1. **Analysis & Design**
   1. Current System
   2. New System
   3. Program Specification
      1. Screen Design
      2. Input Specification
      3. File Specification
      4. CRT Forms
      5. Processing and Validation
   4. Program Design - PSEUDOCODE
2. **Development & Testing**
   1. Program Listing
   2. Testing
      1. Test Plan
      2. Test Cases
      3. Test Log
3. **Implementation** 
   1. User Manual
   2. Conclusion
      1. Strengths
      2. Weaknesses
      3. Enhancements

**Chapter 1: Analysis and Design**

**CURRENT SYSTEM**

The Manual Ticketing Systems that they are using is that they have to buy tickets in bundles of one’s, two’s, five’s and ten’s which is equivalent to Philippine currency. The conductor should memorize the fare matrix in order for him to know the fare of each passenger depending on how far their destination is. Then conductor must make sure that each passenger gets a ticket and it’s paid for. With the continuous coming in of passengers, there are times that conductors usually get confused of the tickets they give to the passengers. That’s why bus liners hired inspectors to check whether the tickets that the conductors give are not recycled and is accurate based on the fare matrix.

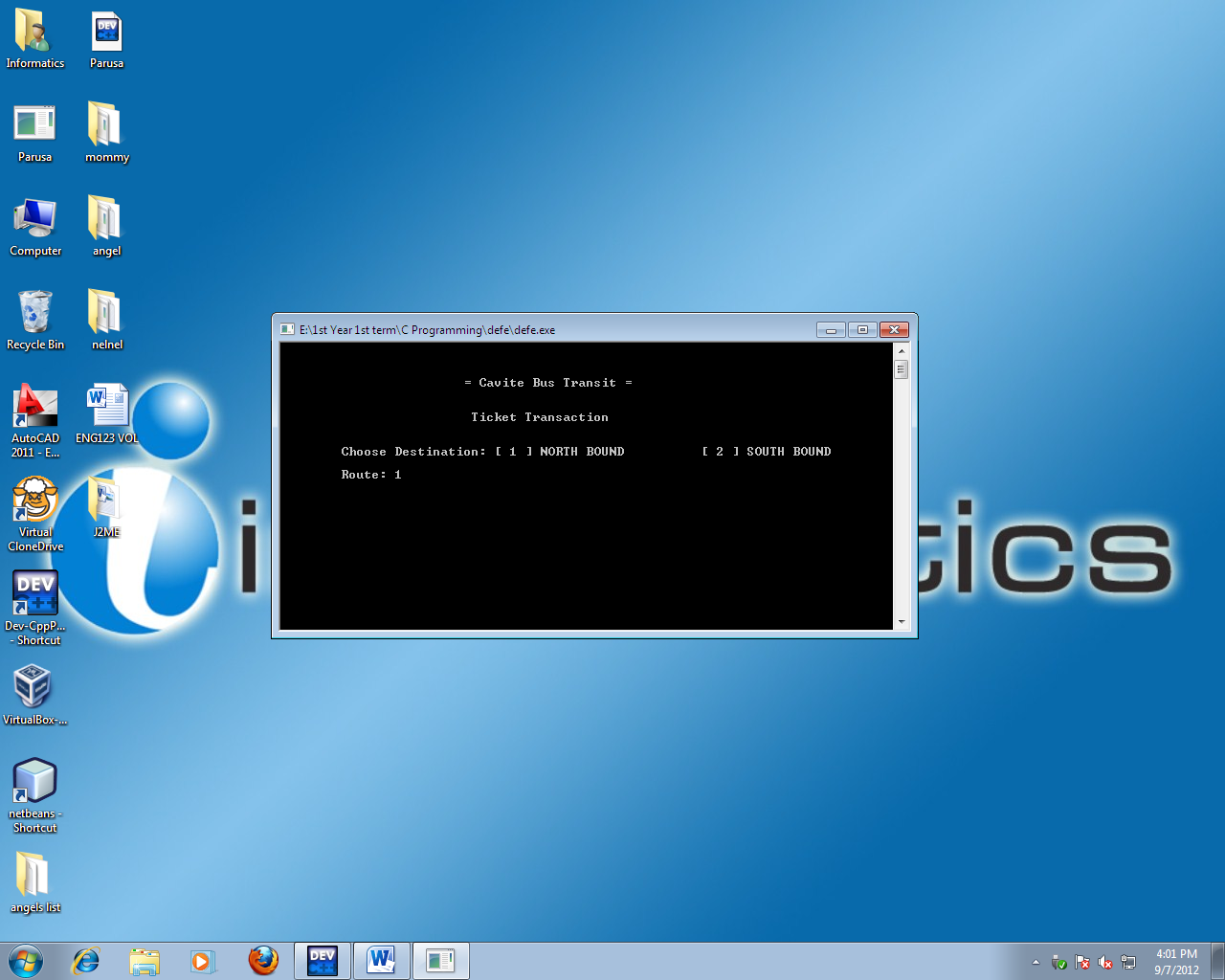
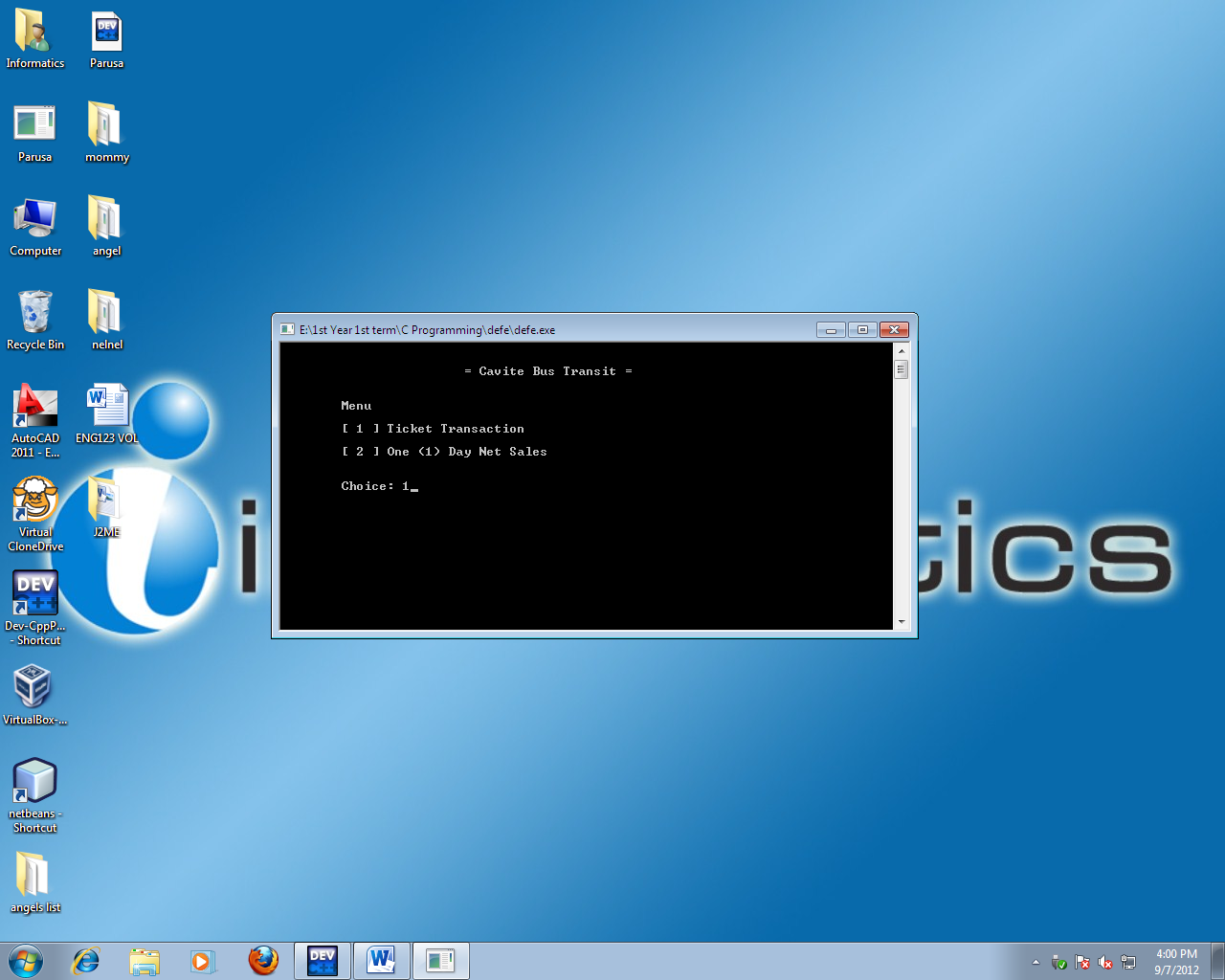
**PROPOSED SYSTEM**

