

# AN INTELLIGENT CREDIT CARD FRAUD DETECTION SYSTEM

<sup>1</sup>EKWEALOR Oluchukwu Uzoamaka; <sup>2</sup>Dr. ANUSIUBA  
Overcomer Ifeanyi Alex; <sup>3</sup>EZURUKA Evelyn Ogochukwu  
<sup>4</sup>UCHEFUNA Charles Ikenna

<sup>1,2,3</sup> Department of Computer Science, Faculty of Physical Sciences, Nnamdi Azikiwe University Awka

<sup>4</sup> Department of Computer Science, Federal Polytechnic Oko

<sup>1</sup>uzoamakaekwee@gmail.com <sup>2</sup>oi.anusiuba@unizik.edu.ng

<sup>3</sup>eo.ezuruka@unizik.edu.ng <sup>4</sup>[iykenna@outlook.com](mailto:iykenna@outlook.com)

**DOI: 10.26821/IJSHRE.9.2.2021.9207**

## ABSTRACT

The objective of this work is to develop an Intelligent Credit Card Fraud Detection System which can detect and prevent online credit, card fraud. Hidden Markov Model was applied in determining the spending habit or profile of credit card holders. And with the spending profile established, it becomes possible to determine if an incoming transaction from a card holder is fraudulent or not by comparing any new transaction with the credit card holder's spending history. Any deviation from the actual spending habit is seen as a probable fraud and will be restricted while further verification is carried out. The methodology adopted for this work is Structured System Analysis and Design Methodology (SSADM) and the programming language used is PHP-MYSQL. The system developed can be used by any bank or financial institutions dealing on credit cards to detect and prevent all kinds of credit card fraud as it has been found to be capable of detecting such possible online fraud.

## 1.1 Background of the Study

A secured and trusted inter-banking network for electronic commerce requires high speed verification and authentication mechanisms that allow legitimate users easy access to conduct their business, while thwarting fraudulent transaction attempts by others. Fraudulent electronic transactions are already a significant problem, one that will grow in importance as the number of access points in the nation's financial information system grows. (Stolfo et al., 1998).

The popularity of on-line shopping is growing day by day. According to an AC Nielsen study conducted in 2005, one-tenth of the world's population is shopping on-line. In today's increasingly electronic society and with the rapid advances of electronic commerce on the Internet, the use of credit cards for purchases has become convenient and necessary.

Credit card transactions have become the de facto standard for Internet and Web based e-commerce. The US government estimates that credit cards accounted for approximately US \$13 billion in Internet sales during 1998. This figure is expected to grow rapidly each year. Germany and Great Britain have the largest number of on-line shoppers and credit card is the most popular mode of payment (59%). About 350 million transactions per year were reportedly carried out by Barclaycard, the largest credit card company in the UK, towards the end of the last century (Hand, D.J. et al, 2000).

The increase in number of credit cards and credit card transactions, which has become more significant in recent years, has increased the number of types of relative frauds. The total credit card fraud in the USA itself is reported to be \$2.7 billion in 2005 and estimated to be \$3.0 billion in 2006 out of which \$1.6 billion and \$1.7 billion, respectively, are the estimates of on-line fraud (Statistics for General and On-line Card Fraud, 2007).

Credit card fraud cases are increasing every year. In 2008, number of fraudulent activities through credit card had increased by 30 percent because of various ambiguities in issuing and managing credit Cards. Credit card fraudulent is approximately 1.2% of the total transaction amount, although it is not small amount as compared to total transaction amount which is in trillions of dollars in 2007. Card issuers must take more precaution against fraud detection and financial losses. When banks lose money because of credit card fraud, cardholders pay for all of that loss through higher interest rates, higher fees, and reduced benefits. In addition to financial losses, fraud may cause distress, loss of service, and loss of customer confidence (Hoath, 1998). Hence, it is in both the banks' and the cardholders' interest to reduce illegitimate use of credit cards by early fraud detection.

Online credit card fraud has become a huge problem on internet environment because this is the beneficial place for fraudsters stealing users' card details. Huge amount of money are stolen through online transactions and this causes e-consumers significant concerns because of its dramatic speed day-by-day (Bhattacharyya et al., 2011); these issues are challenging the development of online credit card transactions. Consequently, detection of fraudulent transactions and online credit card frauds has become essential to maintain the viability of online transactions and banking systems (Patil et al., 2010).

Over the years, along with the development of fraud detection methods, fraudsters have evolved their fraud methods to avoid detections (Bhattacharyya et al, 2011). Therefore, fraud detection methods need to be developed also. Many algorithms have evolved to detect online credit card frauds based on data mining techniques. This powerful tool helps to extract and analyze various types of data to give decisions to detect and prevent frauds. Some data mining techniques used for credit card fraud detection are neural networks, decision trees, random forests, Hidden Markov Model, Social Network Analysis and logistic regression (Bhattacharyya et al., 2011; Brause et al 1999; Patil et al., 2010; Thiruvadi & Patel, 2011).

For many years, the credit card industry has studied computing models for automated detection systems; recently, these models have been the subject of academic research, especially with respect to e-commerce.

Credit card based purchases can be categorized into two types:

- (a) Physical card and
- (b) Virtual card.

In a physical-card based purchase, the cardholder presents his card physically to a merchant for making a payment. To carry out fraudulent transactions in this kind of purchase, an attacker has to steal the credit card. If the cardholder does not realize the loss of card, it can lead to a substantial financial loss to the credit card company.

In the second kind of purchase, only some important information about a card (card number, expiration date, secure code) is required to make the payment. Such purchases are normally done on the Internet or over the telephone. To commit fraud in these types of purchases, a fraudster simply needs to know the card details. Most of the time, the genuine cardholder is not aware that someone else has seen or stolen his card information.

The best way to detect this kind of fraud is to analyze the spending patterns on every card and to figure out any inconsistency with respect to the “usual” spending patterns. Fraud detection based on the analysis of existing purchase data of cardholder is a promising way to reduce the rate of successful credit card frauds. Since humans tend to exhibit specific behaviorist profiles, every cardholder can be represented by a set of patterns containing information about the typical purchase category, the time since the last purchase, the amount of money spent, etc. Deviation from such patterns is a potential threat to the system. Hidden Markov Model will be helpful to find out the fraudulent transaction by using spending profiles of user.

## 1.2 Statement of the Problem

Although some systems have been developed to check credit card frauds, it is obvious that we still witness some difficulties and draw backs. This is because the existing systems can only detect fraud after the fraud is done that is, the fraud is detected after the compliant of the card holder and so the card holder faced a lot of trouble before the investigation is concluded.

Another major issue with existing systems is that they require labeled data for both genuine as well as fraudulent transactions to train the classifiers and getting real world fraud data is one of biggest problems associated with credit card fraud detection. Also, these approaches cannot detect new kinds of frauds for which labeled data is not available. Also, large data is needed for which a log is maintained that consists of all the transaction.

Presently, due to rapid and wide usage of credit-cards (physical usage and virtual usage), it is difficult to

draw out how and for what the credit-cards are used. Therefore, the IP addresses of transactions made online are captured for verification purpose. Thus, when a credit-card fraud is committed, this process needs help from the cyber crime in order to investigate the case (Srivatsa 2008). Typically, this process is difficult, time taking and not completely reliable. Therefore to avoid the above disadvantages, an Intelligent Credit Card Fraud Detection System, is proposed which is easier and the best way to detect the credit-card frauds.

