

Payment Application

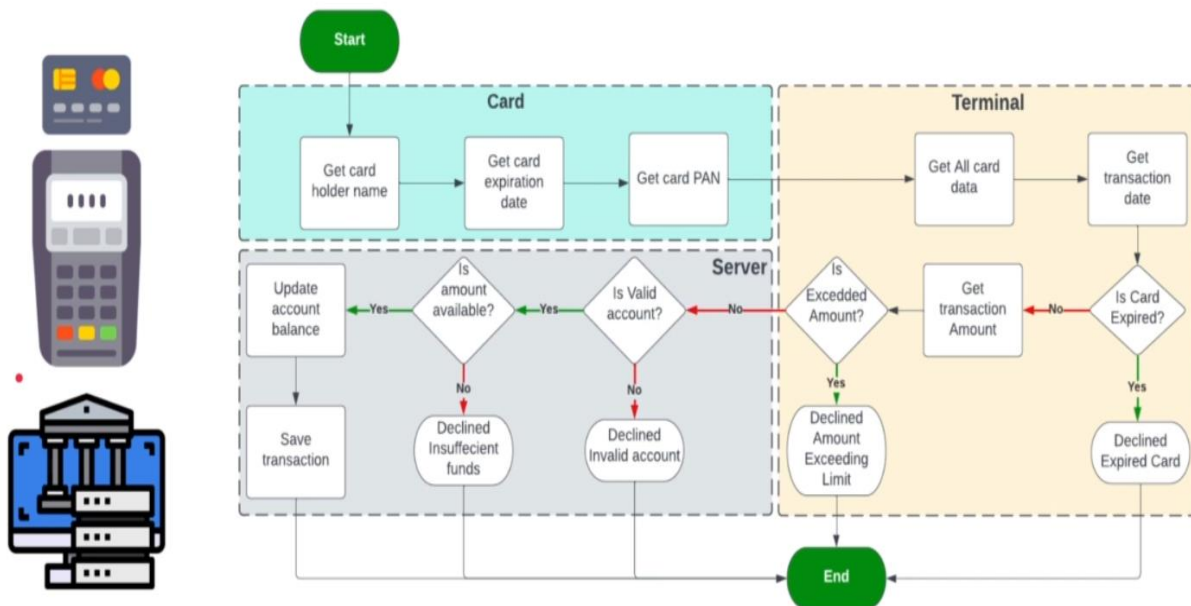
Project 1

By: Ibrahim Mohamed Ibrahim

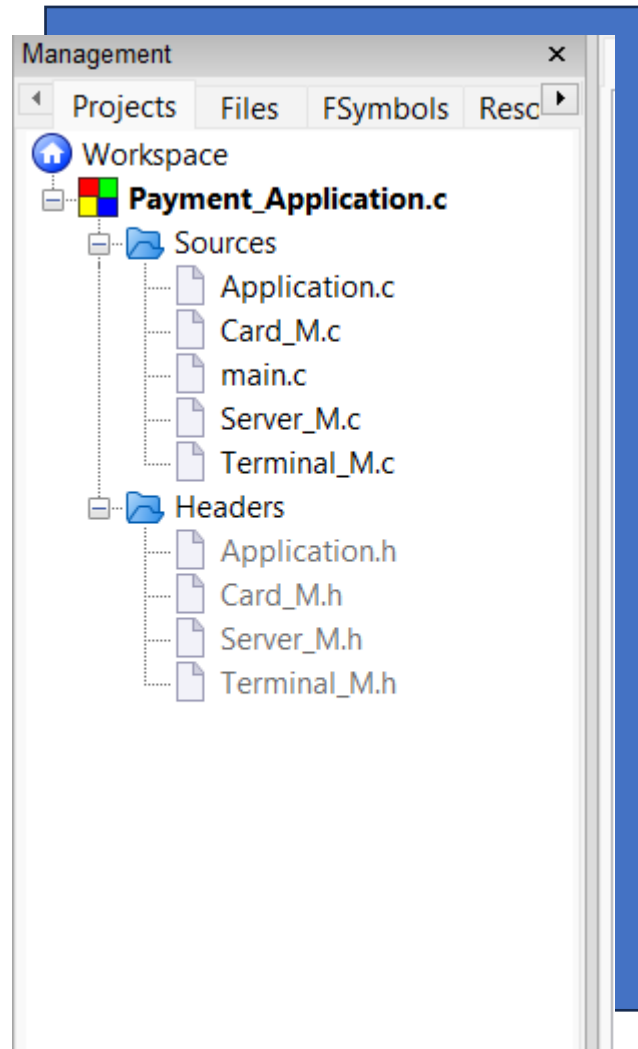
Description: This is a transaction processing application that consists of several modules. These modules are responsible for different aspects of the transaction process, including card data management, terminal interactions, and server operations.