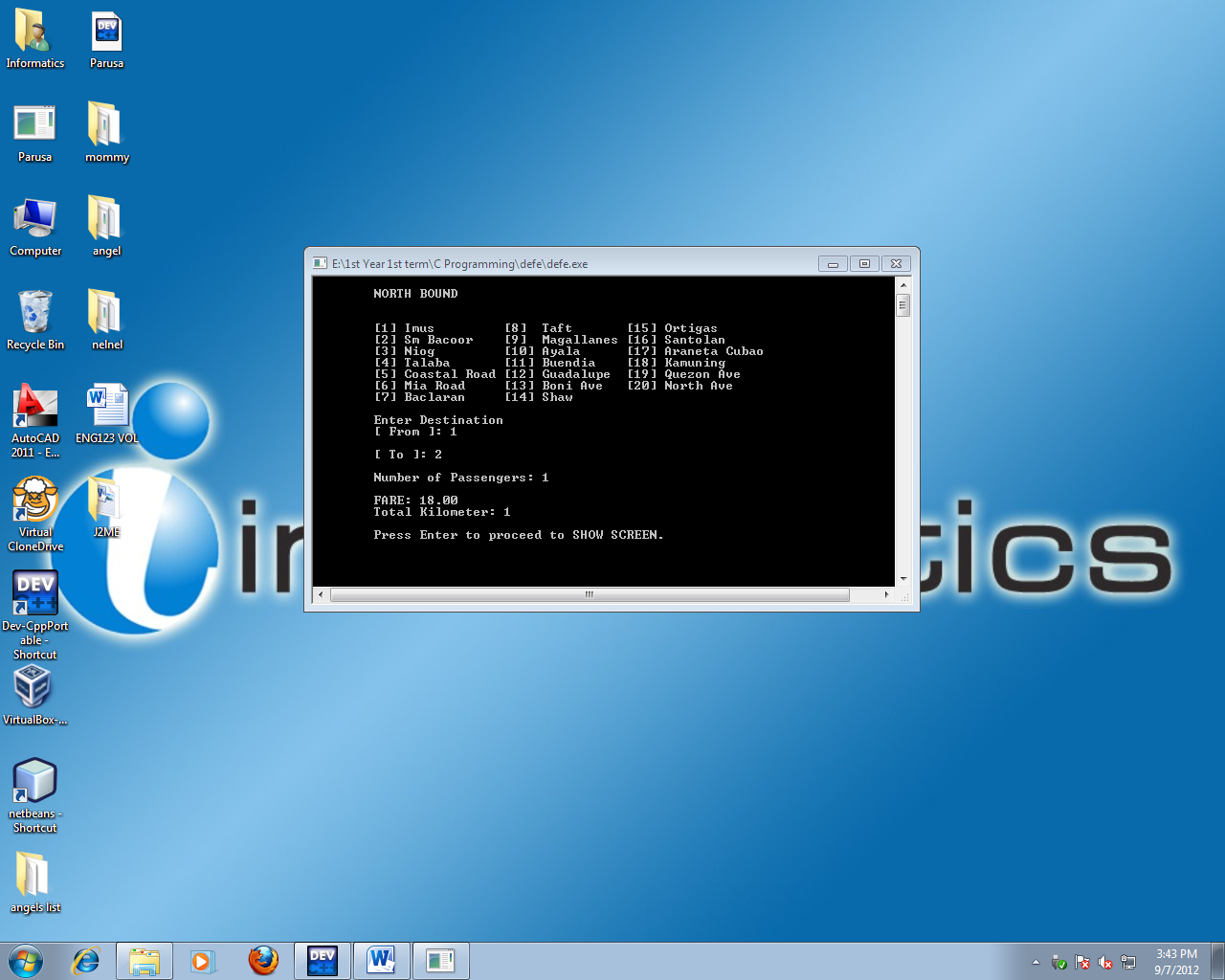
The population in Metro Manila is constantly increasing. Along with this increase comes additional traffic and congestion on surrounding roadways. There are numerous amounts of commuting students and employees are in rush, workers in the city and surrounding areas, and added construction and archeological projects. At peak points during the day many buses are greatly affected by traffics, and some buses lose their profits because of the bus ticket system that they are using.

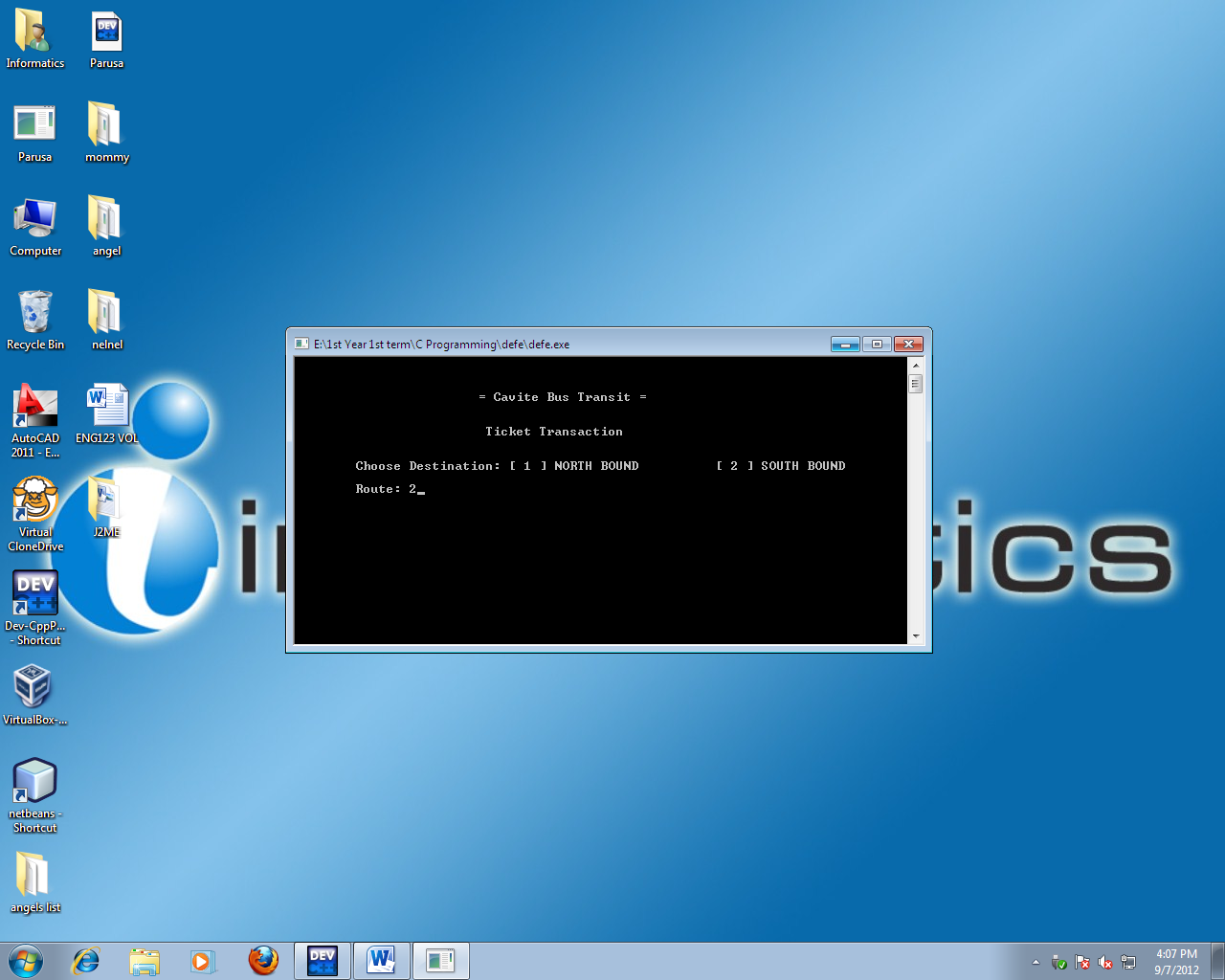
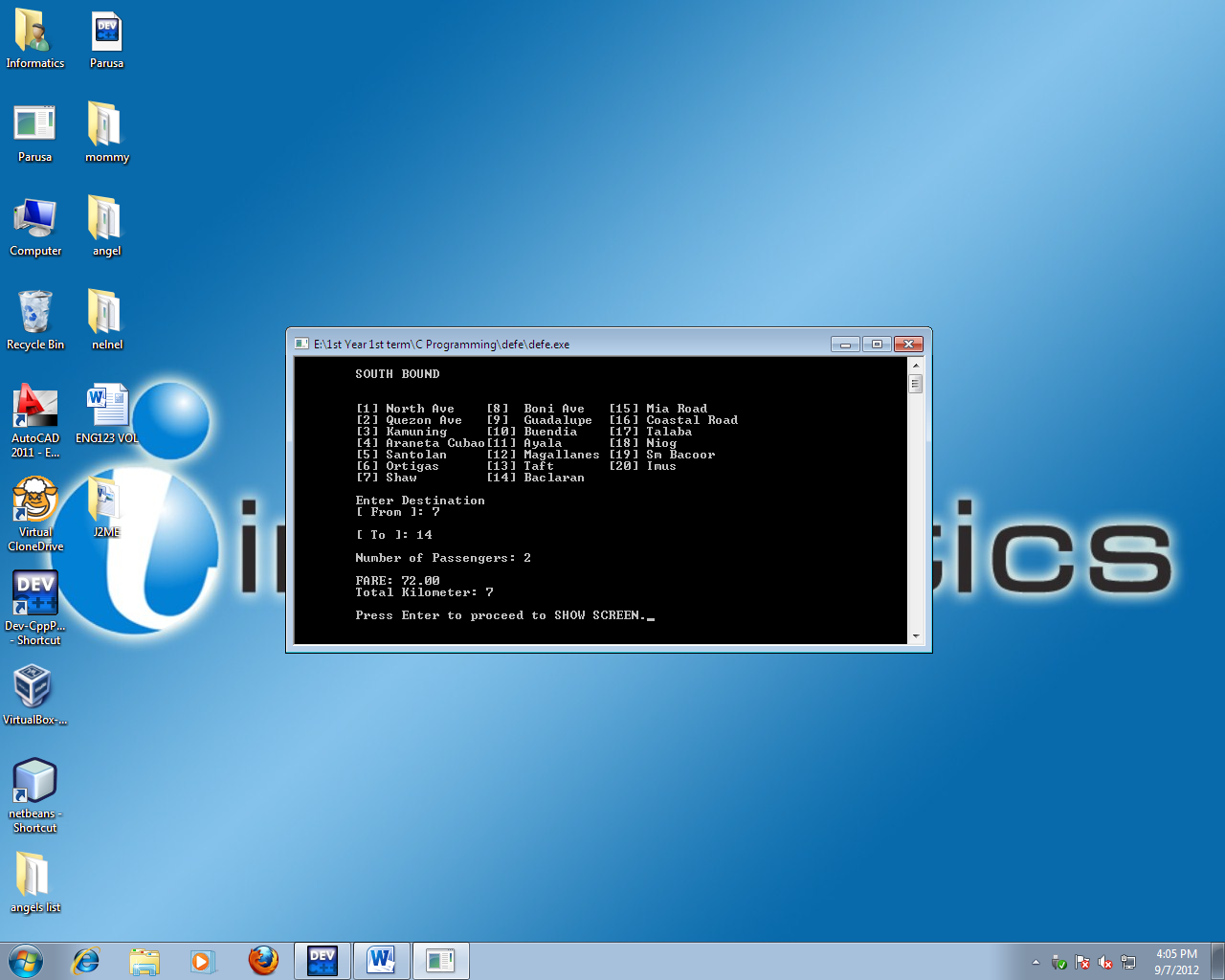
For example, the Casper bus liner, they use manual ticketing system which can easily confused the conductors and most passengers take advantage of it by cheating. Because of this, bus liners lose profit. And because of that, I recommend that they use Automated Ticketing System that would accurately calculate their profit and make the conductor’s job much easier.