### 1.3 Objectives of the Study

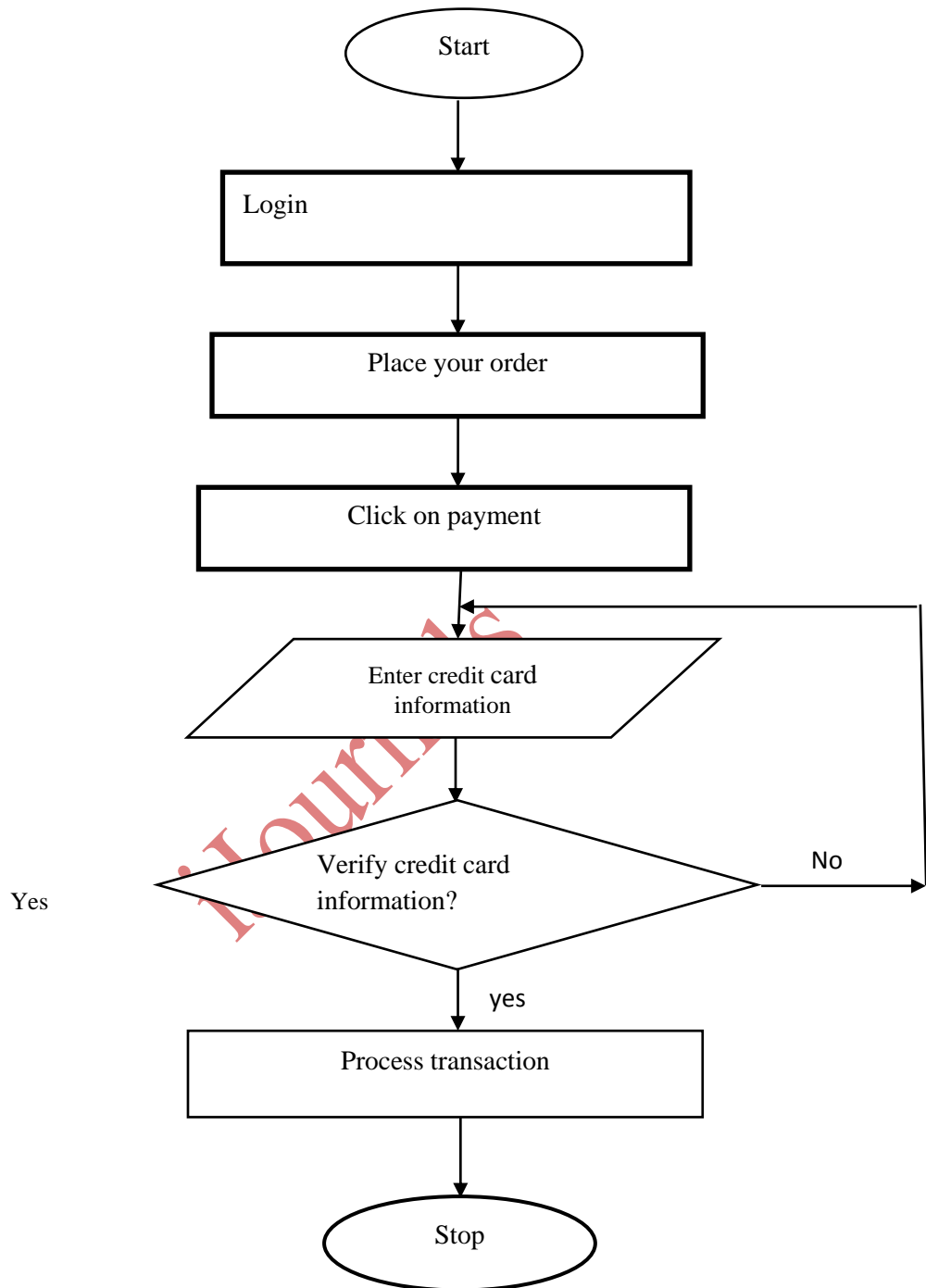
The objectives of this work include developing a system that should be able to:

- i. Identify the spending profile or habit of credit card holders and apply this knowledge in deciding whether a transaction is fraudulent or not.
- ii. Determine the spending patterns on every credit card during transaction to figure out any inconsistency with respect to the “usual” spending patterns.
- iii. Prevent credit card fraudulent transactions from being effected based on any observed differences in the card user’s spending profile by sending alarm to the issuing bank.
- iv. Reduce the number of False Positives transactions (i.e. transactions that are detected as fraudulent transaction whereas they are actual transactions) associated with existing fraud detection systems by ensuring that genuine transactions are not rejected.

### 1.4 Analysis of the Existing System

In existing models, the bank verifies credit card information, Card Verification Value (CVV), Date of expiry etc., but all these information are available on the card itself. Presently, banks also request users to register their credit card for online secure password, but after feeding details of card at merchant site, it will only be transferred to a secure gateway established at bank’s own server; it does not verify that the transaction is fraudulent or not. So once hackers get secure code of credit card by any source, it is very difficult to trace fraudulent transaction. Thus the fraud can only be detected after the complaint of the card holder, that is after the fraud is committed and this makes the card holder to face a lot of trouble before the investigation is finished. Also, due to rapid and wide usage of credit-cards (physical usage and virtual usage), it is difficult to draw out how and for what the credit-cards are used. The IP addresses of the transactions made online can be captured for corroboration. Thus, when a credit-card fraud is committed, this process needs help from the cyber crime in order to investigate the case (Srivatsa 2008). Typically, this process is not completely reliable, difficult and time taking.

**Flowchart of the Existing System**



**Figure 1: Flowchart of the Existing System**

### **1.5 Analysis of the Proposed System**

Having observed the weaknesses and inefficiencies associated with the present fraud detection systems, this work introduced an Intelligent Credit Card Fraud Detection System using Hidden Markov Model (HMM) that does not need any fraud transactions information of the credit-card and still can detect the fraud actions.

This model considers the spending routines of the credit-card holder. Here, transactions of a credit card processing series are modeled by the stochastic procedure. The information related to the purchases with respect to an individual credit card holder is not identified by that particular bank's Fraud Detection System which issued the credit-card. This is the primary factor of Markov chain, which is signified but not noticeable. The credit card transactions can be viewed or known by stochastic process which gives the series of the spending information.

A Hidden Markov Model is a finite set of states; each state is linked with a probability distribution (Rabiner, 1989). Transitions among these states are governed by a set of probabilities called transition probabilities. In a particular state a possible outcome or observation can be generated which is associated symbol of observation of probability distribution. It is only the outcome, not the state that is visible to an external observer and therefore states are "hidden" to the outside; hence the name Hidden Markov Model.



Hidden Markov Model is a perfect solution for addressing detection of fraud transaction through credit card. One more important benefit of the HMM-based approach is an extreme decrease in the number of False Positives transactions recognized as malicious by a fraud detection system even though they are really genuine.



In this prediction process, HMM consider mainly three price value ranges such as.

- Low (l),
- Medium (m) and,
- High (h).

First, it will be required to find out transaction amount belonging to a particular category either it will be in low, medium, or high ranges and it involves two modules which include:

#### **i. Online Shopping**

It comprises of many steps, first is to login into a particular site to purchase goods or services, then choose an item and next step is to go to payment mode where credit card information will be required. After filling all these information, the page will be directed to the fraud detection system which will be installed at bank's server or merchant site.

## ii. Fraud Detection System

All the information about credit card (Like Credit card number, credit card CVV number, credit card Expiry date, name on credit card etc.) will be checked with credit card database. If user entered database is correct then it will ask Personal Identity number (PIN). After matching of Personal Identity number (PIN) with database and account balance of user's credit card is more than the purchase amount, the fraud checking module will be activated.

The verification of all data will be checked before the first page load of credit card fraud detection system. If user credit card has less than ten (10) transactions, it will directly ask to provide personal information to do the transaction. Once database of ten (10) transactions is obtained, fraud detection system will start working. By using this observation, it determines users spending profile.

The purchase amount will be checked with spending profile of user. By transition probabilistic calculation based on HMM, it concludes whether the transaction is real or fraud. If transaction may be concluded as fraudulent transaction then user must enter security information. This information is related with credit card (like account number, security question and answer which are provided at the time of registration). If transaction is not fraudulent then it will direct to give permission for transaction. If the detected transaction is fraudulent then the security information form will arise. It has a set of question where the user has to answer them correctly to do the transaction. These forms consist of information such as personal, professional, address; dates of birth, etc and are available in the database. If the information entered by the user matches with database information, then transaction will be done securely else user transaction will be terminated and transferred to online shopping website.

### 1.6 Specification of the Hidden Markov Model

Hidden Markov model can be fully specified with five parameters according to Rabiner, (1989).

1.  $N$ , the number of states in the model. We denote the set of all possible states as  $S = \{S_1, S_2, \dots, S_N\}$ , the state at time  $t$  as  $q_t$ .
2.  $M$ , the number of distinct observation symbols per state, i.e., the discrete alphabet size of the output set. We denote the set of all possible output symbols as  $V = \{v_1, v_2, \dots, v_m\}$ . the output symbols at time  $t$  as  $O_t$ . The sequence of observed symbols is denoted as  $O = O_1 O_2 \dots O_T$ .
3. The state transition probability distribution  $A = \{a_{ij}\}$ , where  $a_{ij} = P[q_{t+1} = S_j | q_t = S_i]$ ,

$$1 \leq i, j \leq N.$$

4. The observation symbol probability distribution in state  $j$ ,  $B = \{b_j(k)\}$ , where  $b_j(k)$

$$= P[O_t = V_k | q_t = S_j], 1 \leq j \leq N, 1 \leq k \leq M.$$

5. The initial state distributions  $\pi_i = P[q_1 = S_i], 1 \leq i \leq N$

A compact representation of the model is  $\lambda = (A, B, \pi)$ , where  $N, M$  are implicitly implied by  $A$  and  $B$ , (Rabiner, 1989).

