

CREDIT CARD FRAUD DETECTION

Submitted in partial fulfillment of the requirements for
the award of
Bachelor of Engineering degree in Computer Science and Engineering

By



Jammula Ramya(37110621)

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI

SALAI, CHENNAI – 600 119

MARCH - 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project report is the bonafide work of **Jammula Ramya(Reg. No. 37110621)** who carried out the project entitled "**CREDIT CARD FRAUD DETECTION**" under my supervision from **November 2020** to **March 2021**.

Internal Guide
Dr.V.Nirmalrani, M.E.,Ph.D.,

Head of the Department

Submitted for Viva voce Examination held on_____

Internal Examiner

External Examiner

DECLARATION

I **Jammula Ramya** here by declare that the Project Report entitled “**CREDIT CARD FRAUD DETECTION**” is done by me and under the guidance of **Dr.V.Nirmalrani, M.E.,Ph.D.**, Department of Computer Science and Engineering at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala, M.E., Ph.D., Dean**, School of Computing, **Dr. S. Vigneswari, M.E., Ph.D., and Dr. L. Lakshmanan, M.E., Ph.D., Heads of the Department** of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide, **Dr.V.Nirmalrani, M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

The utilization of Visa is predominant in current society. However, clearly the quantity of Visa misrepresentation cases is continually expanding notwithstanding the chip cards overall coordination and existing insurance frameworks. This is the reason the issue of extortion recognition is vital at this point. The Visa extortion recognition highlights utilizes client conduct and area examining to check for bizarre examples. These examples incorporate client qualities, for example, client spending designs just as normal client geographic areas to confirm his personality. On the off chance that any surprising example is detected, the framework requires revivification. In this venture, a method for 'Charge card Fraud Detection' is created. As fraudsters are expanding step by step. Furthermore, deceptive exchanges are finished by the charge card and there are different kinds of misrepresentation. So to tackle this difficult blend of strategy is utilized like Genetic Algorithm, Behavior Based Technique and SET (Secure Electronic Transaction), Machine learning, Data Mining. By this exchange is tried independently and whatever suits the best is additionally continued. Furthermore, the first objective is to identify misrepresentation by separating the above methods to improve result. In this task the overall portrayal of the created extortion location framework and correlations between models based are (design recognition). In the last segment of this paper the consequences of evaluative testing and relating ends are thought of. At the point when an invalid user (fraud) utilizes a bank exchange, While exchange first bank authoriser check whether the client is legitimate client or an invalid client. Assuming the client is invalid, the bank authoriser block the exchange.

TABLE OF CONTENTS

	ABSTRACT	V
	LIST OF FIGURES	Viii
	LIST OF ABBREVATIONS	IX
CHAPTER No.	TITLE	PAGE No.
1.	INTRODUCTION	1
	1.1 PROBLEM DEFINITION	2
	1.2 GENETIC ALGORITHM	2
	1.2.1 PSEUDO CODE OF GENETICALGORITHM	2
2	LITERATURE REVIEW	3
3.	AIM AND SCOPE OF PRESENT INVESTIGATION	10
	3.2 SCOPE AND OBJECTIVE	10
	3.2.1 OBJECTIVE	10
	3.2.2 SCOPE	10
	3.3 SYSTEM REQUIREMENTS	11
	3.3.1 HARDWARE REQUIREMENTS	11
	3.3.2 SOFTWARE REQUIREMENTS	11
	3.4 LANGUAGE SPECIFICATION	11
	3.5 APPLICATION OF JAVA	11
	3.6 FEATURES OF JAVA	12
	3.6.1 SIMPLE	12
	3.6.2 OBJECT ORIENTED	12
	3.6.3 ROBUST	12
	3.6.4 PLATFORM INDEPENDENT	12
	3.6.5 SECURE	13
	3.6.6 MULTI TREADING	13
	3.6.7 ARCHITECTURAL NEUTRAL	13
	3.6.8 PORTABLE	13
	3.6.9 HIGH PERFORMANCE	14
	3.7 COLLECTION FRAMEWORK	14
	3.8 MYSQL	14
	3.9 FEASIBILITY STUDY	15