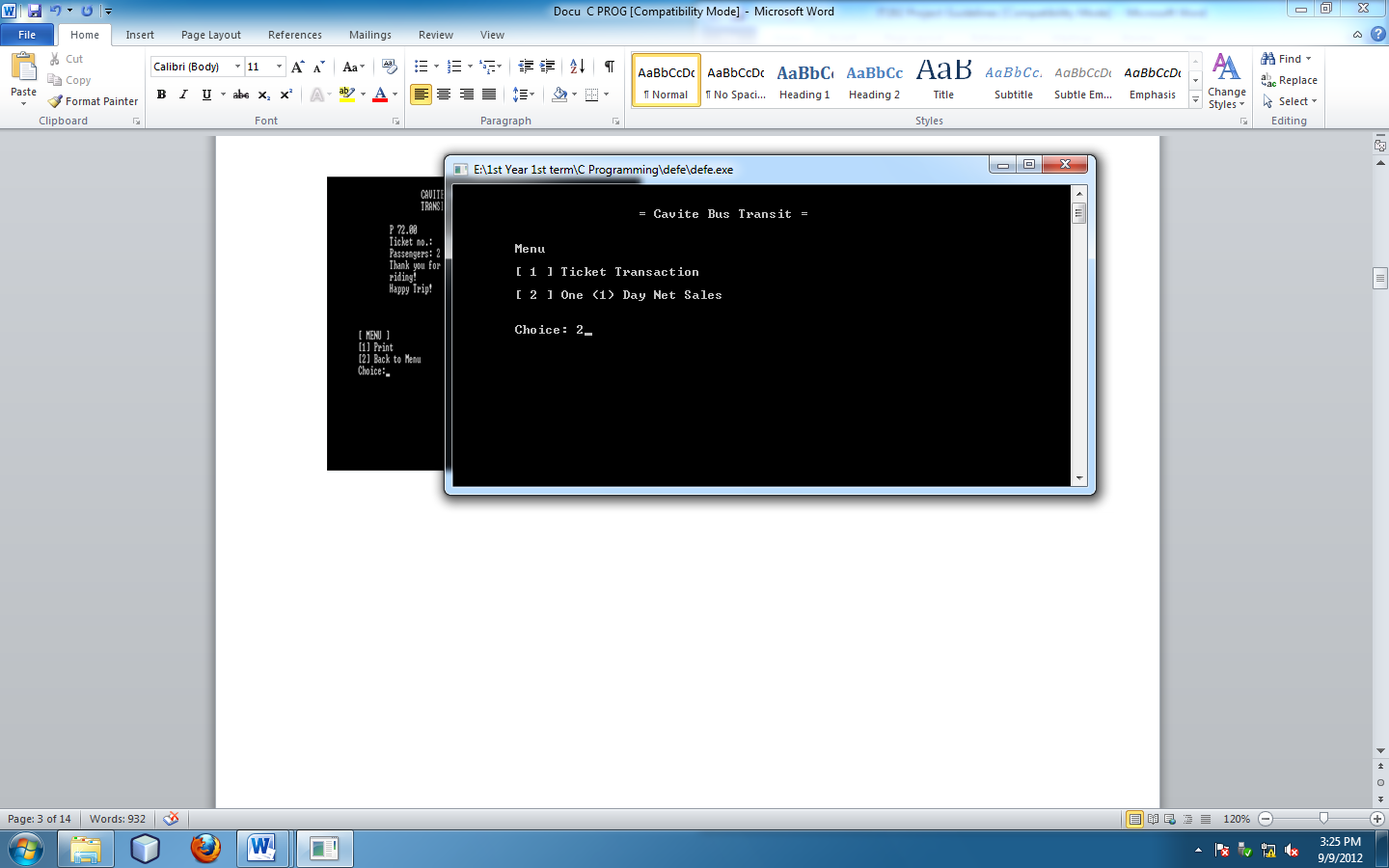
**PROGRAM SPECIFICATION**

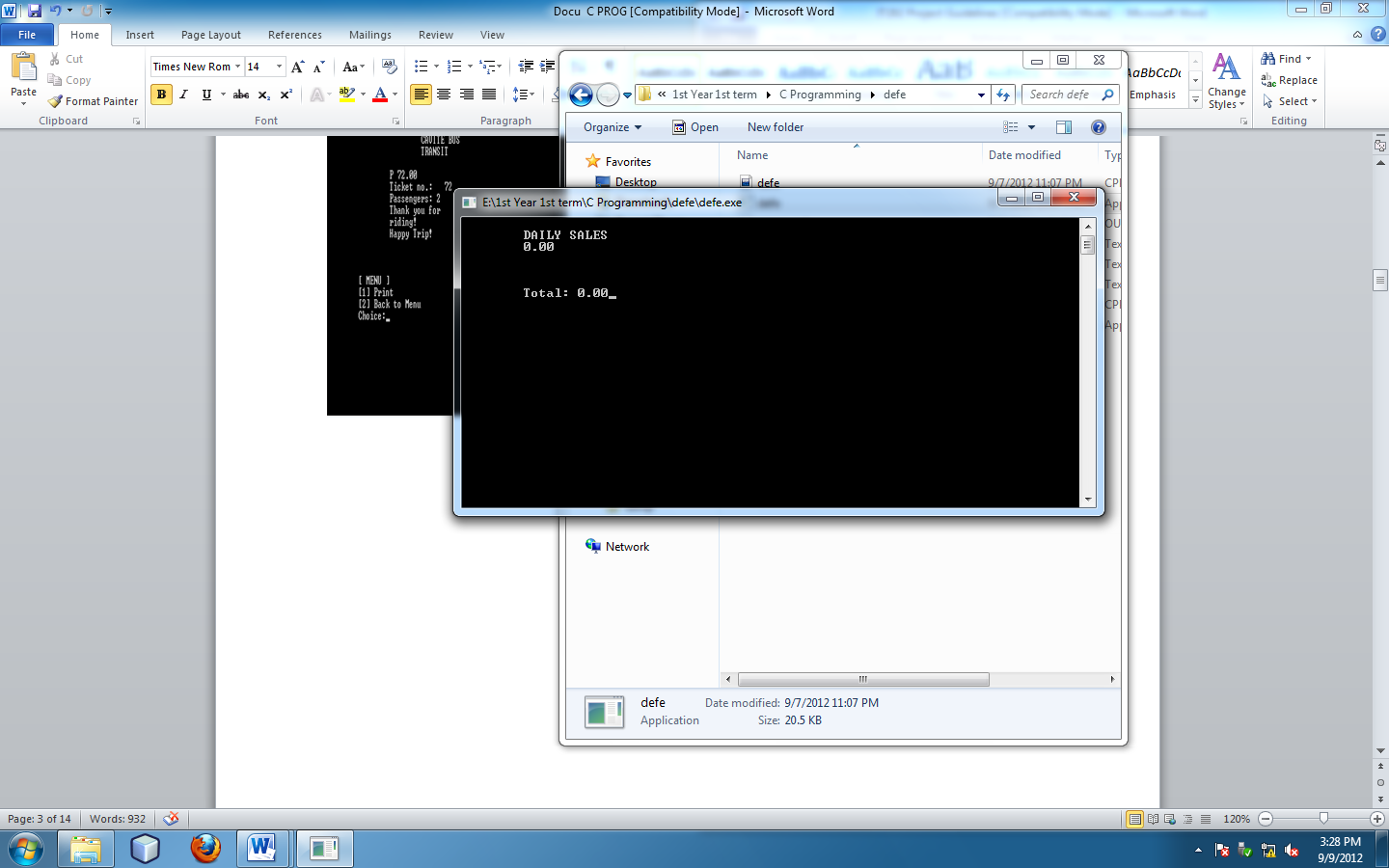
**SCREEN DESIGN**



****



****



**INPUT SPECIFICATION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **Range** |
| x | Input choice in Menu | Int | - | 1-3 |
| to | Input choice in to | Int | - | 1-19 |
| fr | Input choice in from | Int | - | 2-20 |
| pass | Input number of passenger/s | Int | - | 1-60 |
| choice | Input number for North or Sound Bound | Int | - | 1-2 |

**FILE SPECIFICATION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **Range** |
| kmeter | Total Kilometer | Int | - | 1-20 |
| i | Iteration | Ctr | - | - |
| tick | Ticket Number | Int | - | 1-9999 |
| fare | Total of kilometer with passengers | Float | - | 1-9999 |
| t1/t2 | Used for computation | Float | - | 1-9999 |

**CRT FORMS**

=Cavite Bus Transit=

Ticket Transaction

Enter Destination: [1] North route [2] South route

Route:1

=Cavite Bus Transit=

Menu

[1] Ticket Transaction:

[2] One (1) day net sales:

Choice: 1

Cavite Bus

Transit

P18.00

Ticket no. 18

Passenger: 1

Thank you for riding!

Menu

[1]Print

[2]Back to Main menu

Choice:1

NORTH BOUND

[1]Imus [8]Taft [15]Ortigas

[2]SmBacoor [9]Magallanes[16]Santolan

[3]Niog [10]Ayala [17] AranetaCubao

[4]Talaba [11]Buendia [18]Kamuning

[5]Coastal Rd[12]Guada [19]Quezon Ave.

[6]Mia Rd [13]Boni Ave. [20] North Ave.

[7]Baclaran [14]Shaw Blvd.

Enter Destination:

To: 1

From: 2

Number of passengers: 1

Fare: 18.00

Total Km: 1

Proceed to the show screen…

Daily Sales

18.00

Total Sales: 18.00

=Cavite Bus Transit=

Menu

[1] Ticket Transaction:

[2] One (1) day net sales:

Choice: 2

Daily Sales

18.00

18.00

Total Sales: 36.00

=Cavite Bus Transit=

Menu

[1] Ticket Transaction:

[2] One (1) day net sales:

Choice: 2

Cavite Bus

Transit

P18.00

Ticket no. 18

Passenger: 1

Thank you for riding!

Menu

[1]Print

[2]Back to Main menu

Choice:1

SOUTH BOUND

[1]North Ave [8]Boni [15]MiaRd

[2]Quezon Ave. [9]Guada [16]CoastalRd

[3]Kamuning [10]Buendia [17] Talaba

[4]AranetaCubao [11]Ayala [18]Niog

[5]Santolan [12]Magallanes [19]SmBacoor

[6]Ortigas [13]Taft [20] Imus

[7]Shaw [14]Baclaran

Enter Destination:

To: 1

From: 2

Number of passengers: 1

Fare: 18.00

Total Km: 1

Proceed to the show screen…

=Cavite Bus Transit=

Ticket Transaction

Enter Destination: [1] North route [2] South route

Route:2

=Cavite Bus Transit=

Menu

[1] Ticket Transaction:

[2] One (1) day net sales:

Choice: 1

**PROCESSING AND VALIDATION**

**FORMAT CHECK**

Ticket Transaction: Shows you the route transaction. It can only input number.

One day net sales: Shows you the total net sales made by the end of the day.

**SIZE CHECK**

Normally, it can only input numbers.

**RANGE CHECK**

North and South Bound Route: Basically, you can enter a specific range of numbers “**1-20**”.

Number of Passengers: For a normal bus, you can input a maximum of **60** passengers.