### 1.7 Hidden Markov Model for Credit Card Transaction Processing

To map the credit card transaction processing operation in terms of an HMM, we start by first deciding the observation symbols in our model. We quantize the purchase values  $x$  into  $M$  price ranges  $V_1, V_2 \dots V_m$ , forming the observation symbols at the issuing bank. The actual price range for each symbol is configurable based on the spending habit of individual cardholders. HMM determines these price ranges dynamically by applying a clustering algorithm on the values of each cardholder's transactions.

We use  $V_k, k=1, 2 \dots M$  to represent both the observation symbol as well as the corresponding price range.

In this work, only three price ranges were considered, namely, low ( $l$ ), medium ( $m$ ) and high ( $h$ ). Our set of observation symbols is, therefore,  $V = \{l, m, h\}$  making  $M=3$ . For example:

Let  $l = (0, \text{₹}200]$ ,  $m = (\text{₹}200, \text{₹}600]$  and  $h = (\text{₹}600, \text{credit card limit}]$ . If a cardholder performs a transaction of ₹250, then the corresponding observation symbol is  $m$ .

A credit cardholder makes different kinds of purchases of different amounts over a period of time. One possibility is to consider the sequence of transaction amounts and look for deviations in them. However, the sequence of types of purchase is more stable compared to the sequence of transaction amounts. The reason is that a cardholder makes purchases depending on his need for procuring different types of items over a period of time. This, in turn, generates a sequence of transaction amounts. Each individual transaction amount usually depends on the corresponding type of purchase. Hence, we consider the transition in the type of purchase as state transition in our model. The type of each purchase is linked to the line of business of the corresponding merchant. This information about the merchant's line of business is not known to the issuing bank running the Fraud Detection System. Thus, the type of purchase of the cardholder is hidden from the Fraud Detection System. The set of all possible types of purchase and equivalently, the set of all possible lines of business of merchants form the set of hidden states of the HMM.

### 1.8 Dynamic Generation of Observation Symbols

For each cardholder, we train and maintain an HMM. To find the observation symbols corresponding to individual cardholder's transactions dynamically, we run a clustering algorithm on his past transactions. Normally, the transactions that are stored in the issuing bank's database contain many attributes. For this work, only the amounts that the cardholder spent were considered in his transactions. Although, there are various clustering techniques. K-means clustering algorithm was used to determine the clusters. K-means is an unsupervised learning algorithm for grouping a given set of data based on the similarity in their attribute values.

Each group formed in the process is called a cluster. The number of clusters  $K$  is fixed a priori. The grouping is done by assigning each observation to the cluster whose mean is closest to it.

In this work,  $K$  is the same as the number of observation symbols  $M$ . Let  $c_1, c_2, \dots, c_m$  be the centroids of the generated clusters. These centroids or mean values are used to decide the observation symbols when a new transaction comes in.

Let  $x$  be the amount spent by the cardholder in transaction. Fraud Detection System generates the observation symbol for  $x$  (denoted by  $O_x$ ) as follows:

$$O_x = \underset{i}{\text{Varg min}} |x - c_i| \quad (1)$$

Where:  $V$  = Observation symbol i.e  $V = (l, m, h)$   $X$  = amount spent by card holder in transaction,  $C_i$  = the centroid or mean of the generated clusters. ie.  $cl, cm, ch$ .

As mentioned before, the number of symbols is 3 in our system. Considering  $M = 3$ , if we apply X-means algorithm on the transactions in table 1, we get the clusters as shown in table 2 with  $cl, cm$  and  $ch$  as the respective centroids. It may be noted that the amounts 6, 8 and 10 have been clustered together as  $cl$  resulting in a centroid of 8. The percentage ( $p$ ) of total number of transactions in this cluster is thus 30%. Similarly, amounts 15, 15, 20, 20 and 25 have been grouped in the cluster  $cm$  with centroid 19 while amounts 40 and 80 have been grouped together in cluster  $ch$  with centroid 60.  $cm$  and  $ch$ , thus, contain 50% and 20% of the total number of transactions. When the Fraud Detection System receives a transaction for this cardholder, it measures the distance of the purchase amount  $x$  with respect to the means  $cl, cm$  and  $ch$ , to decide (using Eq. 1) the cluster to which the transaction belongs and hence, the corresponding observation symbol.

For example, if  $x = \$10$ , then from table 2 using Eq.1, the observation symbol is  $V_1 = l$ .

Table 1: Sample Transactions with the Amount spent in each Transaction

Transaction number	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup>	4 <sup>th</sup>	5 <sup>th</sup>	6 <sup>th</sup>	7 <sup>th</sup>	8 <sup>th</sup>	9 <sup>th</sup>	10 <sup>th</sup>
Amount (₹)	40	25	15	6	8	20	15	20	10	80

Table 2: Output of K-means Clustering Algorithm

Cluster mean/centroid name	C1	Cm	Ch
Observation symbol	V1=1	V2=m	V3=h
Mean value (Centroid)	8	19	60
Percentage of total transactions (p)	30	50	20

### 1.9 Spending Profile of Cardholders

The spending profile of a cardholder suggests his normal spending behavior. Cardholders can be broadly categorized into three groups based on their spending habits, namely, high spending (hs) group, medium spending (ms) group and low spending (ls) group.

Cardholders that belong to the high spending group, normally use their credit cards for buying high-priced items. Similar definition applies to the other two categories also. Spending profiles of cardholders are determined at the end of the clustering step.

Let  $Y$  be the percentage of total number of transactions of the cardholder that belong to cluster with mean  $c_i$ .

Then, the spending profile (SP) of the cardholder  $g$  is determined as follows:

$$SP(g) = \underset{i}{V_{\arg \max}}(Y) \quad (2)$$

Where: SP ( $g$ ) = Spending profile of card holder  $g$ ,  $V$  observation symbol ie.  $V = (1, m, h)$ ,  $Y$  = the percentage of total number of transactions of the card holder that belongs to each cluster.

Thus, spending profile denotes the cluster number to which most of the transactions of the cardholder belong.

From the example in table 2, the spending profile of the cardholder is 2, i.e.  $m$  and hence the cardholder belongs to the medium spending group.