	3.9.1 ECONOMIC FEASIBILITY	16
	3.9.2 TECHNICAL FEASIBILITY	16
	3.9.3 SOCIAL FEASIBILITY	16
4.	EXPERIMENTAL OR MATERIAL METHODS	17
	4.1 PURPOSE	17
	4.2 DESIGN METHODOLOGY	17
	4.2.1 EXISTING SYSTEM	17
	4.2.1.1 DISADVANTAGES	18
	4.2.2 PROPOSED SYSTEM	18
	4.2.2.1 ADVANNTAGES	19
	4.3 SYSTEM DESIGN	19
	4.3.1 INPUT DESIGN	19
	4.3.2 OUTPUT DESIGN	19
	4.4 DATAFLOW DIAGRAM	20
	4.4.1 ROLE OF DFD	20
	4.5 MODULE DESCRIPTION	25
	4.5.1 MODULES	25
	4.5.1.1 USER GUI	25
	4.5.1.2 CRITICAL VALUE IDENTIFICATION	25
	4.5.1.3 FRAUDDETECTION USING	25
	GENERIC ALGORITHM	
	4.6 SYSTEM ARCHITECTURE	25
	4.6.1 PROBLEM STATEMENT	26
	4.7 SYSTEM TESTING	27
	4.7.1 TEST PATTERN	27
	4.7.2 VERIFICATION	27
	4.7.3 VALIDATION	27
	4.7.4 BASICS OF SOFTWARETESTING	27
	4.7.4.1 BLACKBOX TESTING	27
	4.7.4.2 WHITEBOX TESTING	27
	4.8 TYPES OF TESTING	28
	4.8.1 UNIT	28
	4.8.2 INTEGRATION	28
	4.8.3 FUNCTIONAL	28
	4.8.4 SYSTEM	29
	4.8.5 STRESS	29
	4.8.6 PERFORMANCE	29
	4.8.7 USABILITY	29
	4.8.8 ACCEPTANCE	29
	4.8.9 REGRESSION	29
5.	RESULTS AND DISCUSSIONS	30
6.	CONCLUSIONS & FUTURE ENHANCEMENT	40
	6.1 CONCLUSION	40
	6.2 FUTURE ENHANCEMENT	40
	REFERENCES	41
	APPENDIX	

A.PAPER ACCEPTANCE MAIL
B.PLAGARISM REPORT
C.JOURNAL PAPER
D.SOURCE CODE

LIST OF FIGURES

FIGURE No.	FIGURE NAME	PAGE No.
3.6.4	PLATFORM INDEPENDENT	13
3.7	COLLECTION FRAMEWORK	14
4.4.1	ROLE OF DATAFLOW DIAGRAM	21
4.4.2	DATAFLOW DIAGRAM 1	22
4.4.3	DATAFLOW DIAGRAM 2	23
4.4.4	DATAFLOW DIAGRAM 3	24
4.6	SYSTEM ARCHITECTURE	26
5.1.1	HOME PAGE	30
5.1.2	ADMIN PAGE	31
5.1.3	ADMIN BALANCE	32
5.1.4	DEPOSIT PAGE	33
5.1.5	AVAILABLE BALANCE	34
5.1.6	PAYMENT TRANSACTION	35
5.1.7	ADMIN QUESTION PAGE	36
5.1.8	CHECK VALID USER	37
5.1.9	PAYMENT TRANSACTION HISTORY	38
5.1.10	DATASTORED IN DATABASE SERVER	39

LIST OF ABBREVIATIONS

ABBREVIATIONS

DEFINITIONS

SET

SECURE ELECTRONIC TRANSACTION

ML

MACHINE LEARNING

GA

GENERIC ALGORITHM

CHAPTER 1

INTRODUCTION

As of late, the predominant information mining concerns individuals with Mastercard misrepresentation identification model dependent on information mining. Since our concern is drawn nearer as a grouping issue, traditional information mining calculations are not straightforwardly material. So an elective methodology is made by utilizing broadly useful meta heuristic methodologies like hereditary calculations.