**SUGGESTED CALCULATION**

Kilometers: kmeter = ( to-fr )

Ticket Fare: fare = ( ( ( kmeter \* 3.00 ) + 15.00 ) \* pass );

**PROGRAM DESIGN**

**PSEUDOCODE**

**DO**

Display “ Menu “

Display “[1]Ticket Transaction”

Display “[2]One day net sales”

Display “Choice:”

Input choice

**IF**

choice is = 1 go to ticket transaction

**ELSE IF**

choice is = 2 go to sales

**ELSE**

choice is greater than 2 print “INVALID CHOICE!”

**END IF**

**END DO**

**DO**

Print “Ticket Transaction”

Print “[1]North Bound”

Print “[2]South Bound”

Print “Choice”

**IF**

choice is = 1 go to North Bound route

**ELSE IF**

choice is = 2 go to South Bound route

**ELSE**

choice is greater than 2 print “INVALID CHOICE!”

**END IF**

**END DO**

**DO**

Print “North Bound”

Print “[1] Imus [8] Taft [15] Ortigas”

Print “[2] Sm Bacoor [9] Magallanes [16] Santolan”

Print “[3] Niog [10] Ayala [17] Araneta Cubao”

Print “[4] Talaba [11] Buendia [18] Kamuning”

Print “[5] Coastal Road [12] Guadalupe [19] Quezon Ave”

Print “[6] Mia Road [13] Boni Ave [20] North Ave”

Print “[7] Baclaran [14] Shaw”

Enter destination

Enter “from” **THEN**

Enter “to”

Get total number of passengers

**IF**

To is greater than from proceed to computation

**ELSE**

From is greater than to proceed to computation

**END IF**

**END DO**

**DO**

Print “SOUTH BOUND”

Print “[1] North Ave [8] Boni Ave [15] Mia Road”

Print “[2] Quezon Ave [9] Guadalupe [16] Coastal Road”

Print “[3] Kamuning [10] Buendia [17] Talaba”

Print “[4] Araneta Cubao[11] Ayala [18] Niog”

Print “[5] Santolan [12] Magallanes [19] Sm Bacoor”

Print “[6] Ortigas [13] Taft [20] Imus”

Print “ [7] Shaw [14] Baclaran”

Enter destination

Enter “from” **THEN**

Enter “to”

Get total number of passengers

**IF**

To is greater than from proceed to computation

**ELSE**

From is greater than to proceed to computation

**END IF**

**END DO**

**DO**

Display “total fare”

Display “ticket number”

Display “number of passengers”

Print “choice 1 for print ticket”

Print “choice 2 for back to main menu”

**IF**

Choice 1 is equal to 1 go to add details to database

**ELSE IF**

Choice 2 is equal to 2 go to main menu

**ELSE**

Choice is greater than or equal to 3 print “INVALID CHOICE!”

**END IF**

**END DO**

**CHAPTER 2: DEVELOPMENT AND TESTING**

**PROGRAM LISTING**

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#include<string.h>

#define p printf

#define s scanf

// FUNCTIONS

voidticketdest();

void destination();

void dbase();

void comp();

void menu();

void ticket();

void north();

void south();

void show();

void print();

void start();

voiddsplysale();

// VARIABLES

int i, x, y, to, fr, pass, tick, choice, kmeter;

float t1,t2;

int place[20];

float fare, total,total2;

main()

{

start();

getch();

}

void start()

{

p("\n\n\t\t\t= Cavite Bus Transit =");

p("\n\n\n\tMenu");

p("\n\n\t[ 1 ] Ticket Transaction");

p("\n\n\t[ 2 ] One (1) Day Net Sales");

p("\n\n\n\tChoice: ");

s("%d", &x);

if (x == 1) menu();

else if (x == 2) dsplysale();

else

{ p("\n\n\n\tINVALID CHOICE! Press any key to continue... "); getch(); system ("cls"); start();

getch();}

}

voiddsplysale()

{ system("cls");

p("\n\tDAILY SALES");

comp();

getch(); system("cls"); start(); }

void menu()// NORTH AND SOUTH BOUND ROUTE

{

system("cls");

p("\n\n\n\t\t\t= Cavite Bus Transit =\n\n");

p("\n\t\t\t Ticket Transaction\n ");

p("\n\n\tChoose Destination: ");

p("[ 1 ] NORTH BOUND [ 2 ] SOUTH BOUND ");

p("\n\n\tRoute: ");

s("%d", &y);

if (y == 1) north();

else if ( y == 2) south();

else {p("INVALID CHOICE!"); getch(); system("cls"); menu(); getch(); }

}

voidticketdest()

{

p("\n\tEnter Destination");

p("\n\t[ From ]: ");

s("%d",&fr);

p("\n\t[ To ]: ");

s("%d",&to);

p("\n\tNumber of Passengers: ");

s("%d",&pass);

if(to >fr) { destination();

p("\n\tFARE: %2.2f",fare);

p("\n\tTotal Kilometer: %d\n", kmeter);

p("\n\tPress Enter to proceed to SHOW SCREEN."); getch(); show();

}

if(to <fr)

{ system("cls"); destination(); south(); ticketdest(); getch(); }

}

// NORTH ROUTE

void north()

{

system("cls");

p("\n\tNORTH BOUND\n\n");

p("\n\t[1] Imus [8] Taft [15] Ortigas");

p("\n\t[2] SmBacoor [9] Magallanes [16] Santolan");

p("\n\t[3] Niog [10] Ayala [17] AranetaCubao");

p("\n\t[4] Talaba [11] Buendia [18] Kamuning");

p("\n\t[5] Coastal Road [12] Guadalupe [19] Quezon Ave");

p("\n\t[6] Mia Road [13] Boni Ave [20] North Ave");

p("\n\t[7] Baclaran [14] Shaw\n");

ticketdest();

}

// SOUTH ROUTE

void south()

{

system("cls");

p("\n\tSOUTH BOUND\n\n");

p("\n\t[1] North Ave [8] Boni Ave [15] Mia Road");

p("\n\t[2] Quezon Ave [9] Guadalupe [16] Coastal Road");

p("\n\t[3] Kamuning [10] Buendia [17] Talaba");

p("\n\t[4] AranetaCubao[11] Ayala [18] Niog");

p("\n\t[5] Santolan [12] Magallanes [19] SmBacoor");

p("\n\t[6] Ortigas [13] Taft [20] Imus");

p("\n\t[7] Shaw [14] Baclaran\n");

ticketdest();

}

void show()

{

system("cls");

tick = tick+1;

for(i=0;i<22;i++);

p("\n\t\t\tCAVITE BUS\n\t\t\tTRANSIT\n");

p("\n\t\tP %.2f",fare);

p("\n\t\tTicket no.: %4.0f",tick);

p("\n\t\tPassengers: %d",pass);

p("\n\t\tThank you for ");

p("\n\t\triding!");

p("\n\t\tHappy Trip!");

dbase();

p("\n\n\n\n\t[ MENU ]");

p("\n\t[1] Print");

p("\n\t[2] Back to Menu");

p("\n\tChoice:");

s("%d",&choice);

if(choice == 1) { print(); }

else if(choice == 2) { system("cls");start(); }

else (choice >=3 ); { p("\n\tINVALID CHOICE!"); getch(); system("cls"); show(); getch(); }

}

void print()

{ system("cls");start(); }

void destination()

{

kmeter=(to-fr);

if(kmeter<=3)

{

fare = 15.00;

}

else if(kmeter>=4)

{

fare = ((kmeter\*3.00)+15.00);

}

else (pass>=2);

{

fare = (((kmeter\*3.00)+15.00)\*pass);

}

}

void dbase()

{

FILE \*dbase;

dbase=fopen("kinita.txt","a+");

fprintf(dbase,"%.2f\n\n",fare);

fclose(dbase);

}

void comp()

{

FILE \*dbaseko;

dbaseko=fopen("kinita.txt","a+");

while(!feof(dbaseko))

{

fscanf(dbaseko,"%f",&total);

printf("\n\t%.2f\n\n",total);

total2=total2+total;

}

p("\n\n\tTotal: %.2f",total2);

fclose(dbaseko);

}

**TESTING**

**TEST PLAN**

|  |  |
| --- | --- |
| TEST CASES | OBJECTIVES |
| Case 1 | To check for the correct entry range of the options in the main menu. |
| Case 2 | To check for the correct entry of the options in the Ticket Transaction |
| Case 3 | To show the Destination if you press 1 or 2 |
| Case 4 | To show the fare total and the total kilometers after picking the destination |
| Case 5 | To check that the number of characters entered for the passport number does not exceed 15. |

**Testing – Test cases**

##### Test case : 1

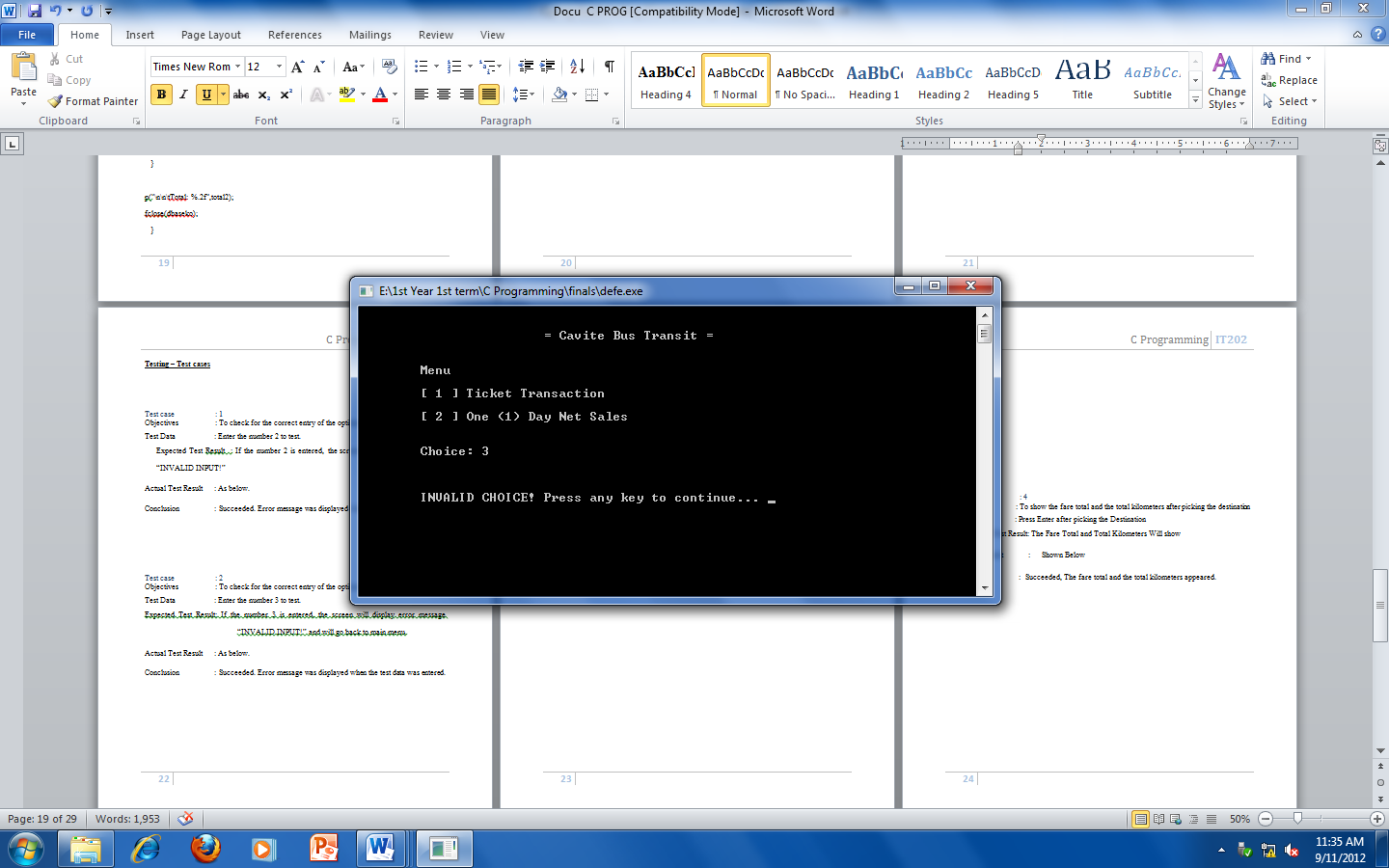
Objectives : To check for the correct entry of the options in the main menu.