1.10 Flowchart of the Proposed System

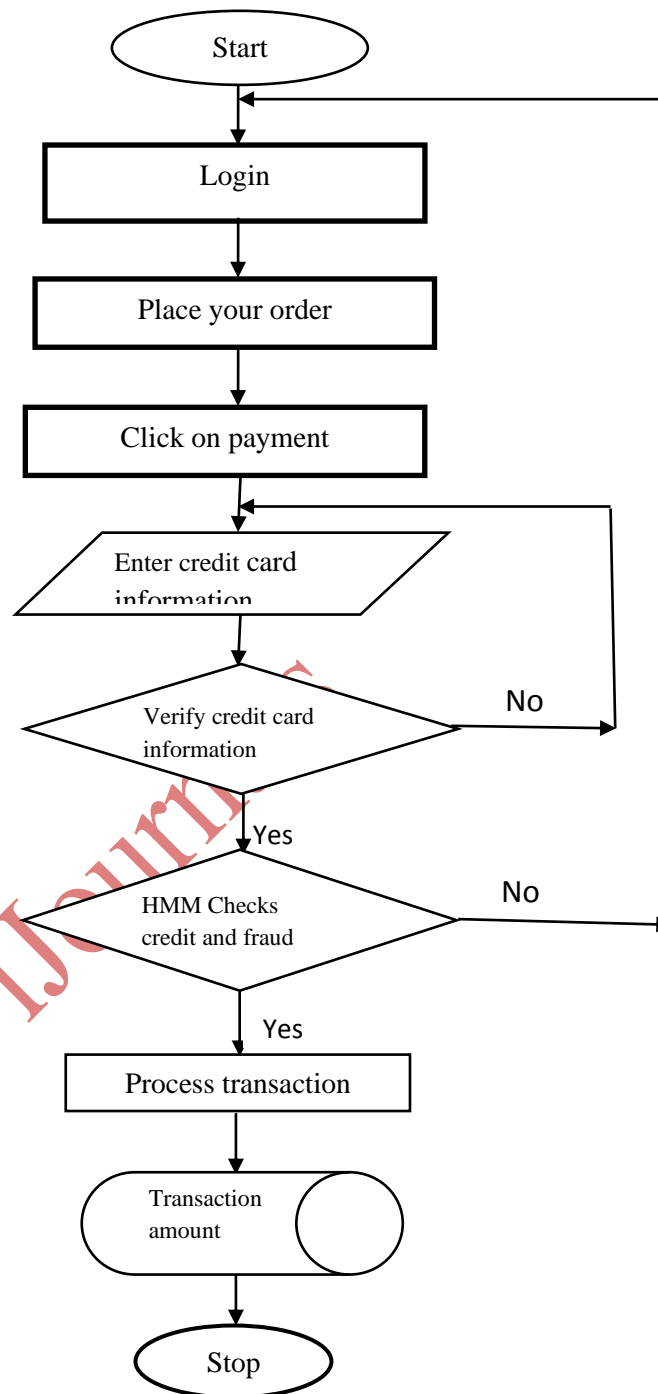


Figure 2: Flowchart of the Proposed System

1.11 High Level Model of the Proposed System

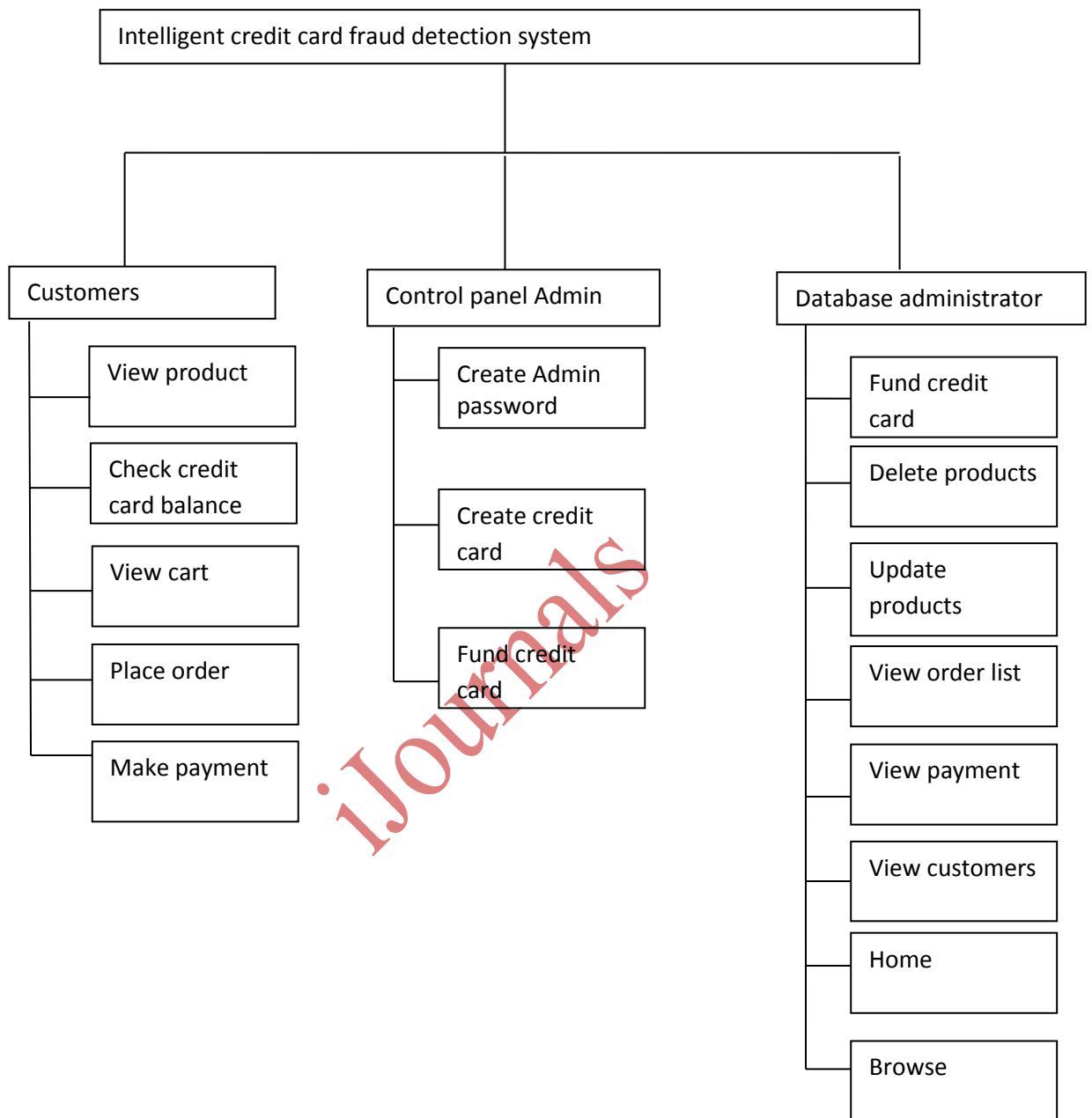
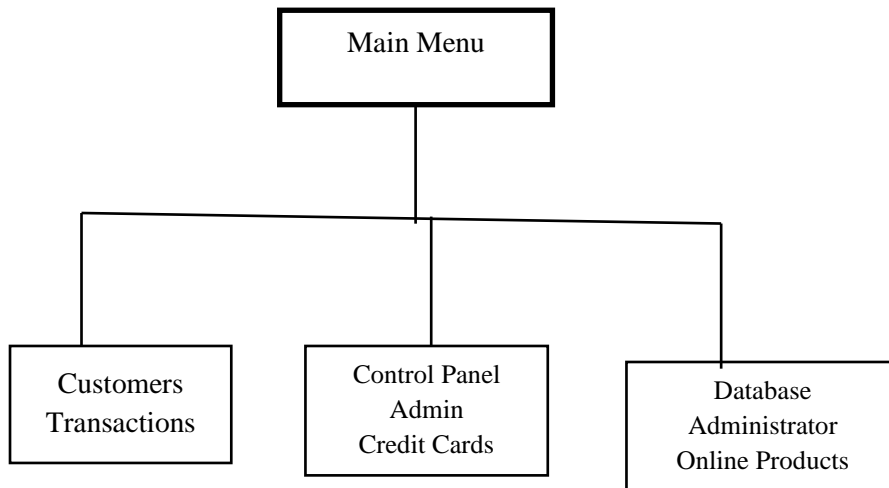


Figure 3: High Level Model of the Proposed System

**1.12 Main Menu**

The main menu is made up of three sub-menus which include customers’ transactions, control panel admin credit cards and database administrator online products.



**Figure 4: Main Menu**

**1.13 Database Specifications**

MySQL was used in the design of the new system’s database. The structures of the database are shown below:

- ecard
- tbl\_admin
- tbl\_cpanel
- tbl\_payment
- tbl\_product
- tbl\_user

**Table 3: Ecard table**

FIELD	TYPE	NULL	KEY	DEFAULT
First name	Varchar (20)	YES		(NULL)
Last name	Varchar (20)	YES		(NULL)
Card number	Varchar (40)	NO	PRI	
Signature	Varchar (20)	YES		(NULL)
Expdate	Date	YES		(NULL)
Pin	Int (6)	YES		(NULL)
Amount	Double	YES		(NULL)
Age	Int (6)	YES		(NULL)
Pet name	Varchar (20)	YES		(NULL)

**Table 4: Admin Login**

Field	Type	Null	Key	Default
Username	Varchar (10)	NO	PRI	
Password	Varchar (20)	YES		(NULL)

**Table 5: Order Table**

Field	Type	Null	Key	Default
Sn	Int(4)	NO	PRI	(NULL)
Id	Int(4)	YES		(NULL)
User	Varchar (20)	YES		(NULL)
Dates	Date	YES		(NULL)
Status	Varchar (10)	YES		(NULL)
Price	Double	YES		(NULL)
Product	Varchar (20)	YES		(NULL)

**Table 6: Cpanel Table**

Field	Type	Null	Key	Default
Username	Varchar (10)	NO	PRI	
Password	Varchar (20)	YES		(NULL)

**1.14 Program Modules Specification**

In the course of developing the program, we adopted Top – Down Design Approach which makes use of the fundamental program solving techniques. The software was structured in such a way that each subsystem was selected and executed independently. The task was divided into several modules, which were brought together to give the solution to the problem. These modules include the following:

**•User Module**

The user module was designed to enable users sign up, in order to view the company products and place purchase orders. On registration, the user obtains user name and password. Only valid credit card accounts can be used by the user in making payment for the purchases.

**•Admin Module**

The admin uses this module to create company products, maintain the product database and as well view customers purchase order. This module can also enable administrator to view account reports as well as upload product pictures on the web site.

**•Control Panel Module**

This module is used to create admin password as well as manage customer’s credit card account.

**1.15 Input Specification**

The new system was designed in such a way that well-structured forms will be used to capture customer’s transaction information as well as some administrative updates. Below are some of the input forms designed in the new system.