This undertaking is to propose a charge card extortion recognition framework utilizing hereditary calculation. Hereditary calculations are developmental calculations which target getting better arrangements as time advances. At the point when a card is duplicated or taken or lost and caught by fraudsters it is normally utilized until its accessible cutoff is drained. Accordingly, as opposed to the quantity of effectively characterized exchanges, an answer which limits the complete accessible cutoff on cards subject to misrepresentation is more conspicuous. It points in limiting the bogus alarms utilizing hereditary calculation where a bunch of span esteemed boundaries are improved.

1.1 PROBLEM DEFINITION

To build up a Visa extortion discovery framework utilizing hereditary calculation. During the Visa exchange, the extortion is distinguished and the quantity of bogus alarm is being limited by utilizing hereditary calculation. Rather than boosting the quantities of accurately characterized exchanges we characterized a target work where the misclassification costs are variable and along these lines, right grouping of certain exchanges are a higher priority than effectively ordering the others. The calculation starts with multi-populace of arbitrarily created chromosomes. These chromosomes go through the activities of choice, hybrid and change. Hybrid consolidates the data from two parent

chromosomes to deliver new people, abusing the best of the current age, while transformation or haphazardly changing a portion of

the boundaries permits investigation into different locales of the arrangement space. Regular choice through a difficult explicit expense work guarantees that simply the best fit chromosomes stay in the populace to mate and create the future. Upon emphasis, the hereditary calculation meets to a worldwide arrangement

1.2 GENETIC ALGORITHM

Hereditary calculations are developmental calculations which target getting better arrangements as time advances. Since their first presentation by Holland, they have been effectively applied to numerous difficult areas from stargazing to sports, from advancement to software engineering, and so on They have additionally been utilized in information digging primarily for variable determination and are for the most part combined with other information mining calculations. In this examination, we attempt to tackle our order issue by utilizing just a hereditary calculation arrangement.

1.2.1 Pseudo code of genetic algorithm

Instate the populace

Assess starting populace

Rehash

Perform serious determination

Apply hereditary administrators to produce new arrangements

Assess arrangements in the populace

Until some intermingling measures is fulfilled

CHAPTER 2

LITERATURE REVIEW

2.1 "A Survey of credit and conduct scoring: determining: monetary danger of loaning to clients"

AUTHOR: L. C. Thomas (2000):-

Credit scoring and conduct scoring are the strategies that assist associations with choosing whether or not to concede Visa to shoppers who apply to them. This article studies the procedures utilized both measurable and operational exploration based to help these choices. It additionally examines the need to consolidate monetary conditions into the scoring frameworks and the manner in which the frameworks could change from assessing the likelihood of a shopper will bring to the loaning association two of the significant advancements being endeavored nearby. It brings up how fruitful has been this under-investigated territory of estimating monetary danger. 2000 Elsevier Science B.V. Protected by copyright law

2.2 "Credit Scoring Methods. Czech Journal of Economics and Finance"

AUTHOR: M. Vojtek and E. Kocenda (2006):-

The paper audits the best-created and most as often as possible applied techniques for credit scoring utilized by business banks when assessing advance applications. The creators focus on retail advances - applied exploration in this section is restricted, however there has been a sharp expansion in the volumes of advances to retail

customers lately. Login examination is distinguished as the most incessant credit-scoring strategy utilized by banks. In any case, other non parametric strategies are far reaching regarding design acknowledgment. The strategies assessed have potential for application in post-progress nations.

2.3 "Continuous Visa extortion identification utilizing computational knowledge"

AUTHOR: Jon T.S.Quah*, M. siganesh (2007):-

Internet banking and web based business have been encountering quick development in the course of recent years and show enormous guarantee of development even later on. This has made it simpler for fraudsters to enjoy new and esoteric methods of submitting Mastercard extortion over the web. This paper centers around ongoing misrepresentation identification and presents another and imaginative methodology in understanding spending examples to translate potential extortion cases. It utilizes self-association guide to translate, channel and break down client conduct for location of extortion. 2007 Elsevier Ltd. Protected by copyright law. Catchphrase: self-arranging map; unaided learning; Risk scoring; Transaction.