Test Data : Enter the number 3 to test.

Expected Test Result : If the number 3 is entered, the screen will display error message “INVALID CHOICE! Press any key to continue…”

Actual Test Result : As below.

Conclusion : Succeeded. Error message was displayed when the test data was entered.



##### Test case : 2

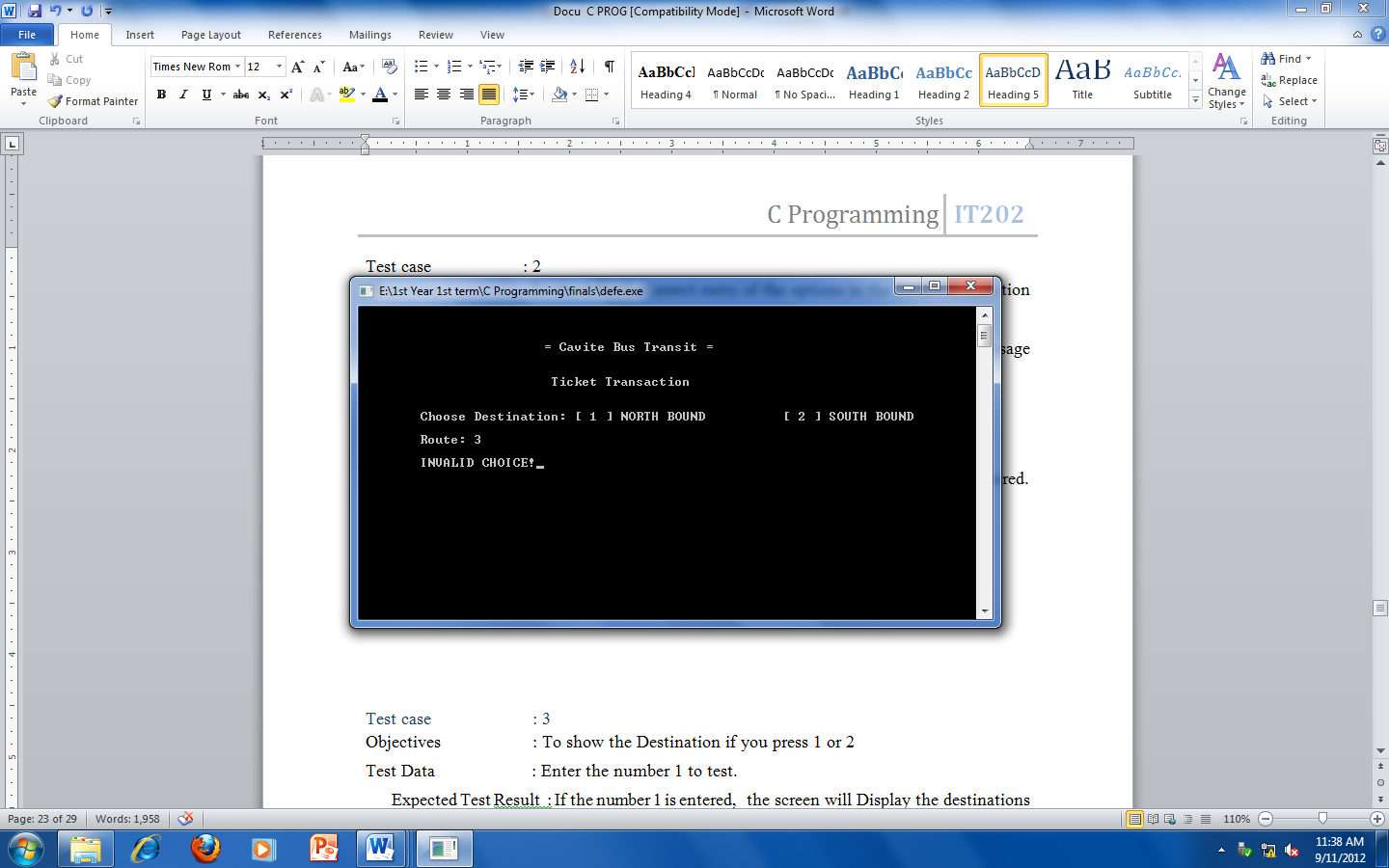
Objectives : To check for the correct entry of the options in the Ticket Transaction

Test Data : Enter the number 3 to test.

Expected Test Result: If you input 3 , the screen will display error message “INVALID CHOICE!”.

Actual Test Result : As below.

Conclusion : Succeeded. Error message was displayed when the test data was entered.



##### Test case : 3

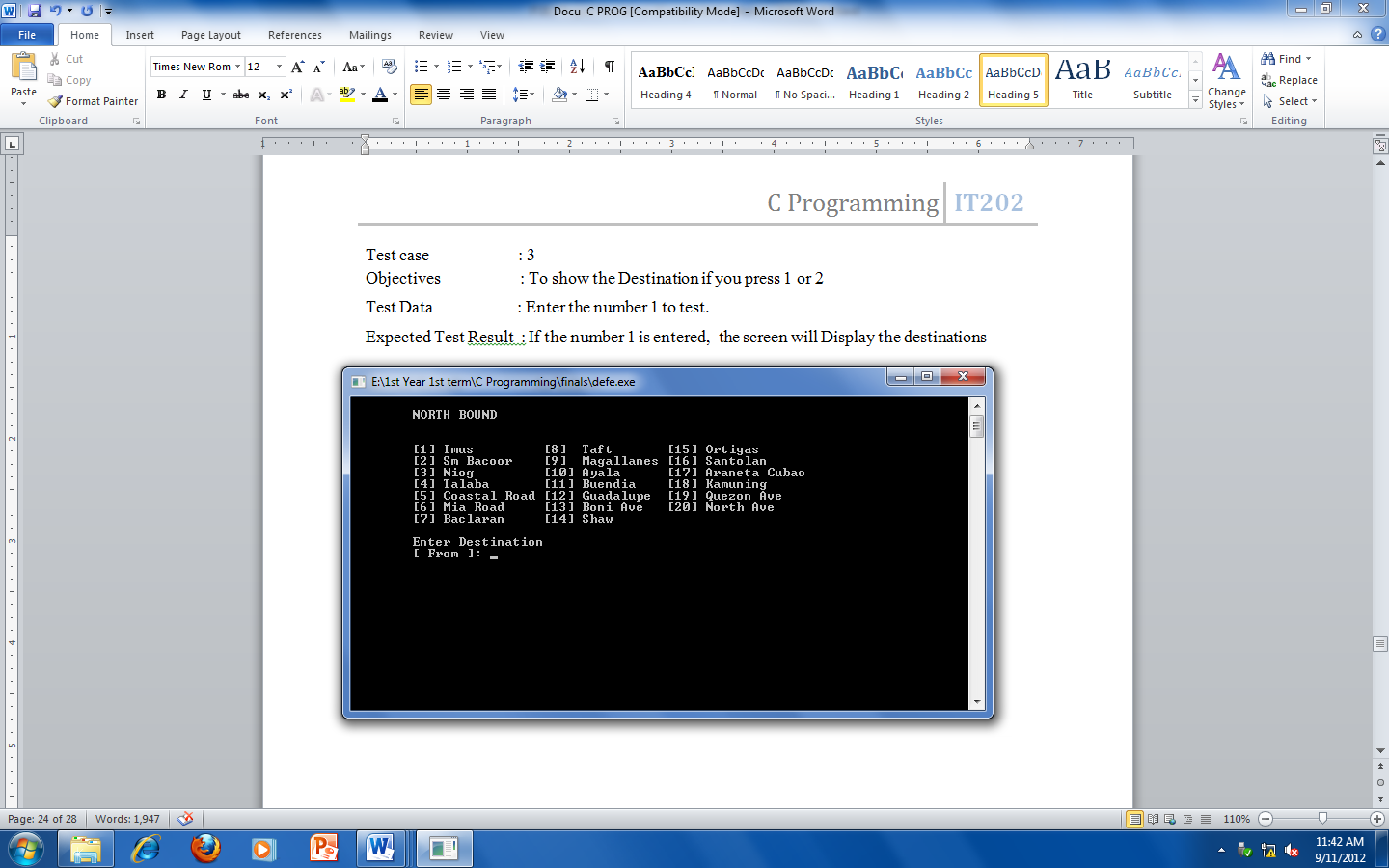
Objectives : To show the Destination if you press 1 or 2

Test Data : Enter the number 1 to test.

Expected Test Result : If the number 1 is entered, the screen will Display the destinations

Actual Result : As below.

Conclusion : Succeeded. The destinations appeared.