LOGIN FORM	
User name	<input type="text"/>
Password	<input type="password"/>
<input type="button" value="Login"/>	

PRODUCT ORDER PAYMENT FORM USING CREDIT CARD
<input type="text" value="USER SURNAME"/>
<input type="text" value="USER OTHER NAME"/>
<input type="text" value="ADDRESS"/>
<input type="text" value="PRODUCT"/>
<input type="text" value="QUANTITY OF PRODUCT"/>
<input type="text" value="PRODUCT AMOUNT"/>
<input type="text" value="TOTAL COST"/>
<input type="button" value="Pay"/>

The image displays two screenshots of web forms. The first is titled "New Product Entry Form" and contains three input fields for "Product name", "Price", and "Image". The "Image" field is accompanied by a "Browse..." button. Below these fields is an "Add product" button. The second screenshot is titled "Credit Card Balance Checking Form" and contains three input fields for "Credit Card", "Pin", and "Balance". Below these fields is a "Check" button.

**Figure 8: Credit Card Balance Checking Form**

### 1.16 Output Specification

The website was created to enable admin users to retrieve vital information from the site for management use. Reports on credit card balance, product ordering, customer's file and income report can be generated from the system. Below are some of the report formats designed new systems.

**Table 7: Product Order Report**

User	Product ID	Product name	Price	Status	Date
Chidi	1	Tv	70,000	Pd	2020-08-13
Oge	2	Laptop	82,000	Pd	2020-12-01

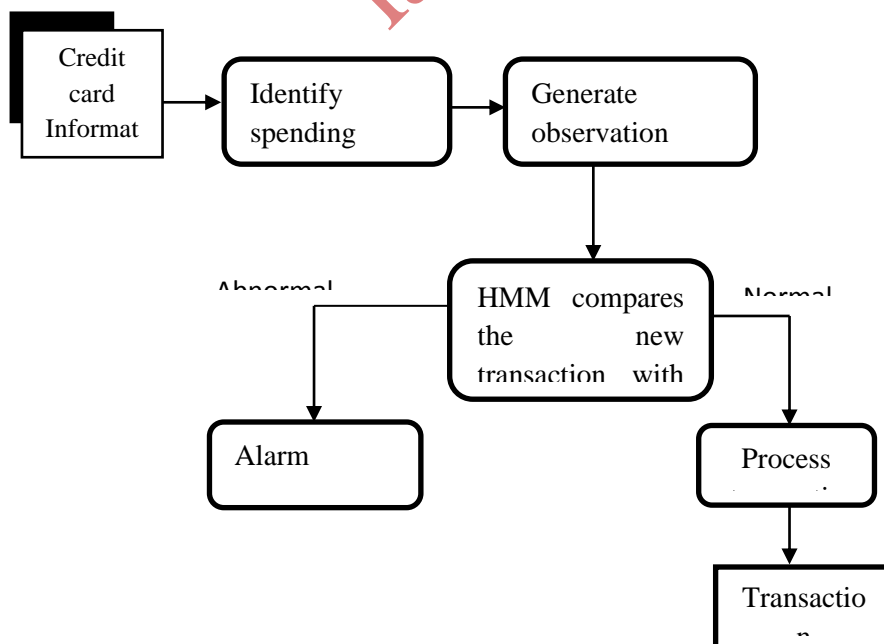
**Table 8: Income Report**

User	Price	Date
Chidi	200,000	2020-04-01

**Table 9: Customer Register**

Username	First Name	Last Name	Phone	E-mail	Home
Chibaby	Chika	Obi	08030416778	chi@gmail.com	Awka