The Card_M module is designed to retrieve card data, such as the cardholder's name, primary account number, and expiration date. It also includes error handling to ensure that invalid inputs are handled appropriately. **The Terminal_M** module is responsible for managing terminal data operations. It obtains the transaction date, checks for card expiration, and validates transaction amounts, ensuring that data is retrieved accurately without any errors.

The Server_M module manages server-side functionalities, such as processing transactions, validating accounts, ensuring fund availability, saving transaction records, and handling various server-related errors.



- All files of Program:



I. Card_M.h file

```
Card_M.h X
D: > Materials > GDSC_Payment_app > Payment_Application.c > Card_M.h
1  #ifndef Card_M_H_
2  #define Card_M_H_
3
4  typedef enum EN_CardError_t
5  {
6      Card_Ok,
7      Wrong_Name,
8      Wrong_Exp_Data,
9      Wrong_Pan
10 }
11 }EN_CardError_t;
12
13 typedef struct ST_CardData_t
14 {
15     char Card_HolderName[25];
16     char PrimaryAccountNumber[20];
17     char CardExpirationDate[6];
18 }
19 }ST_CardData_t;
20
21 ST_CardData_t *CardData;
22
23
24
25 // *****
26 //           APIs
27 // *****
28
29 EN_CardError_t getCardHolderName(ST_CardData_t *CardData);
30 EN_CardError_t getCardExpirationDate(ST_CardData_t *CardData);
31 EN_CardError_t getCardPAN(ST_CardData_t *CardData);
32
33
34 #endif // Card_H_
35
```

II. Card_M.c file

```

D:\> Materials > GDSC_Payment_app > Payment_Application.c > C Card_M.c
1  #include "Card_M.h"
2
3  #include <stdio.h>
4  #include <stdint.h>
5  #include <stdlib.h>
6  #include <string.h>
7
8  extern ST_CardData_t *CardData;
9
10 /*
11  This function prompts the user to input the card holder's name and stores it in structure.
12  It performs validation to ensure that the input meets the specified criteria:
13  >> the name should be between 20 and 24 characters long and consist only of alphabetic characters.
14  If the input is invalid, it returns an error code Wrong_Name, otherwise, it returns Card_Ok.
15  */
16
17 EN_CardError_t getCardHolderName(ST_CardData_t *CardData)
18 {
19     uint8_t Card_HolderName[25] = {0};
20
21     printf("Enter a Card Holder Name: ");
22     gets(Card_HolderName);
23
24
25     uint8_t Num_Characters = strlen(Card_HolderName);
26
27
28     if(24 < Num_Characters || 20 > Num_Characters || Card_HolderName == NULL )
29     {
30         return Wrong_Name;
31     }else
32     {
33         strcpy(CardData->Card_HolderName , Card_HolderName);
34     }
35

```

III. Terminal_M.h file

```
Terminal_M.h X Terminal_M.c ●
D: > Materials > GDSC_Payment_app > Payment_Application.c > C Terminal_M.h
1  #ifndef Terminal_M_H
2  #define Terminal_M_H
3
4  #include <stdint.h>
5  #include "Card_M.h"
6
7
8  typedef struct ST_terminalData_t
9  {
10     float transAmount;
11     float maxTransAmount;
12     uint8_t transactionDate[11];
13 }
14 ST_terminalData_t;
15
16
17 ST_terminalData_t *termData;
18
19 typedef enum EN_terminalError_t
20 {
21     TERMINAL_OK,
22     WRONG_DATE,
23     EXPIRED_CARD,
24     INVALID_CARD,
25     INVALID_AMOUNT,
26     EXCEED_MAX_AMOUNT,
27     INVALID_MAX_AMOUNT,
28 }
29 EN_terminalError_t;
30
31
32
33 // *****
34 //          APIs
35 // *****
```