2.4 "Identification A Novel and Successful Credit Card Fraud System Implemented in a Turkish Bank"

AUTHOR: Ekrem Duman, Ayse Buyukkaya, Ilker Elikucuk (2008):-

We built up a charge card misrepresentation location answer for a significant bank in Turkey. The examination was finished in around three years and of the created framework has been being used since February 2013. It had an extraordinary effect in the standard based misrepresentation location measure utilized by the bank. Undoubtedly, while a lot of the guidelines have been dispensed with and the number has been diminished to a large portion of, a huge expansion in misrepresentation identification has been recorded.

2.5 "Perceiving charge card blackmail by genetic estimation and scatter search "

AUTHOR: Ekrem Duman a,M.Hamdi Ozcelik b (2011):-

In this assessment we develop a methodology which improves a charge card distortion area plan at present being used in a bank. With this course of action each trade is scored and subject to these scores the trades are appointed tricky or authentic. In coercion area courses of action the customary objective is to restrict the wrongly assembled number of trades. At any rate when in doubt, wrong gathering of each trade don't have the very effect in that if the hand of fraudsters its whole open cutoff is spent.

2.6 "Industry Paper: A Prototype for Credit Card Fraud Management"

AUTHOR: AlexanderArtikis, NikosKatzouris, NCSR Demokritos, AnthensGreece(2017):-

To forestall issues and exploit openings before they even happen, the examination project speed proposed a system, and built up a model for proactive occasion driven choice making.We present the utilization of this philosophy to Master card misrepresentation the executives. The AI segment of the speed model backings the online development of extortion designs, permitting it to adjust to the persistently changing misrepresentation types.Moreover, the UI of the model empowers extortion investigation to make the most out of the aftereffects of computerization and in this manner arrive at educated decisions.Unlike most scholarly exploration on charge card extortion the board, the appraisal of the model depends on agent exchange informational indexes.

2.7 "An Information Digging Based Framework for Charge Card Extortion Recognition in e-tail"

AUTHOR: Nuno Carneiro, Goncalo Figueira, Miguel Costa(2017):-

The paper investigates the mix of manual and programmed order, gives bits of knowledge into the total improvement cycle and thinks about various AI strategies. The

paper would thus be able to help analysts and professionals to plan and execute information digging based frameworks for extortion discovery or comparative issues.

2.8 “Credit Card Fraud Detection System Using Intelligent Agents and Enhanced Security Features”

AUTHOR: Amanze,B.C.,Asogwa,D.C.&Chukwuneke,C.I (2018):-

Visa extortion can be recognized utilizing shrewd specialists during exchanges. Keen Agents helps to get a high misrepresentation exchange inclusion joined with low bogus alert rate, thus giving a superior and advantageous approach to identify fakes. Utilizing insightful agent, customers design is dissected and any deviation from the standard example is viewed as false exchange. In this paper, the canny specialist is

utilized to distinguish the extortion when exchange is in progress. The current misrepresentation discovery strategies are not able to identify the extortion when exchange is in progress. As the utilization of Mastercard has expanded the Mastercard misrepresentation has additionally expanded significantly.

2.9 “Charge Card Misrepresentation Recognition Utilizing nearby anomaly Factor and Detachment woods”

AUTHOR: S Joel Franklin(2019):-

Misrepresentation distinguishing proof is a essential issue confronting enormous monetary establishments, which has caused because of the ascent in charge card installments. This paper brings another methodology for the prescient recognizable proof of charge card installment cheats zeroed in on Isolation Forest and Local Outlier Factor. The proposed arrangements contains the relating stages: pre -preparing of informational collections, preparing and arranging, assembly of choices and investigation of tests. In this article, the conduct attributes of right and inaccurate exchanges are to be instructed by two sorts of calculations neighborhood exception factor and disconnection backwoods

2.10 “HOBA: A tale include designing approach for Mastercard extortion location with a profound learning engineering”