**1.17 Overall Data Flow Diagram of the Proposed System**



**Figure 9: Overall Data Flow Diagram of the Proposed Solution.**

1.18 Flowchart

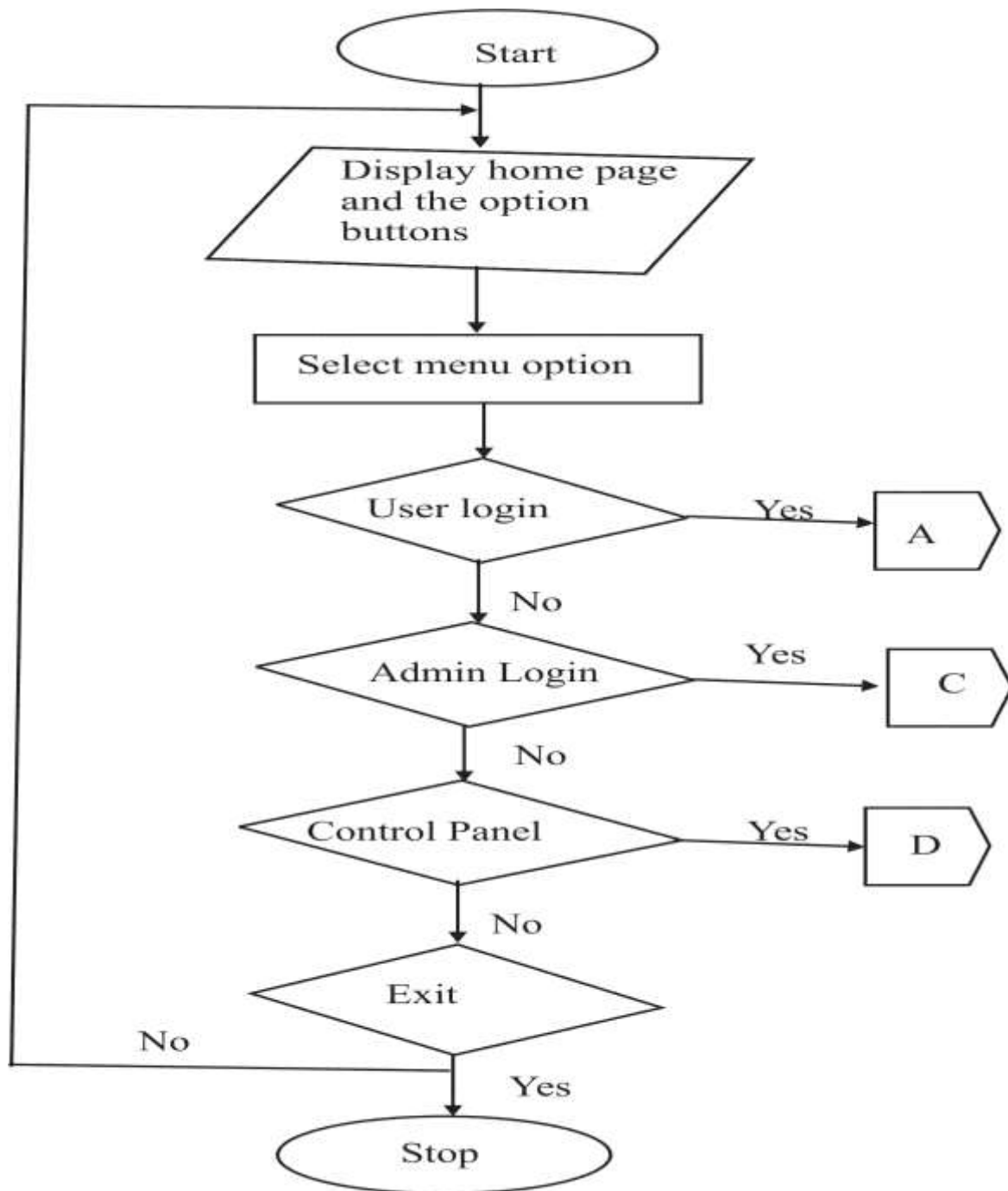


Figure 10: Home Page Flowchart

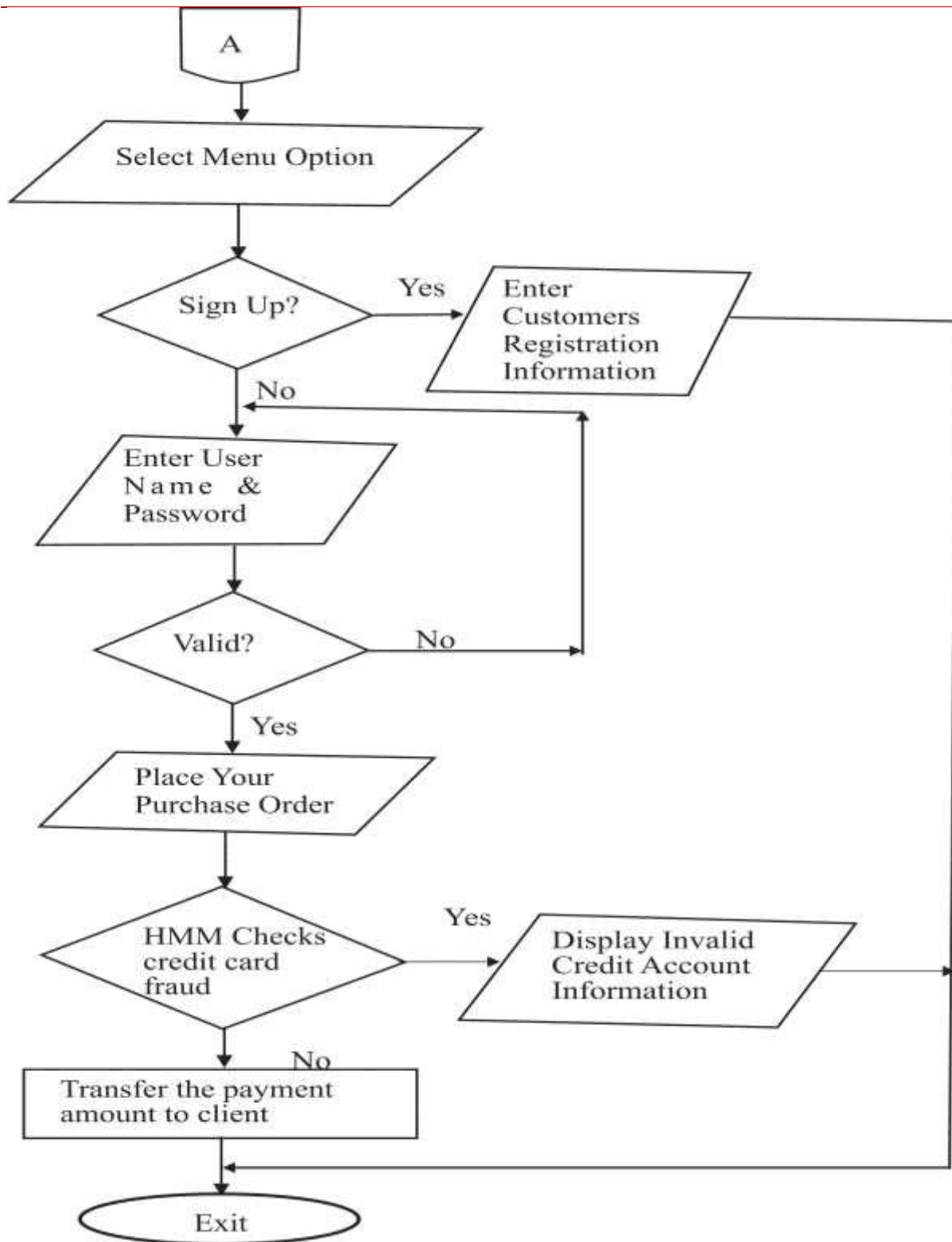


Figure 11: User Login Flowchart

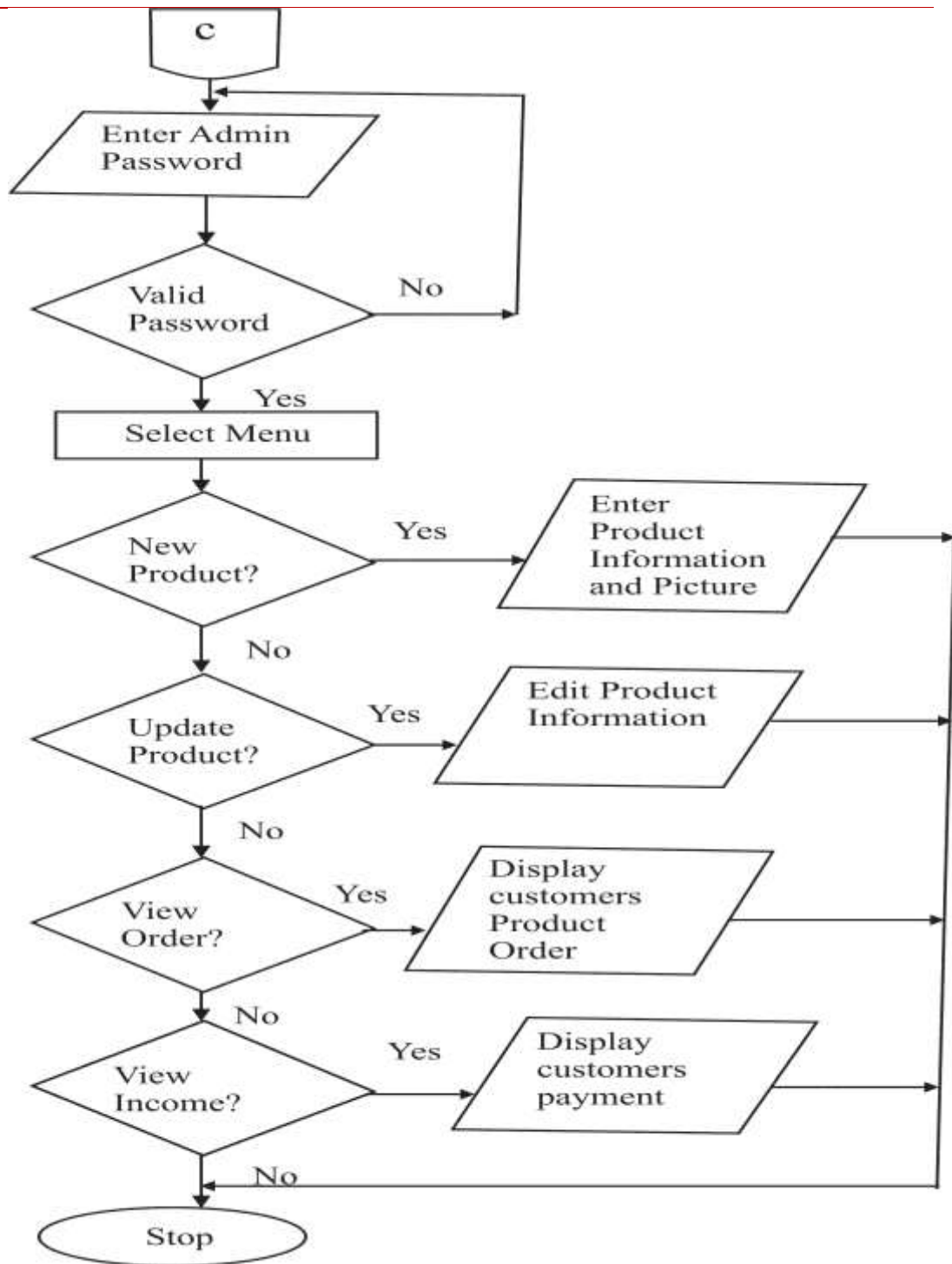


Figure 12: Admin Login Flowchart

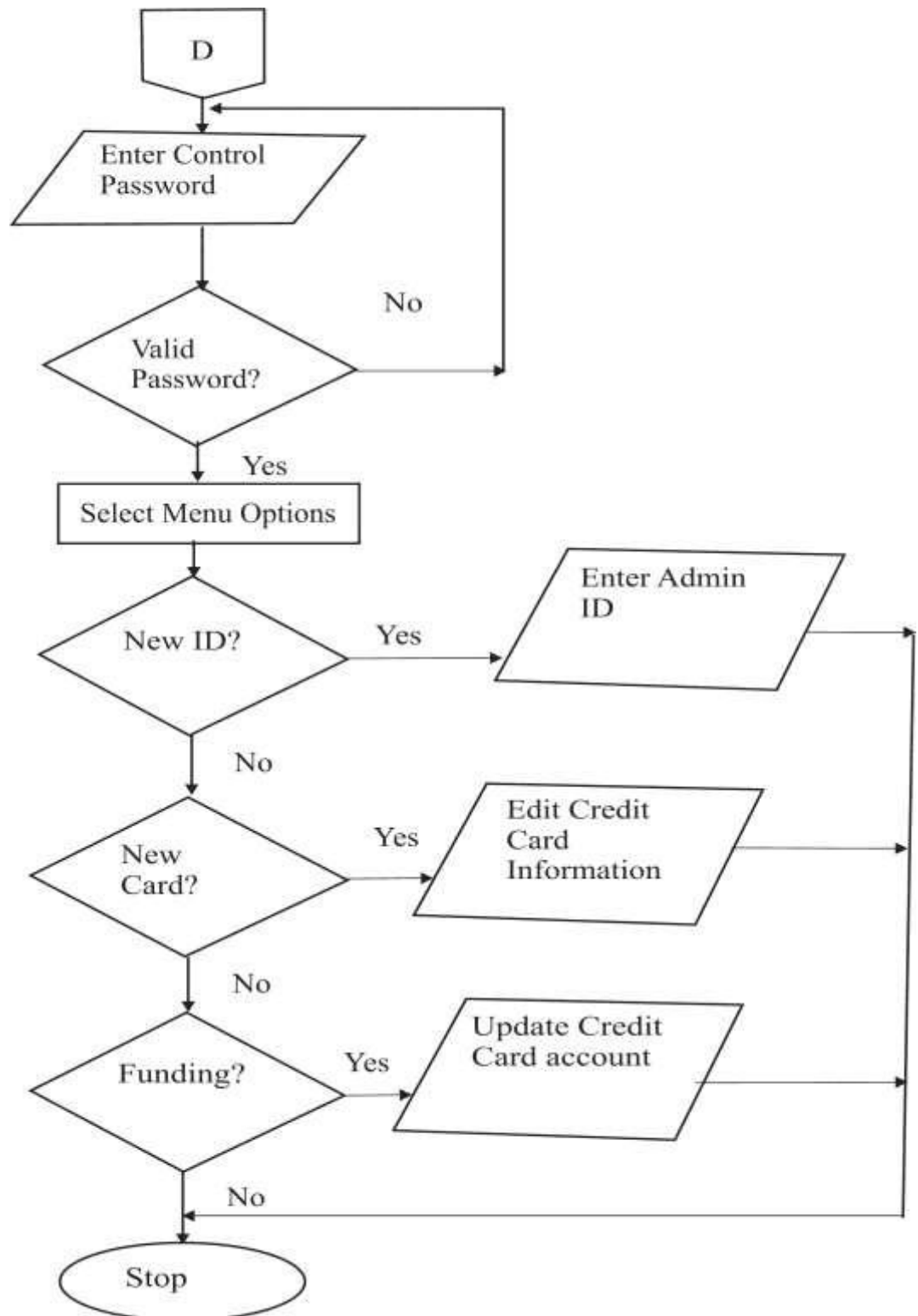


Figure 13: Control Panel Flowchart

### **1.18 Choice and Justification of Programming Language Used**

A lot of programming languages were considered in the course of designing this system putting a number of factors into consideration which includes online database access, data transmission via networks, database security, online database retrieval, multi user network access, online data capture, etc.

PHP-MySQL was chosen to enable us achieve the set objectives. Apart from being user friendly, PHP-MySQL helps to design an interface that can be modified programmatically. PHP based application can be developed using any integrated development Environment with any local host installed. PHP does not require complex coding structure to develop a web application; hence the developers can easily develop complex web applications in less usage of time. Also MYSQL database is a robust database that can guarantee database integrity, database protection, and accommodate large database.

### **1.19 Conclusion**

The work, an intelligent based credit card fraud detection system using Hidden Markov Model which is based on card holder's spending habits has been proved to be very effective in eradicating frauds associated with credit card transaction, as it thoroughly investigates every credit card transactions to ensure that any fraudulent transactions are restricted while reducing false positive transaction by ensuring that genuine card users are not denied transactions.

### **REFERENCES**

- Bhattacharyya, S., Jha, S., Tharakunne L. K. & Westland. J. C. (2011), Data mining for credit card fraud: A comparative study. *Decision Support Systems* 50, pp. 602-613, DOI: 10.1016/j.dss.2010.08.008.
- Brause, R., Langsdorf, T., Flepp, M. (1999), Neural Data Mining for Credit
- Hand, D.J., Blunt, G., Kelly, M.G., And Adams, N.M. (2000). Data Mining For Fun and Profit. *Statistical Science* 15, Pp. 111-131.
- Hoath, P. (1998). Telecoms fraud, the gory details. *Computer Fraud & Security* 20(1), Pp. 10-14.
- Patil, D. D., Karad, S. M., Wadhai, V.M., Gokhale, J.A. & Halgacaiuir. P. S. (2010). Efficient Scalable Multi-Level Classification; scheme far Credit Card Fraud Detection. *International Journal of Computer Science and Network Security*. Vol. 10, No. 8, Pp. 17-18.
- Rabiner. L.R. (1989). A Tutorial on Hidden Markov Models and Selected Applications In Speech

Recognition, *Proceedings of The IEEE*, Vol. 77, No. 2, Pp. 257-286.

Srivastava, A. Kundu, S. Sural, A.K. Majumdar, (2008). Credit Card Fraud Detection Using Hidden Markov Model, *IEEE Transactions on Dependable And Secure Computing*, Kharagpur, India, Pp. 212-2.6.

Stolfo, S. J., Fan. D. W, Lee, W., Prodrorakis. A. And Chan, P. K.. (1998). Cost-Based Modeling for Fraud and Intrusion Detection Results from the Jam Project, *Proceedings of Darpa Information Survivability Conference And Exposition Vol. 2 Pp. 130-144.*

Thiruvadi S. and Patel, S.C. (2011). Survey of Data Mining- Techniques Used in Fraud Detection and Prevention. *Information Technology Journal*, 1 0 (4) Pp. 710-716.