IV. Terminal_M.c file

```
C Terminal_M.c • C Server_M.h C Server_M.c C Application.h C Application.c C main.c
D: > Materials > GDSC_Payment_app > Payment_Application.c > C Terminal_M.c

1
2 #include "Terminal_M.h"
3 #include "Card_M.h"
4 #include "Server_M.h"
5
6 #include <stdio.h>
7 #include <stdint.h>
8 #include <stdlib.h>
9 #include <string.h>
10 #include <time.h>
11
12 extern ST_terminalData_t *termData;
13 extern ST_transaction_t *transData;
14
15 // This function will ask for the transaction data and store it in terminal data.
16 EN_terminalError_t getTransactionDate(ST_terminalData_t *termData)
17 {
18     uint8_t cardTransactionDate[10] = {0};
19
20     printf("Enter a transaction date (DD/MM/YYYY): ");
21     gets(cardTransactionDate);
22
23     int Num_characters = strlen(cardTransactionDate);
24
25     if (Num_characters != 10 || cardTransactionDate[2] != '/' || cardTransactionDate[5] != '/' || cardTransactionDate == NULL)
26     {
27         return WRONG_DATE;
28     }
29     else
30     {
31         int transMonth, transYear, transDay;
32         sscanf(cardTransactionDate, "%d/%d/%d", &transDay, &transMonth, &transYear);
33         if (transMonth > 12 || transDay > 30)
34         {
35             return WRONG_DATE;
36         }
37         strcpy(termData->transactionDate, cardTransactionDate);
```

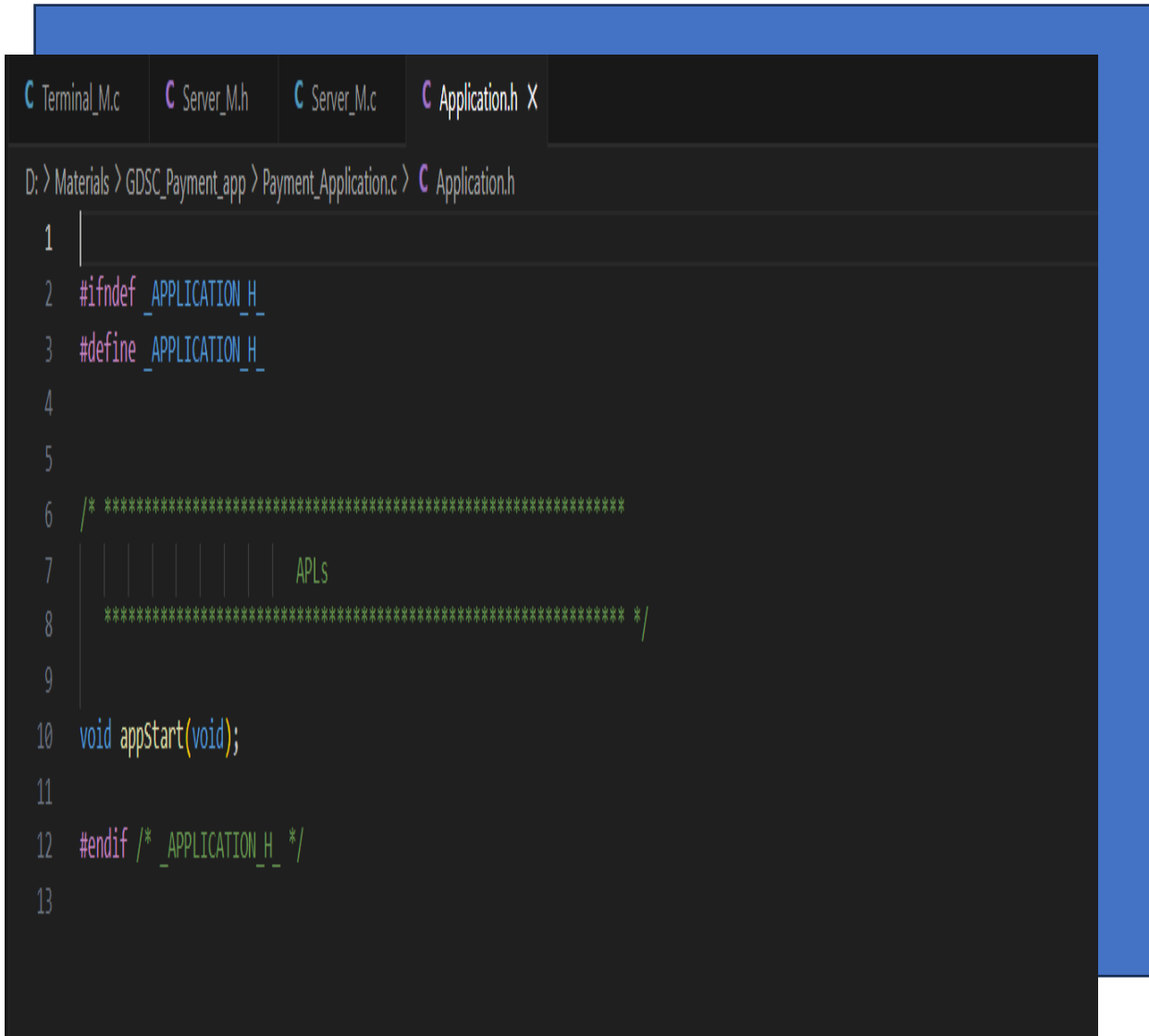
V. Server_M.h file

```
Terminal_M.c  C Server_M.h X
D:\> Materials > GDSC_Payment_app > Payment_Application.c > C Server_M.h
1
2 #ifndef SERVER_M_H_
3 #define SERVER_M_H_
4
5 #include "Card_M.h"
6 #include "Terminal_M.h"
7
8 #include <stdint.h>
9
10 #define MAXACCOUNTS_DB 255
11 #define MAXTRANSACTION_DB 255
12
13
14 typedef enum EN_transState_t
15 {
16     APPROVED,
17     DECLINED_INSUFFICIENT_FUND,
18     DECLINED_STOLEN_CARD,
19     FRAUD_CARD,
20     INTERNAL_SERVER_ERROR,
21
22 }EN_transState_t;
23
24
25 typedef struct ST_transaction_t
26 {
27     ST_CardData_t cardHolderData;
28     ST_terminalData_t terminalData;
29     EN_transState_t transState;
30     uint32_t transactionSequenceNumber;
31
32 }ST_transaction_t;
33
34
35 typedef enum EN_serverError_t
36 {
```