AUTHOR:Xinwei Zhang, Yaoci Han, Wei Xu, Qili Wang(2019):-

The primary commitment of our work is the improvement of an extortion identification framework that utilizes a profound learning design along with a high level component designing cycle dependent on homogeneity - situated conduct examination(HOBA). In view of a genuine data set from one of the biggest banks in china, we lead a similar report to survey the viability of the proposed structure

CHAPTER 3

AIM AND SCOPE OF THE PRESENT INVESTIGATION

3.1 AIM OF THE PROJECT:

The Main Aim of the project to build up an application which has capacity to confine and obstruct the exchange performing by the aggressor from authentic client's charge card details.This framework attempts to distinguish fraudulent exchange before exchange

succed.It points in limiting the bogus cautions utilizing hereditary calculation where a bunch of span esteemed boundaries are enhanced

3.2 SCOPE AND OBJECTIVE:

3.2.1 OBJECTIVE

The Main Objective of this project is to execute Machine learning methods like Generic calculation, Behavior Based Technique and Secure Electronic Transaction to identify credit misrepresentation location.

3.2.2 SCOPE

The Scope of the project is as follows:

The scope of this SRS document continues for the whole life pattern of the project.This report characterizes the last condition of the product necessities settled upon by the clients and designers.Finally toward the finish of the task excecution every one of the functionalities might be recognizable from the SRS to the product.The archive depicts the functionality,performance,constraints,interface and unwavering quality for the whole life pattern of the venture

3.3 SYSTEM REQUIREMENTS

3.3.1 HARDWARE REQUIRMENTS

System	:	Pentium IV 2.4GHZ
HardDisk	:	40 Gb
Ram	:	512 Mb

3.3.2 SOFTWARE REQUIRMENTS

Operating system : Window 7(32 bit)

Coding Language : java

Tool : Netbeans

3.4 LANGUAGE SPECIFICATION

Java is one of the world's generally significant and broadly utilized codes, and it has held this differentiation for a long time. Not at all like some other codes whose impact has weared with entry of time, while Java's has developed.

3.5 APPLICATION OF JAVA

Java is generally utilized in each edge of world and of human existence. Java isn't just utilized in virtual products but at the same time is broadly utilized in planning equipment controlling programming parts. There are in excess of 930 million JRE downloads every year and 3 billion cell phones run java. Following are some other usage of Java:

1. Developing Desktop Applications
2. Web Applications like Linkedin.com, Snapdeal.com etc
3. Mobile Operating System like Android
4. Embedded Systems
5. Robotics and games etc.

3.6 FEATURES OF JAVA

The excellent explanation for formation of Java was to bring transportability and security include into a scripting language. Close to these two significant highlights,

there were numerous different highlights that assumed a significant part in embellishment out the last type of this extraordinary language. Those highlights are;;

3.6.1 Simple

Java is not difficult to learn and its punctuation is very basic, perfect and straightforward. The confounding and questionable ideas of C++ are either forgotten about in Java or they have been re-executed in a cleaner way. Eg: Pointers and Operator Overloading are not there in java yet were a significant piece of C++.

3.6.2 Object Oriented

In java everything is Object which has some data and behaviour. Java can be easily extended as it is based on Object Model.

3.6.3 Robust

Java puts forth an attempt to take out blunder inclined codes by underlining mostly on order time mistake checking and runtime checking. Yet, the primary territories which Java improved were Memory Management and misused Exceptions by presenting programmed Garbage Collector and Exception Handling

3.6.4 Platform Independent

Dissimilar to other programming dialects, for example, C, C++ and so on which are incorporated into stage explicit machines. Java is destined to be compose once, run-anyplace language.