##### Test case : 4

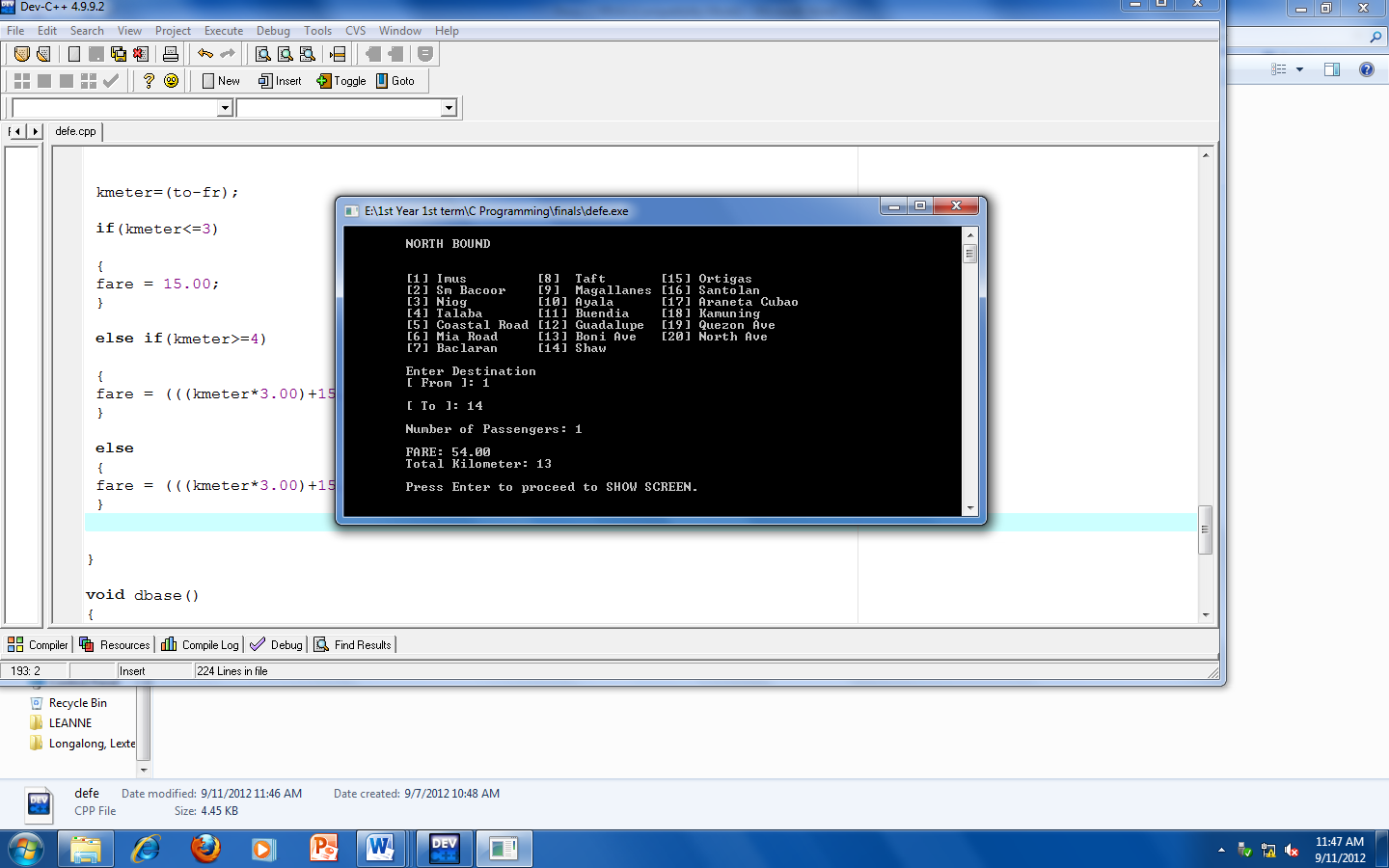
Objectives : To show the fare total and the total kilometers after picking the destination

Test Data : Press Enter after picking the Destination

Expected Test Result: The Fare Total and Total Kilometers Will show

Actual Result : Shown Below

Conclusion : Succeeded, The fare total and the total kilometers appeared.



##### Test case : 5

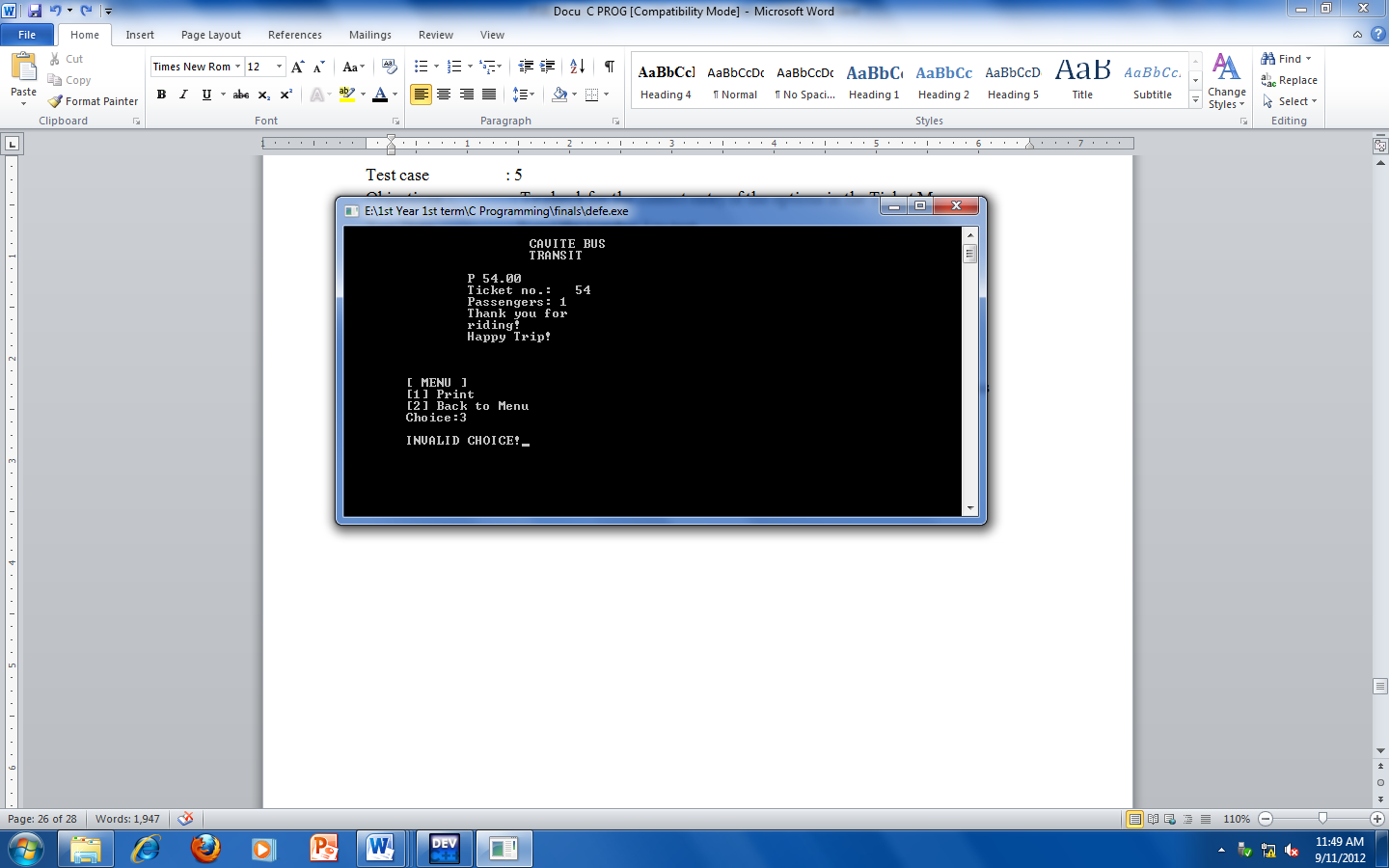
Objectives : To check for the correct entry of the options in the Ticket Menu.

Test Data : Enter the number 3 to test.

Expected Test Result: If the number 3 is entered, the screen will display error message “INVALID CHOICE!”.

Actual Test Result : As below.

Conclusion : Succeeded. Error message was displayed when the test data was entered.

