user supply wrong username and date they would not able to add any experience.

```
*****
**                                **
**          VIEW EXPERIENCE      **
**                                **
*****

ENTER THE USERNAME & DATE OF EXPERIENCE TO BE VIEWED[username yyyy-mm-dd]:shamim775 19Nov2022

HOW WOULD YOU LIKE TO VIEW:

1.WHOLE EXPERIENCE OF THE DAY.
2.EXPERIENCE OF FIX TIME.
ENTER YOUR CHOICE:1

THE WHOLE EXPERIENCE FOR shamim775 19Nov2022 IS:

TIME: 12Am
MEETING WITH: None
MEETING AT: Dhaka
FEELING: Good
EXPERIENCE: Awesome

WOULD YOU LIKE TO CONTINUE VIEWING...(Y/N):
```

Figure- 9.4 (View Experience)

(d)View experience system:

Before view, the earlier experience user needs to enter their username and experience date they want to view. After that they need to enter if they need to watch the entire day's experience or some specific time's experience. If user supply wrong username and date, they cannot view any experience.

```
*****
**                                     **
**                               EDITTING Experience                               **
**                                     **
*****

ENTER THE DATE OF USERNAME & EXPERIENCE TO BE EDITED[username yyyy-mm-dd]:shamim775 19Nov2022

ENTER TIME:[hh:mm]:12Am

YOUR OLD EXPERIENCE WAS AS:
TIME: 12Am
MEETING WITH: None
MEETING AT: Dhaka
FEELING: Good
EXPERIENCE: Awesome

                WHAT WOULD YOU LIKE TO EDIT..
1.TIME.
2.MEETING PERSON.
3.MEETING PLACE.
4.FEELING.
5.experience.
6.WHOLE EXPERIENCE.
7.GO BACK TO MAIN MENU.
    ENTER YOUR CHOICE:
```

Figure – 9.5(Edit Experience)

(e) Edite experience system:

Before edit, the earlier experience user needs to enter their username and experience date they want to view. After that they need to enter if they want to edit whole experience or what specific information they want to edit. If user supply wrong username and date, they cannot edit any experience.


```
*****
**                                **
**          DELETE Experience      **
**                                **
*****

HOW WOULD YOU LIKE TO DELETE.

#DELETE WHOLE EXPERIENCE           [1]
#DELETE A PARTICULAR EXPERIENCE BY TIME [2]
ENTER YOU CHOICE:1

ENTER THE DATE OF EXPERIENCE TO BE DELETED:[username yyyy-mm-dd]:shamim775 19Nov2022
DELETED SUCCESFULLY...
DO YOU LIKE TO DELETE ANOTHER EXPERIENCE.(Y/N):
```

Figure – 9.6(Delete Experience)

(f) Delete Experience:

Before delete, the earlier experience user needs to enter if they want to remove whole experience or what specific information they want to remove. After that they need to enter their username and experience date, they want to continue the process. If user supply wrong username and date, they cannot delete any experience.

```
*****
**                                     **
**                               Contact Menu                               **
**                                     **
*****

=====
[1] Add a new Contact
[2] List all Contacts
[3] Search for contact
[4] Edit a Contact
[5] Delete a Contact
[0] Return Main Menu
=====
Enter the choice:
```

```
...:Search result for 'abid'
=====

...:Name       : abid
...:Phone      : 1235469875
...:Address    : abid@gmail.com
...:Email      : abid@gmail.com

...:Name       : abid
...:Phone      : 0
...:Address    : dhaka
...:Email      : 1@gamail.com

...:Name       : abid
...:Phone      : 2080907882
...:Address    : Chottogram
...:Email      : 254@gmail.com

...:3 match(s) found!
...:Try again?

      [1] Yes      [0] No
_
```

Figure – 9.7 (Contact menu & search contact)

(g) Contact menu & search contact system:

All the input and output process of Contact menu is similar to Add Experience (9.3), View Experience (9.4) & Delete Experience (9.4). But search for contact is an exception. To search a contact user just needs to enter another user name and the system will automatically show all the contacts and their details to the user according to that provided name.