VI. Server_M.c file

```
C Terminal_M.c  C Server_M.h  C Server_M.c X
D:\> Materials \GDSC_Payment_app \Payment_Application.c > C Server_M.c
1
2 #include "Server_M.h"
3 #include "Card_M.h"
4 #include "Terminal_M.h"
5
6 #include <string.h>
7
8 ST_transaction_t Transactions_dataBase[MAXTRANSACTION_DB] = {0};
9
10
11 ST_accountsDB_t Accounts_dataBase[MAXACCOUNTS_DB] =
12 {
13     {5100.5 , RUNNING , "5078031014041793"},
14     {10000.1 , BLOCKED , "5078013579246801"},
15     {62000.2 , RUNNING , "5078543210987654"},
16     {7500.4 , RUNNING , "5078654345098765"},
17     {9150.7 , BLOCKED , "5078346802468137"},
18     {3555.6 , RUNNING , "5078376514436098"}
19 };
20
21
22 uint16_t indexOfCurrentAccount = 0;
23
24
25 EN_transState_t receiveTransactionData(ST_transaction_t *transData)
26 {
27     EN_transState_t ErrorState = APPROVED;
28
29     EN_serverError_t accountError = isValidAccount(&(transData->cardHolderData), Accounts_dataBase);
30     if (accountError == ACCOUNT_NOT_FOUND) {
31         ErrorState = FRAUD_CARD;
32     } else if (accountError == BLOCKED_ACCOUNT) {
33         ErrorState = DECLINED_STOLEN_CARD;
34     }
35
36     if (ErrorState == APPROVED)
```