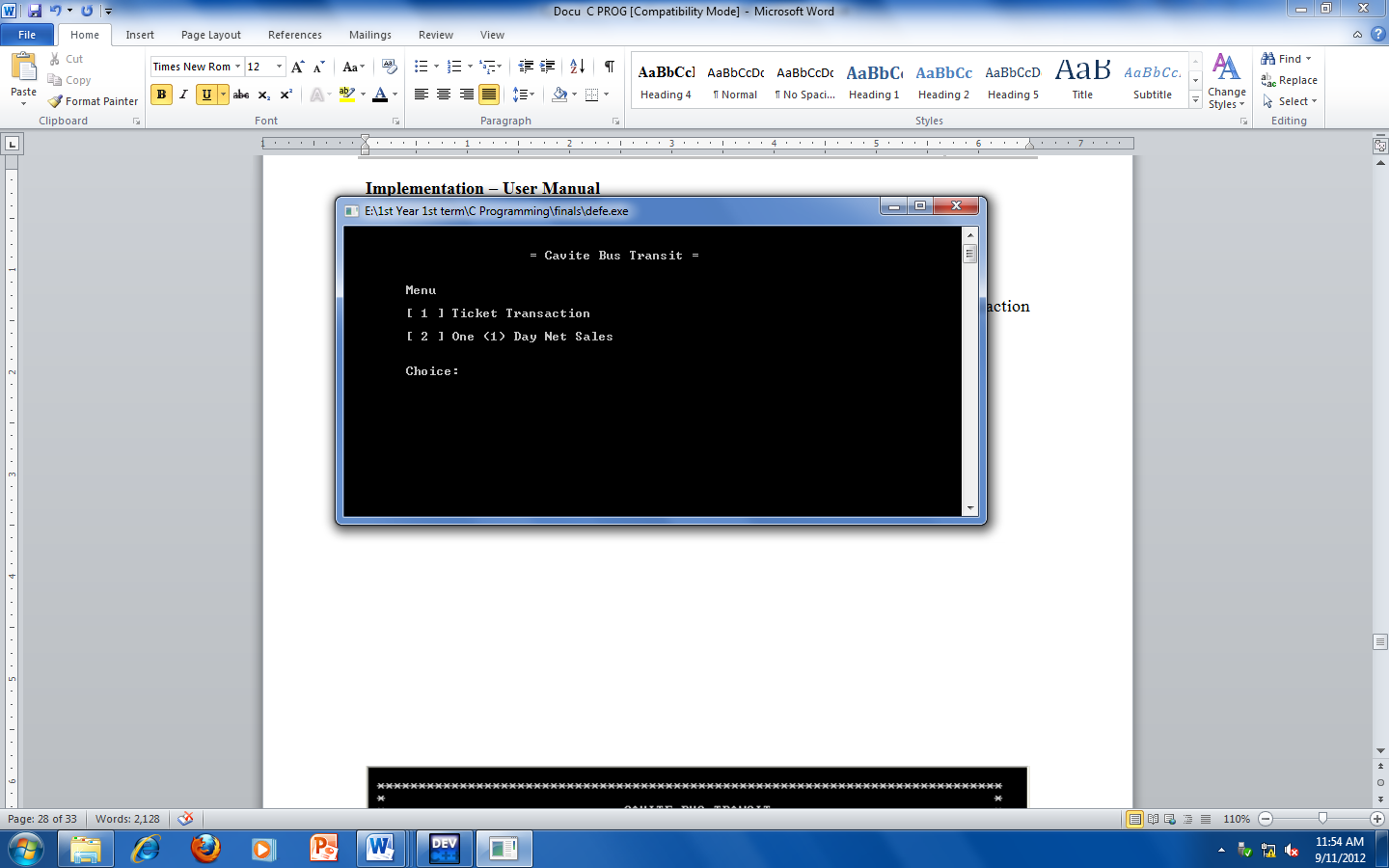


**TEST LOG**

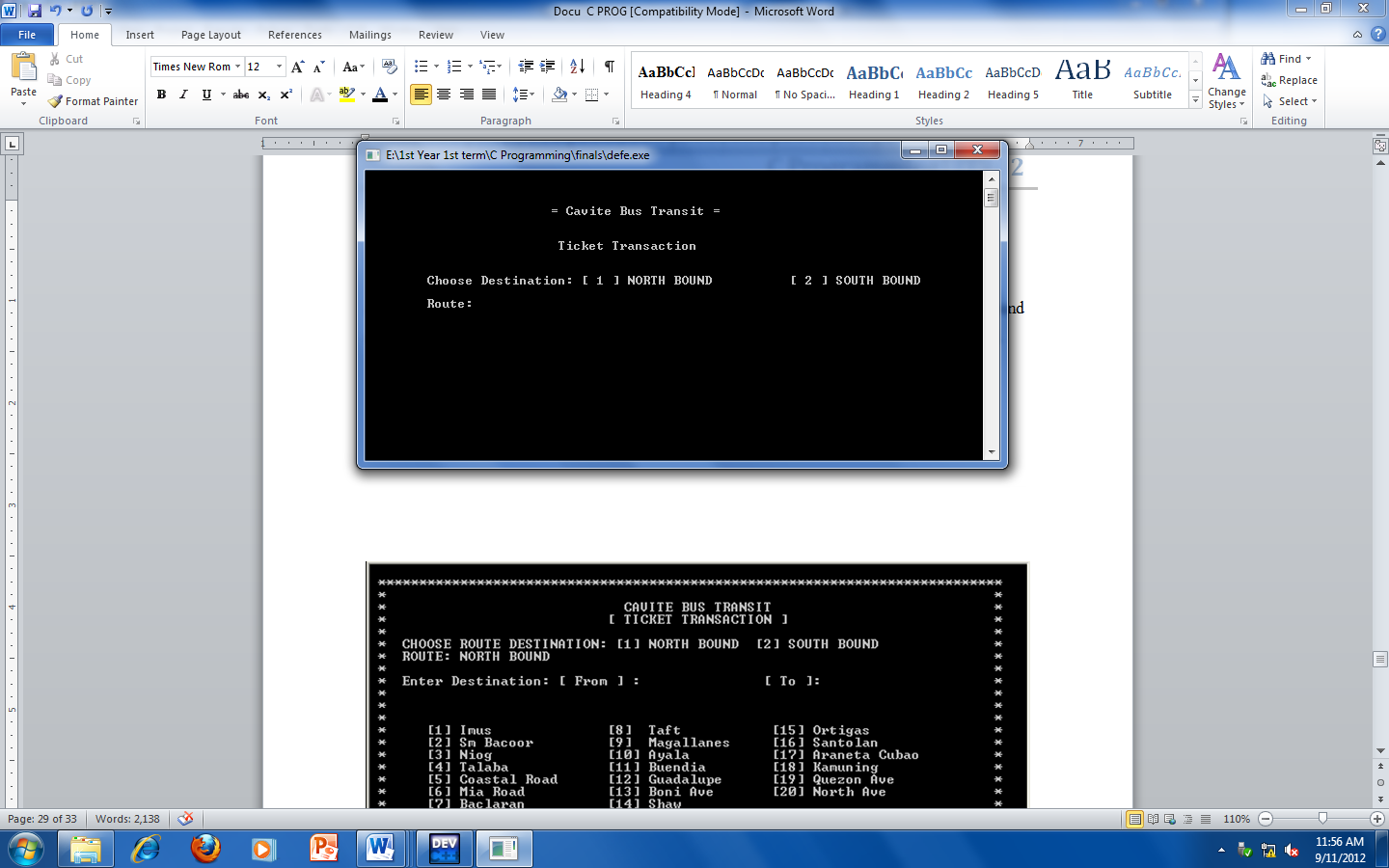
|  |  |
| --- | --- |
| **Test Cases** | **Status** |
| Case 1 | **Successful** |
| Case 2 | **Successful** |
| Case 3 | **Successful** |
| Case 4 | **Successful** |
| Case 5 | **Successful** |

**CHAPTER 3: INPLEMENTATION**

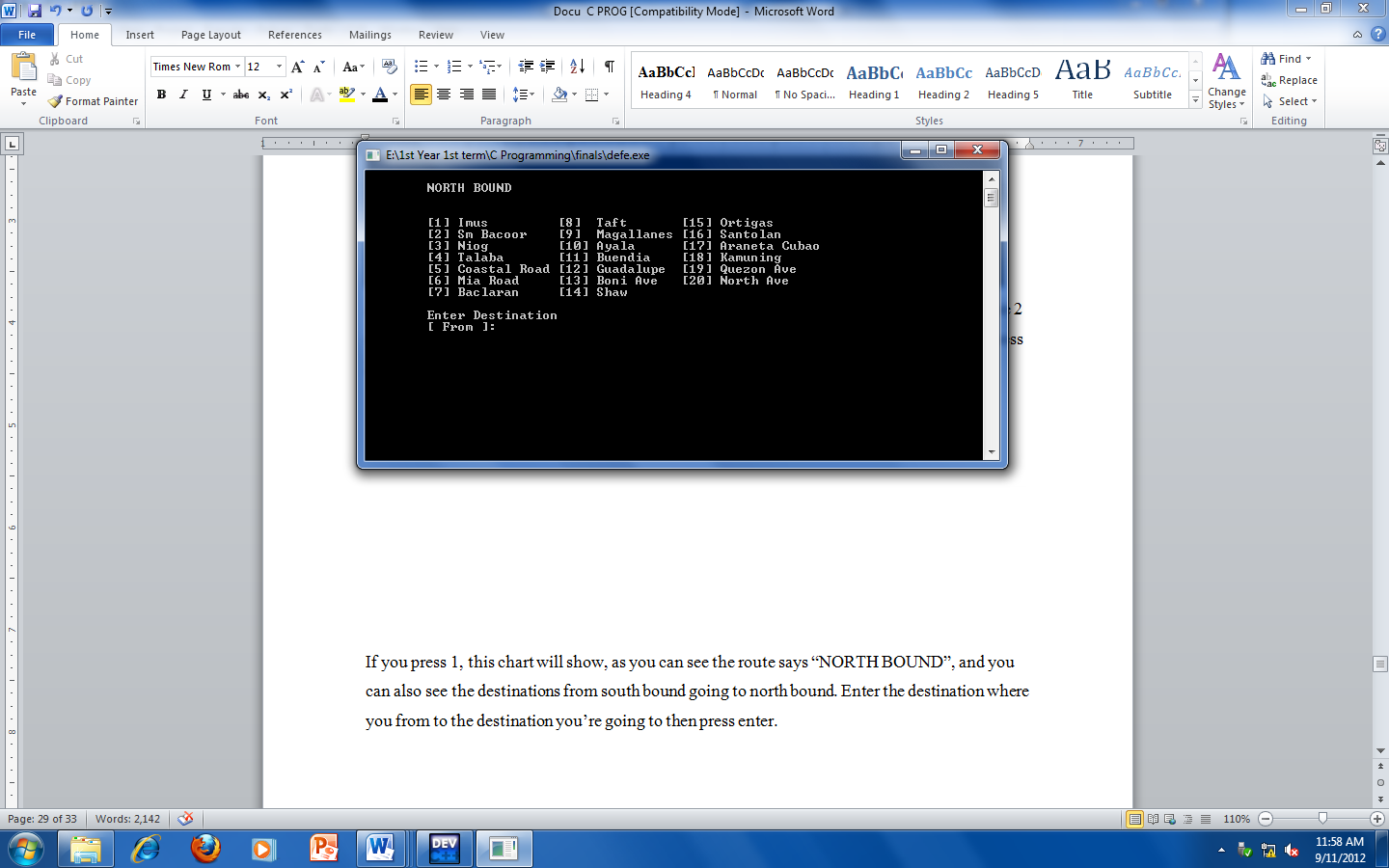
**USERS MANUAL**



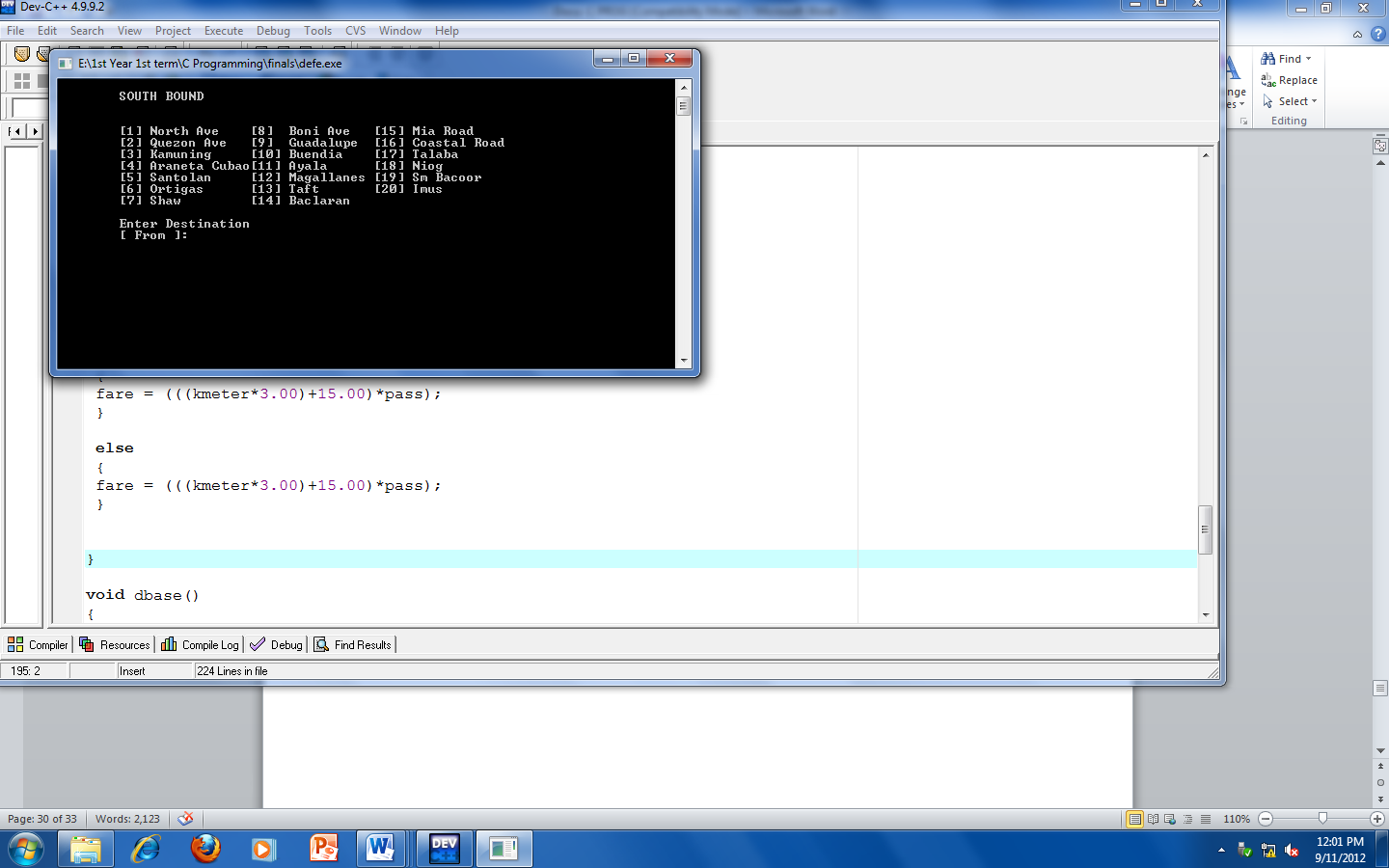
In Main menu, the ticket transaction menu is shown, press “ 1 “ to go to the ticket transaction menu. Press “ 2 “ to go to the Net Sales.