## APPENDIX 1

### SOURCE CODES

```
<td align="right" class="smallfont"><a href="adminlogin.php">Admin</a> 1 <a  
href="cpanel login.php">CPanel</a></td>  
</tr>  
<tr>  
<td>&nbsp;</td>  
<td align="right"><imgsrc="images/protection.jpg width="201"height="30" /></td>  
</tr>  
</table></td>  
</tr>  
</table></td>  
</tr>  
<tr>  
<td height="40" align="left" valign="middle" bgcolor="#EIE1E1"><table width="100%"  
border="0">  
<tr>  
<td width="1%">&nbsp;</td>  
<td width="84%" align="left">&nbsp;</td>  
<td width="15%">&nbsp;</td>  
</tr>  
</table></td>
```

```
</tr>
<tr>
<td height="270" align="left"><table width="990" height="270" border="0" cellpadding="0" cellspacing="0">
<tr>
<td width="150" align="left" valign="top" bgcolor="#000000"><table width="150"
Height="270" border="0" cellpadding="3" Cellspacing="1" bgcolor="#E1E1E1">
<tr>
<tr align="center" >
<td>Product ID </td>
<td>Product name < td>
<td>Price< td>
<td>Date< td>
$login = '<a href="Signin.php">login here </a>';
if ($_POST['username'])
{
$username = $_POST['username'];
$password = $_POST['password'];
$firstname = $_POST['firstname'];
$lastname = $_POST['lastname'];
$phone = $_POST['phone'];
$email = $_POST['email'];
$address = $_POST['address'];

$register = "insert into tbl_user (username, password, firstname, lastname, phone, email, address) values
('$username', '$password', '$firstname', '$lastname', '$phone', '$email', '$address')";

$query = mysql_query($register);

if (mysql_affected_rows == 1)
{
$msg = "Your registration was successful " . $login ;
}
}
```



```
$dt = date("Y-m-d");

$remove = '<a href=payment3.php?cardnumber=' .
$cardnumber . '&username=' . $username . '&password=' . $password . ' &amount=' . $amount .
&pin=' . $pin . ">Sorry, the credit card needs to be authenticated. Click here to continue </a>';

$morkov = "select * from tbl_payment where cardno = '$cardnumber'"; $detect =
mysql_query($morkov) ;//or die(mysql__error());
if (mysql_num_rows($detect)< 10)

{
print '<p bgcolor="#F1F1F1">' . $remove . '</p>';
}
Else
{

//check transaction value
$tim =0;
$valu=0;

while ($details = mysql_fetch_object($detect))
{
$amt = "$details->amount";

$valu =$valu + $amt;
$tim = $tim +1;
}

$avr= $valu / $tim;
$limitlow= $avr/2;
$limithigh = $avr * 3;
if (($amount < $limitlow) || ($amount > $limithigh))
{
print '<p bgcolor="#F1F1F1">' . $remove . '</p>';
}

else
```



```
{
// stop check

$select = "select * from tbl_user where username = '$username' and password = '$password'"; $login =
mysql_query($select) //or die(mysql_error());
if (mysql_num_rows($login)=1)
{
$welcome = $username:
$card = "select * from ecard where cardnumber = '$cardnumber' and pin = '$pin'";
$card 1 = mysql_query($card);
if (mysql_num_rows($card 1 )==1)
{
$gAmount = mysql_fetch_object($card1);
$tAmount= "$gAmount->amount";
if ($tAmount>=$amount)
{
    $balance = $tAmount - $amount;
    $updateCard = "update ecard set amount = '$balance' where cardnumber =
'$cardnumber'";
    $cardQuery = mysql_query($updateCard) or die ("card error");

    $updateOrder = "update tbl_order set status = 'p' where user = '$username'";
    $orderQuery = mysql_query($updateOrder) or die ("Order update error") :

    $payment = "insert into tbl_payment(user,amount,dates, cardno) values
('$username' $amount'$dt', '$cardnumber')";
    Mysql_query($payment);

    $msg = "Transaction completed successful":
```