4.3 Report Summary

Here in report summery we will try to present overall summery of the given system. As a result, the owner of the software will get a clear idea of the software by looking at the chart only. Here in this chart, we took name of the test, we use symbols to show the presence or absence of the function in the system. Then we also kept the comment section, in this section there are information about what update could be done in neat future.

| <u>SL NO</u> | <u>NAME</u> | <u>Yes</u> | <u>No</u> | <u>Comment</u> |
|---------------------|--------------------|-------------------|------------------|--|
| 01 | Register | • | | Needs to hide password |
| 02 | Login | • | | Needs to hide password |
| 03 | ADD Experience | • | | |
| 04 | View Experience | • | | |
| 05 | Delete Experience | • | | |
| 06 | Universal Contact | • | | |
| 07 | Add contact | • | | |
| 08 | List all contacts | • | | |
| 09 | Search contact | • | | Search by mobile number or e-mail address could be added |
| 10 | Edit contact | • | | |
| 11 | Delete contact | • | | |
| 12 | Expenditure | | • | Will be available soon |
| 13 | Feedback | | • | Will be available soon |
| 14 | Admin panel | | • | Will be available soon |
| 15 | Customer Care | | • | Will be available soon |

Chapter-5

(Conclusion)

5.1 Good Features

We have designed the software aiming to improve the lifestyle of our users. We are focused to reduce the usage of wooden paper diary and make it digital. We are also focusing on providing each user of our system to make sure they do not get spammed by someone through mobile calling or massaging. To ensure this we provide universal contact. Which will provide all details information of a mobile number such as location-mail address.

5.2 Limitation of the system

So far, the only limitation of this system is when user login or registered to the system their provided password is not hidden. Because of this issue if someone beside user saw user login once they can steal their login id. It can be fatal sometimes.

5.3 Future Enhancement

The project above is just the beginning. Currently our system is only available to windows users only. But we are focusing to develop android, IOS, Linux version of us.

We are already preparing for necessary steps to complete this huge task. We are also focusing to ensure more user privacy and safety in our system so no user feel like their data is being stolen. Increasing our server facility is also another task we are planning to improve so that we can give more fluent experience in rush hours.

Chapter-6

(User Manual)

6.1 Introduction

The User Manual contains all essential information for the user to make full use of the system. This manual includes a description of the system functions and capabilities, contingencies and alternate modes of operation, and systematic procedures for system access and use. We used different screenshot to make things easier for the users. Here the manual has been divided into many categories so that user can easily identify their problems and can easily find the part that he was looking.

(a) Entering the system:

To enter the system at first one needs to make sure his/her device is compatible with the system. The minimum configuration device needed to run this system is:

- Windows 7 or better
- 2GB ram or better
- 5Mb minimum storage required

If all the minimum configuration of device is meet then user can use our system.

(b)After Welcoming page:

In the welcoming page system will take few seconds to load and the show the registered an account or login with an existed account to the system. If a new user registers to the system, they must fill up the registration form with valid information and create login username and password.

```
*****
**                                     **
**                               Login Menu                               **
**                                     **
*****

    Press '1' to Register
    Press '2' to Login

Enter your choice :
```

- After Welcoming windows

```
*****
**                                     **
**                               Registration                               **
**                                     **
*****

Enter First Name: -
Enter Surname: -
Enter Username: -
Enter Password: -
```

- Registration form

If registration is complete new user can login to the system through login option.

```
*****
**                                     **
**                               Login  **
**                                     **
*****

Username:  -
Password:  -
```

- Login windows

Now user can choose the features of system what they need to do (all the necessary information about this is provided on chapter 3).

```
*****
**                                     **
**                               MAIN Menu  **
**                                     **
*****

[1]|  ADD Experience
[2]|  VIEW Experience
[3]|  EDIT Experience
[4]|  DELETE Experience
[5]|  UNIVERSAL Contact
[6]|  EXIT

|  ENTER YOUR CHOICE:
```

- Main menu

#Thanks#