VII. Application.h file



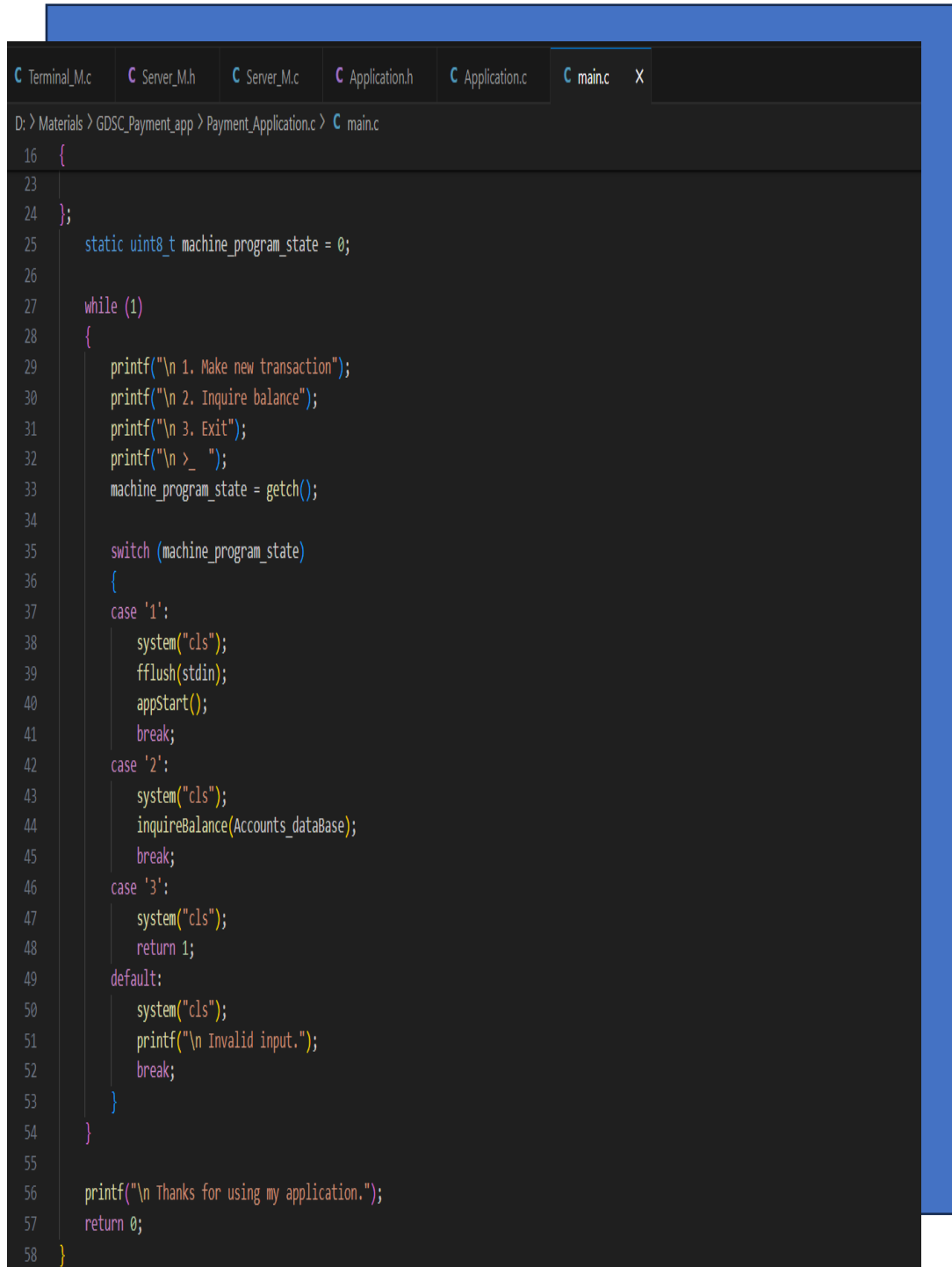
The image shows a code editor window with a dark background and a blue title bar. The editor has four tabs at the top: 'Terminal_M.c', 'Server_M.h', 'Server_M.c', and 'Application.h X'. The 'Application.h' tab is active. The file path in the editor is 'D: > Materials > GDSC_Payment_app > Payment_Application.c > Application.h'. The code in the file is as follows:

```
1 |
2 | #ifndef _APPLICATION_H
3 | #define _APPLICATION_H
4 |
5 |
6 | /* *****
7 | | | | | | | APLs
8 | ***** */
9 |
10 | void appStart(void);
11 |
12 | #endif /* _APPLICATION_H */
13 |
```

VIII. Application.c file

```
C Terminal_M.c  C Server_M.h  C Server_M.c  C Application.h  C Application.c X
D:\> Materials > GDSC_Payment_app > Payment_Application.c > C Application.c
1
2  #include <stdio.h>
3  #include <stdint.h>
4
5  #include "Card_M.h"
6  #include "Terminal_M.h"
7  #include "Server_M.h"
8  #include "Application.h"
9
10
11 void appStart(void)
12 {
13     ST_CardData_t cardData;
14     ST_terminalData_t termData;
15     EN_terminalError_t terminalError;
16     EN_serverError_t serverError;
17     ST_transaction_t *transData;
18     EN_transState_t transError;
19
20     ST_accountsDB_t Accounts_dataBase[MAXACCOUNTS_DB] =
21     {
22         {5100.5 , RUNNING , "5078031014041793"},
23         {10000.1 , BLOCKED , "5078013579246801"},
24         {62000.2 , RUNNING , "5078543210987654"},
25         {7500.4 , RUNNING , "5078654345098765"},
26         {9150.7 , BLOCKED , "5078346802468137"},
27         {3555.6 , RUNNING , "5078376514436098"}
28     };
29
30     int indexOfCurrentAccount = -1;
31     float remainingBalance;
32     /* *****
33     | | | | | Test Card Validation
34     ***** */
35
```