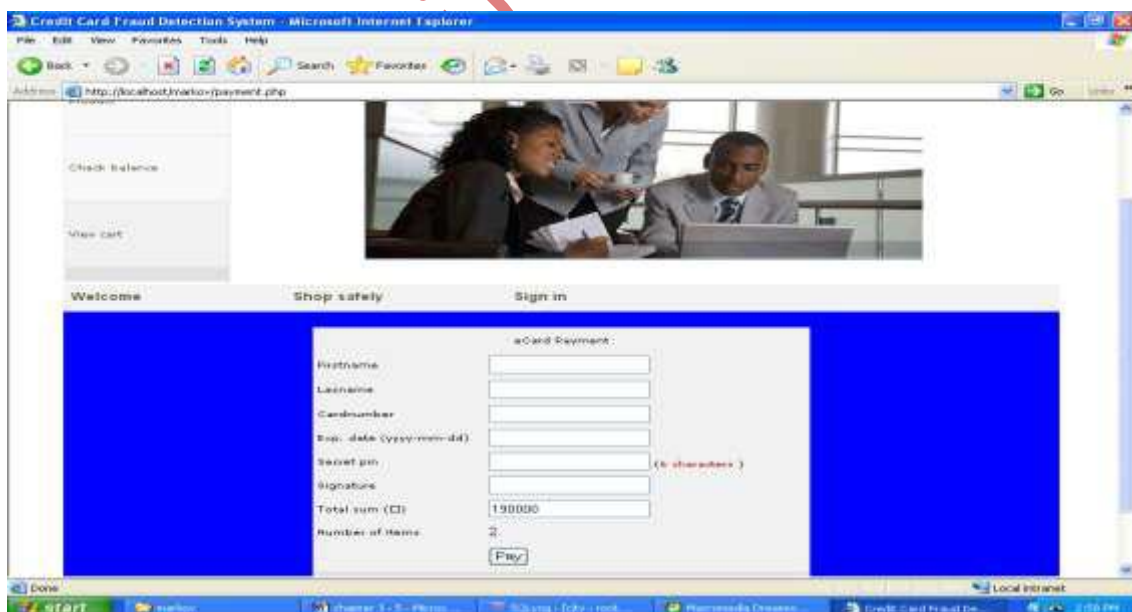
**APPENDIX 2**

**SAMPLE OUTPUT**

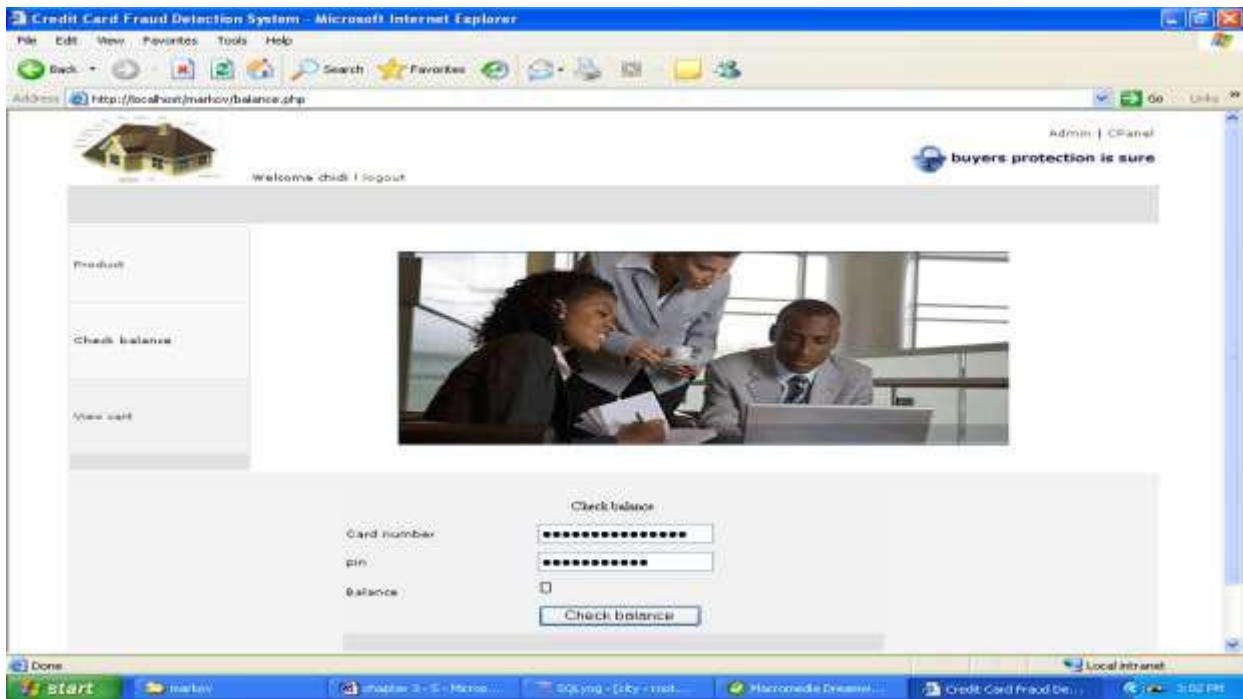
**Login Page**



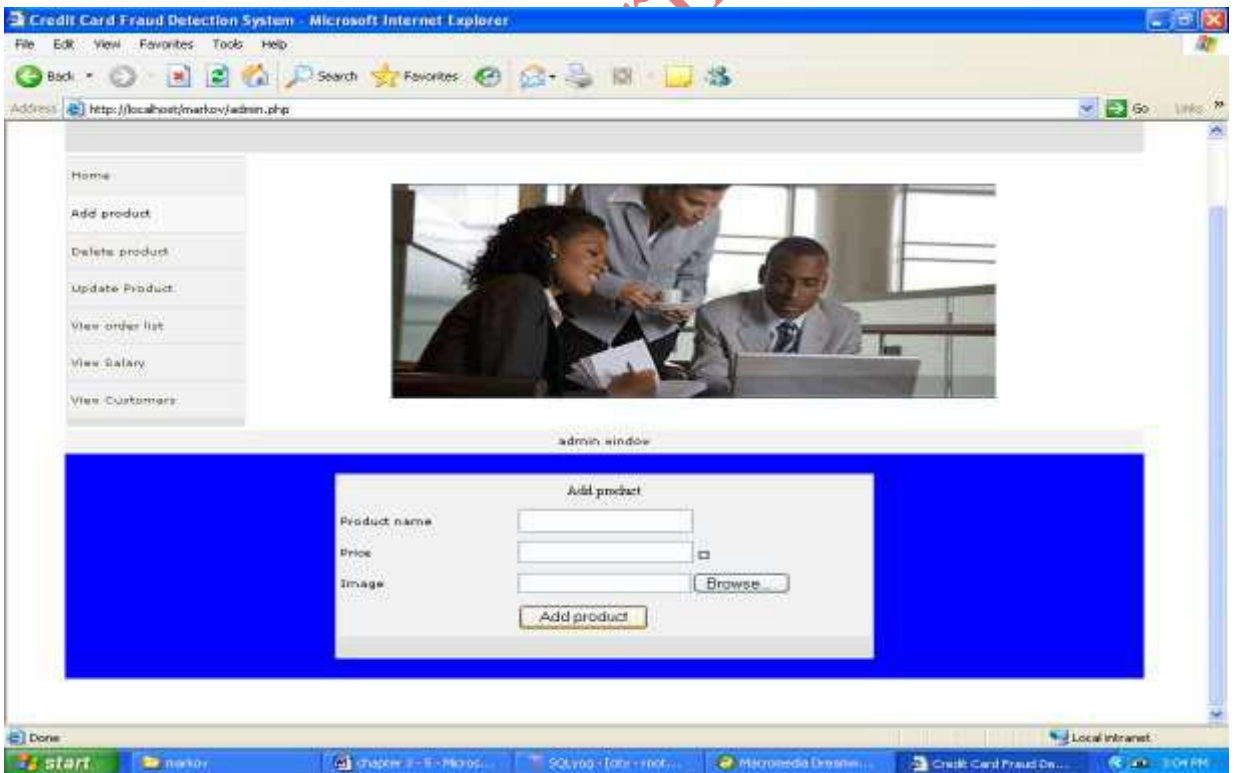
**Product Order Payment Form Using Credit Card**



**Credit Card Balance Checking Form**



**New Product Entry Form**



**Credit Card Account Opening Form**

Address: http://localhost/isekhov/ocart.php

Create admin  
Create credit card  
Fund credit card

CPanel window

Firstname  
Lastname  
Cardnumber: 5878891136611061  
Exp.-date (yyyy-mm-dd)  
Secret pin  
Signature: 66553 (if characters )  
Initial sum USD  
Age  
Real Name

Create

**Admin Password Creation Form**

Address: http://localhost/isekhov/cpanel.php

Home  
Create admin  
Create credit card  
Fund credit card

CPanel window

Admin account

Username  
Password  
Re-password

Create