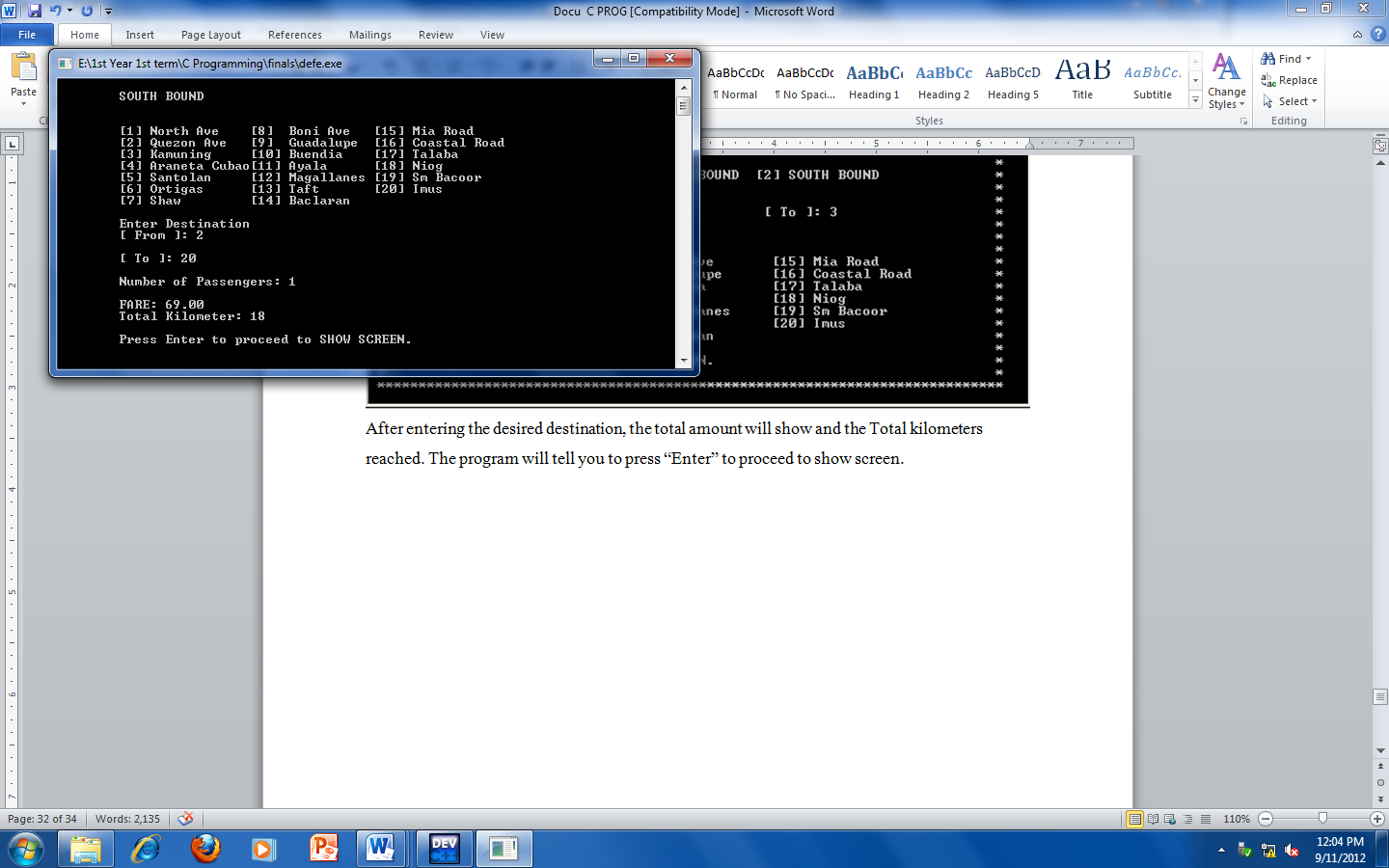
After pressing “ 1 “, this chart will show the ticket transaction menu. As you can see, there are 2 choices, the [1] North Bound and the [2] South bound. Press 1 if going to north bound and press 2 if going to south bound.



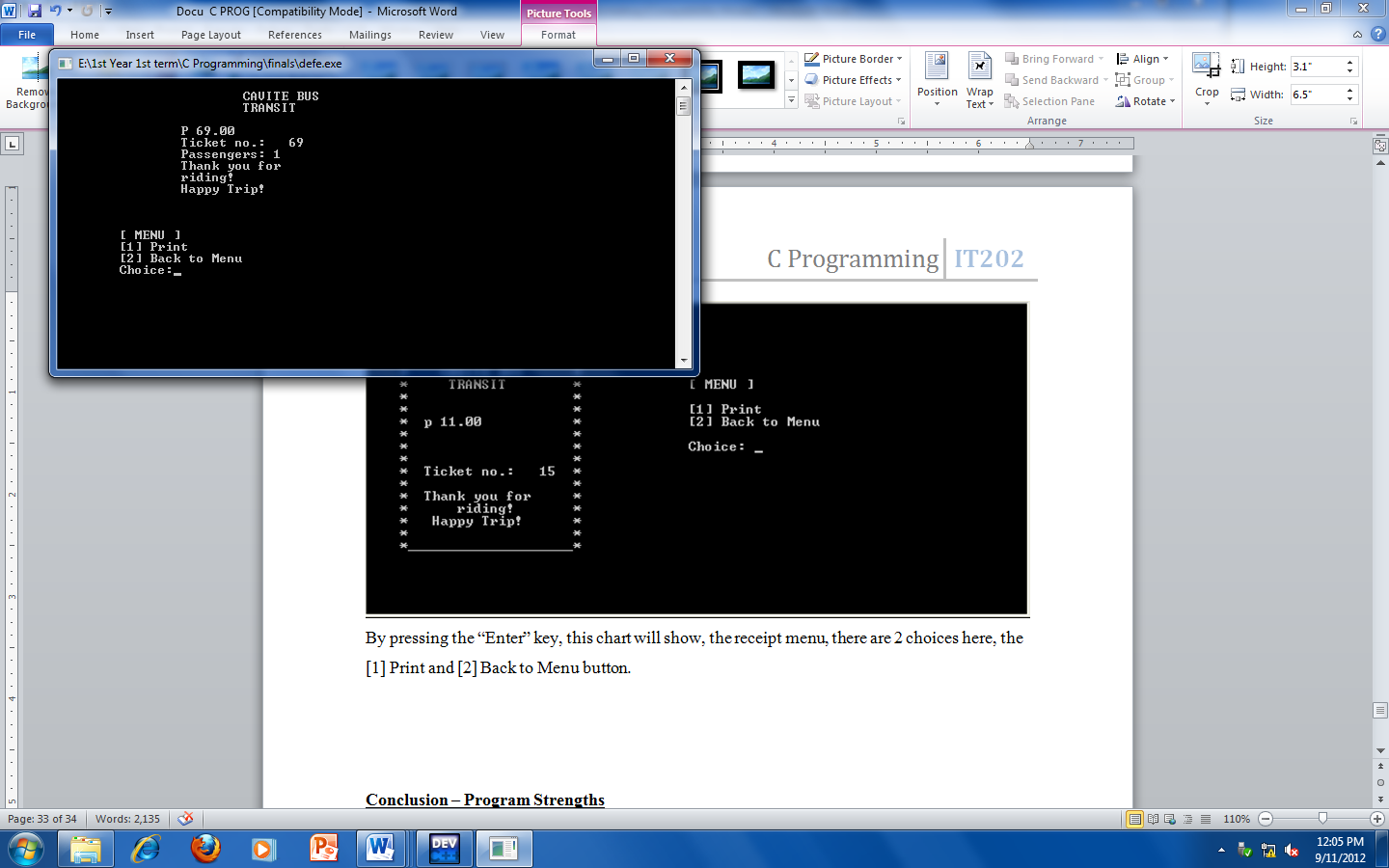
If you press 1, this chart will show you North Bound Transaction. Enter the destination where you from to the destination you’re going to then press enter.



If you press 2, this chart will show you to South Bound Transaction. It is the same as the 1st chart. The difference is the route, now its south bound going to north bound.



After entering the desired destination, the total amount will show and the Total kilometers reached. The program will tell you to press “Enter” to proceed to show screen.



By pressing the “Enter” key, this chart will show, the receipt menu, there are 2 choices here, the [1] Print and [2] Back to Menu button.

**CONLUSIONS**

**Program Strengths**

For my conclusion, I would like to pass some comments about my program. Basically, my system is all about bus ticketing it can compute the fare of a customer based on the kilometers he/she consumed. Every kilometer has a fix rate of P 15.00, now every additional kilometer has a fixed rate of P 3.00 per Km.

**Program Weakness**

My programs weakness is that it can only compute the kilometers consumed. Discount in terms of student/senior/disabled, username and password for security purpose is not yet included.

**Program Enhancements**

In the near future, some enhancements can be made to the program. A fare adjustment can be added to the program so that the user can change it in case there’s a change in the minimum fare.