IX. Main.c file



```
16 {
23
24 };
25 static uint8_t machine_program_state = 0;
26
27 while (1)
28 {
29     printf("\n 1. Make new transaction");
30     printf("\n 2. Inquire balance");
31     printf("\n 3. Exit");
32     printf("\n >_ ");
33     machine_program_state = getch();
34
35     switch (machine_program_state)
36     {
37     case '1':
38         system("cls");
39         fflush(stdin);
40         appStart();
41         break;
42     case '2':
43         system("cls");
44         inquireBalance(Accounts_dataBase);
45         break;
46     case '3':
47         system("cls");
48         return 1;
49     default:
50         system("cls");
51         printf("\n Invalid input.");
52         break;
53     }
54 }
55
56 printf("\n Thanks for using my application.");
57 return 0;
58 }
```

Tests Cases:

```
D:\Materials\GDSC_Payment_ X + v
1. Make new transaction
2. Inquire balance
3. Exit
>_
```

```
D:\Materials\GDSC_Payment_ X + v
Enter a Primary Account Number of Card: 5078543210987654
Enter a Card Holder Name: ibrahim mohamed ibrahim
Enter a Card Expiration Date(MM/YY): 10/24
Enter a transaction date (DD/MM/YYYY): 05/09/2024
Enter transaction max amount:$10000.00
Enter Your amount transaction:$4500.00

==> Successfully Transaction <==

Transaction done by PAN card : 5078543210987654
Remaining balance          : $57500.20

*** Thanks for using my application ***
-----

1. Make new transaction
2. Inquire balance
3. Exit
>_ |
```

D:\Materials\GDSC_Payment_ X + v
Enter a Primary Account Number of Card: 5078654345098760

Error!! Unable to find PAN card in the database.

1. Make new transaction
2. Inquire balance
3. Exit
>_ |

The correct PAN in DB
5078654345098765

D:\Materials\GDSC_Payment_ X + v
Enter a Primary Account Number of Card: 5078654345098765

Enter a Card Holder Name: ibrahim mohamed

Error!! Invalid Card Holder Name

1. Make new transaction
2. Inquire balance
3. Exit
>_ |

Can't meet the
requirements of card
holder name

D:\Materials\GDSC_Payment_ X + v
Enter a Primary Account Number of Card: 5078654345098765

Enter a Card Holder Name: ibrahim mohamed ibrahim

Enter a Card Expiration Date(MM/YY): 15/24

Error!! Invalid Card Expiration Date

1. Make new transaction
 2. Inquire balance
 3. Exit
- >_ |

The month exceed 12
and this not valid.

D:\Materials\GDSC_Payment_ X + v
Enter a Primary Account Number of Card: 5078654345098765

Enter a Card Holder Name: ibrahim mohamed ibrahim

Enter a Card Expiration Date(MM/YY): 10/24

Enter a transaction date (DD/MM/YYYY): 35/13/2024

Error!! Invalid Transaction Date

1. Make new transaction
 2. Inquire balance
 3. Exit
- >_ |

The day exceed 30 and month
exceed 12 so give error message.

```
D:\Materials\GDSC_Payment_ X + v
Enter a Primary Account Number of Card: 5078654345098765
Enter a Card Holder Name: ibrahim mohamed ibrahim
Enter a Card Expiration Date(MM/YY): 10/24
Enter a transaction date (DD/MM/YYYY): 08/11/2024
```

Error!! Card is expired.

```
1. Make new transaction
2. Inquire balance
3. Exit
>_ |
```

Error because the try to make transaction after expiration date of PAN card.

```
D:\Materials\GDSC_Payment_ X + v
Enter a Primary Account Number of Card: 5078654345098765
Enter a Card Holder Name: mohsen mohamed mahmoud
Enter a Card Expiration Date(MM/YY): 03/16
Enter a transaction date (DD/MM/YYYY): 01/11/2015
Enter transaction max amount:$10000.00
Enter Your amount transaction:$9500.00
```

Error!! NOT Enough balance

```
1. Make new transaction
2. Inquire balance
3. Exit
>_ |
```

This PAN card its balance in DB is 7500.4 and User try to withdraw 9500.0 and this exceed balance.