On aggregation Java program is incorporated into byte code. This byte code is stage autonomous and can be run on any machine, in addition to this byte code design additionally give security. Any machine with Java Runtime Environment can run Java Programs

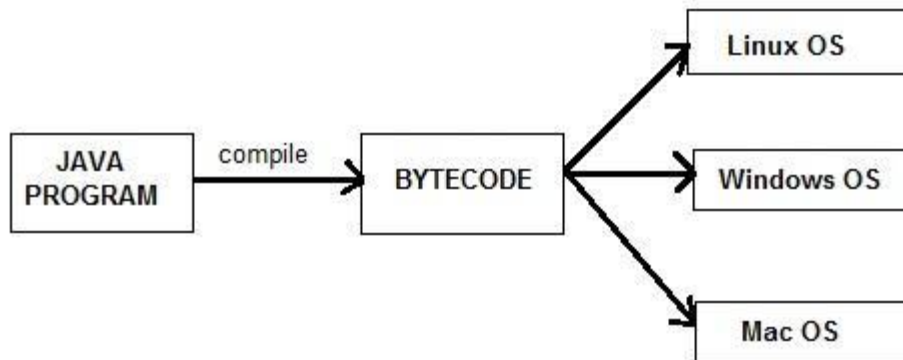


Fig.3.6.4 Platform Independent

3.6.5 Secure

With regards to security, Java is consistently the best option. With java secure highlights it empower us to create infection free, temper free framework. Java program consistently runs in Java runtime climate with practically invalid collaboration with framework OS, consequently it is safer.

3.6.6 Multi-Threading

Java multithreading highlight makes it conceivable to compose program that can do numerous assignments all the while. Advantage of multithreading is that it uses same memory and different assets to execute numerous strings simultaneously, as While composing, syntactic mistakes are checked along.

3.6.7 Architectural Neutral

Compiler produces byte codes, which have nothing to do with a specific PC engineering, consequently a Java program is not difficult to decipher on any machine.

3.6.8 Portable

Java Byte code can be conveyed to any stage. No execution subordinate highlights. Everything identified with capacity is predefined, model: size of crude information types

3.6.9 High Performance

Java is a deciphered language, so it won't ever be pretty much as quick as an incorporated language like C or C++. Be that as it may, Java empowers superior with the utilization of in the nick of time compiler.

3.7 COLLECTION FRAMEWORK

Assortment system was not piece of unique Java discharge. Assortments was added to J2SE 1.2. Preceding Java 2, Java gave adhoc classes like Dictionary,

Vector, Stack and Properties to store and control gatherings of items. Assortment structure gives numerous significant classes and interfaces to gather and sort out gathering of the same items

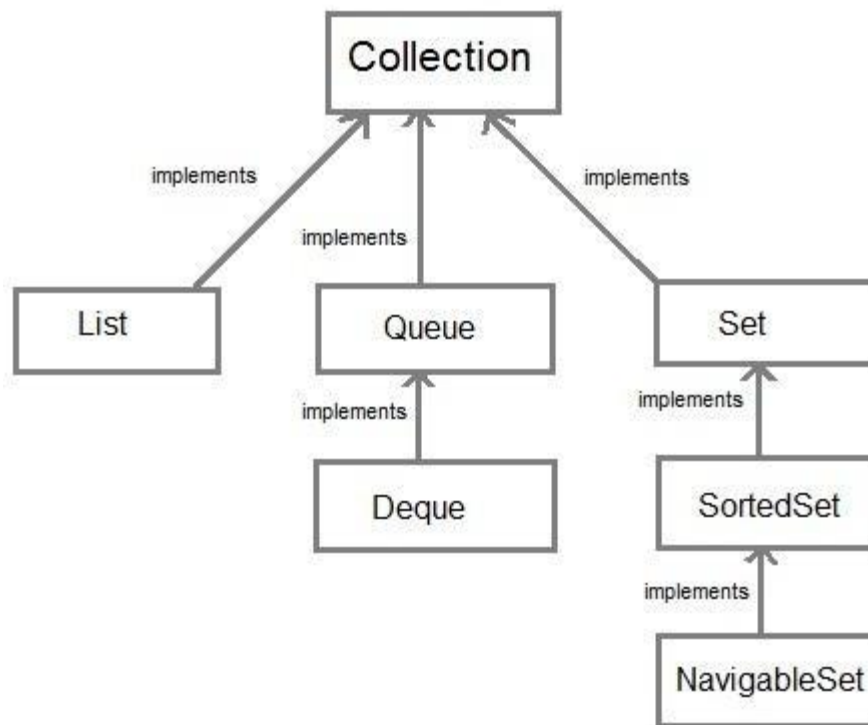


Fig.3.7 COLLECTION FRAMEWORK

3.8 MYSQL

MySQL, authoritatively, yet additionally called "My Sequel" is the world's most broadly utilized open-source social data set administration framework (RDBMS) that runs as a worker giving multi-client admittance to various information bases, however SQLite presumably has more complete implanted arrangements. The SQL expression represents Structured Query Language.

The MySQL improvement project has made its source code accessible under the provisions of the GNU General Public License, just as under an assortment of restrictive arrangements. MySQL was possessed and supported by a solitary revenue driven firm, the Swedish organization MySQL AB, presently claimed by Oracle

Corporation. MySQL is a famous decision of data set for use in web applications, and is a focal segment of the broadly utilized LAMP open source web application programming stack (and other 'AMP' stacks).

Light is an abbreviation for "Linux, Apache, MySQL, Perl/PHP/Python." Freesoftware-open source projects that require a full-highlighted information base administration framework regularly use MySQL.

For business use, a few paid releases are accessible, and offer extra usefulness. Applications which use MySQL data sets include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other programming. MySQL is likewise utilized in some prominent, huge scope sites, including Wikipedia, Google (however not for look), Facebook, Twitter, Flickr and YouTube.

3.9 FEASIBILITY STUDY

The attainability of the undertaking is investigated in this stage and strategic agreement is advanced with an extremely broad arrangement for the task and some quotes. During framework investigation the attainability investigation of the proposed framework is to be completed. This is to guarantee that the proposed framework isn't a weight to the organization. For practicality examination, some comprehension of the significant prerequisites for the framework is fundamental.

The practicality study examines the issue and the data needs of the partners. It looks to decide the assets needed to give a data frameworks arrangement, the expense and advantages of such an answer, and the plausibility of such an answer.

The objective of the achievability study is to consider elective data frameworks arrangements, assess their possibility, and propose the option generally appropriate to the association. The plausibility of a proposed arrangement is assessed regarding its segments.

3.9.1 ECONOMICAL FEASIBILITY

This examination is completed to check the financial effect that the framework will have on the association. The measure of asset that the organization can fill the innovative work of the framework is restricted. The uses must advocated. Along these lines the created framework also inside the spending plan and this was accomplished in light of the fact that the majority of the innovations utilized are uninhibitedly accessible. Just the tweaked items must be bought.

3.9.2 TECHNICAL FEASIBILITY

This examination is done to check the specialized attainability, that is, the specialized prerequisites of the framework. Any framework created should not have an appeal on the accessible specialized assets. This will prompt high requests on the accessible specialized assets. This will prompt high requests being set on the customer. The created framework should have an unobtrusive prerequisite, as just negligible or invalid changes are needed for executing this framework.

3.9.3 SOCIAL FEASIBILITY

The part of study is to check the degree of acknowledgment of the framework by the client. This incorporates the way toward preparing the client to utilize the framework proficiently. The client should not feel undermined by the framework, rather should acknowledge it as a need.

CHAPTER 4

EXPERIMENTAL OR MATERIAL METHODS

4.1 PURPOSE

The motivation behind this record is to characterize the necessities of charge card misrepresentation recognition. In detail, this report will give an overall depiction of our task, including client necessities, item point of view, and outline of prerequisites, general limitations. Furthermore, it will likewise give the particular prerequisites and usefulness required for this undertaking - like interface, utilitarian necessities and execution necessities.

4.2 DESIGN METHODOLOGY

4.2.1 Existing System

- In In instance of Mastercard extortion identification the current framework is identify the a great many fakes has been occur. Existing framework keep up the enormous measure of information when client comes to think about irregularity in exchange he/she submitted question and afterward misrepresentation discovery framework start it working. It first attempts to recognize that misrepresentation has really happen after that it exchanges that was utilized to extortion identification component created by expert and visa cards
- Intrusion identifications to follow extortion area, etc. In the event of existing framework there is no affirmation of recuperation of misrepresentation and Customer fulfillment.
- Secure electronic framework used to examine the conduct of real clients.
- Data Mining instruments to characterize and preprocess the clients information.
- . AI worldview arrangement, with Credit Card Fraud Detection being the base

- Genetic algorithms

4.2.1.1 Disadvantages Of Existing System

- Each installment framework has its cutoff points in regards to the most extreme sum in the record, the quantity of exchanges each day and the measure of yield.
- If Internet association comes up short, you can not get to your online record.
- If you observe the security administers the danger is insignificant. The more awful circumstance when the arrangement of handling organization has been broken, in light of the fact that it prompts the hole of individual information on cards and its proprietors.
- The data pretty much every one of the exchanges, including the sum, time and beneficiary are put away in the data set of the installment framework. Furthermore, it implies the insight office has an admittance to this data. Now and again this is the way for false exercises.

4.2.2 Proposed System

- The aim of the proposed system is to develop a application. which has capability to restrict and block the transaction performing by attacker from genuine user's credit card details.
- As we seen the existing system detects the fraud after fraud has been occurred . The proposed system tries to detect fraudulent transaction before transaction succeed.
- In proposed system we are using pattern recongnization, which works on transaction behavior of user. By Using pattern recongnization, after certain transactions we find one threshold value by using this threshold value we can compare current transaction with threshold value if transaction is quite different

from user behavior then check whether it is genuine or fraud OTP (full form) and security questions are used.

- To get security from hackers we are providing encryption at registration time for password this encryption is done by Secure Hash Algorithm (SHA) algorithm, other encryption algorithms.

4.2.2.1 Advantages Of Proposed System

- Online cautions are accessible if dubious movement is distinguish on your card. The backer will advise if any bizarre charges come through on the card.
- Credit scores are utilized from various perspectives and when an individual uses a Visa capably and makes opportune installments without over limits or late installments, their financial assessment rises.
- By making installments on schedule and keeping the equilibrium low on the card, an individual will get a good deal on the measure of interest charged by the card backer.
- Eco friendly, no individual presence is required, it treats with developing technology, time framework

4.3 SYSTEM DESIGN

4.3.1 Input Design

The information configuration is the connection between the data framework and the client. It contains the creating detail and methodology for information planning and those means are important to place exchange information in to a usable structure for preparing can be accomplished by examining the PC to peruse information from a

composed or printed record or it can happen by having individuals entering the information straightforwardly into the framework. The plan of info centers around controlling the measure of information required, controlling the mistakes, staying away from delay, dodging additional means and keeping the cycle basic. The information is planned in such a manner so it furnishes security and usability with holding the protection. Info Design thought about the accompanying things:

- What information ought to be given as information?
- How the information ought to be masterminded or coded?
- The discourse to direct the working faculty in giving information.
- Methods for planning input approvals and step

4.3.2 Output Design

A quality yield is one, which meets the prerequisites of the end client and presents the data plainly. In any framework consequences of handling are imparted to the clients and to other framework through yields. In yield plan it is resolved how the data is to be uprooted for sure fire need and furthermore the printed copy yield. It is the most significant and direct source data to the client. Productive and clever yield configuration improves the framework's relationship to help client dynamic.

The yield type of a data framework ought to achieve at least one of the accompanying targets.

- Convey data about past exercises, current status or projections of the
- Signal significant occasions, openings, issues, or alerts.
- Trigger an activity.
- Confirm an activity
- Future

4.4 DATA FLOW DIAGRAM

Data Information Flow Diagram (DFD) is a two-dimensional outline that portrays how information is handled and communicated in a framework. The graphical portrayal perceives each wellspring of information and how it communicates with other information sources to arrive at a common yield. To draft an information stream graph one must

- Identify outer data sources and yields
- Determine how the data sources and yields identify with one another
- Explain with designs how these associations relate and what they result in.

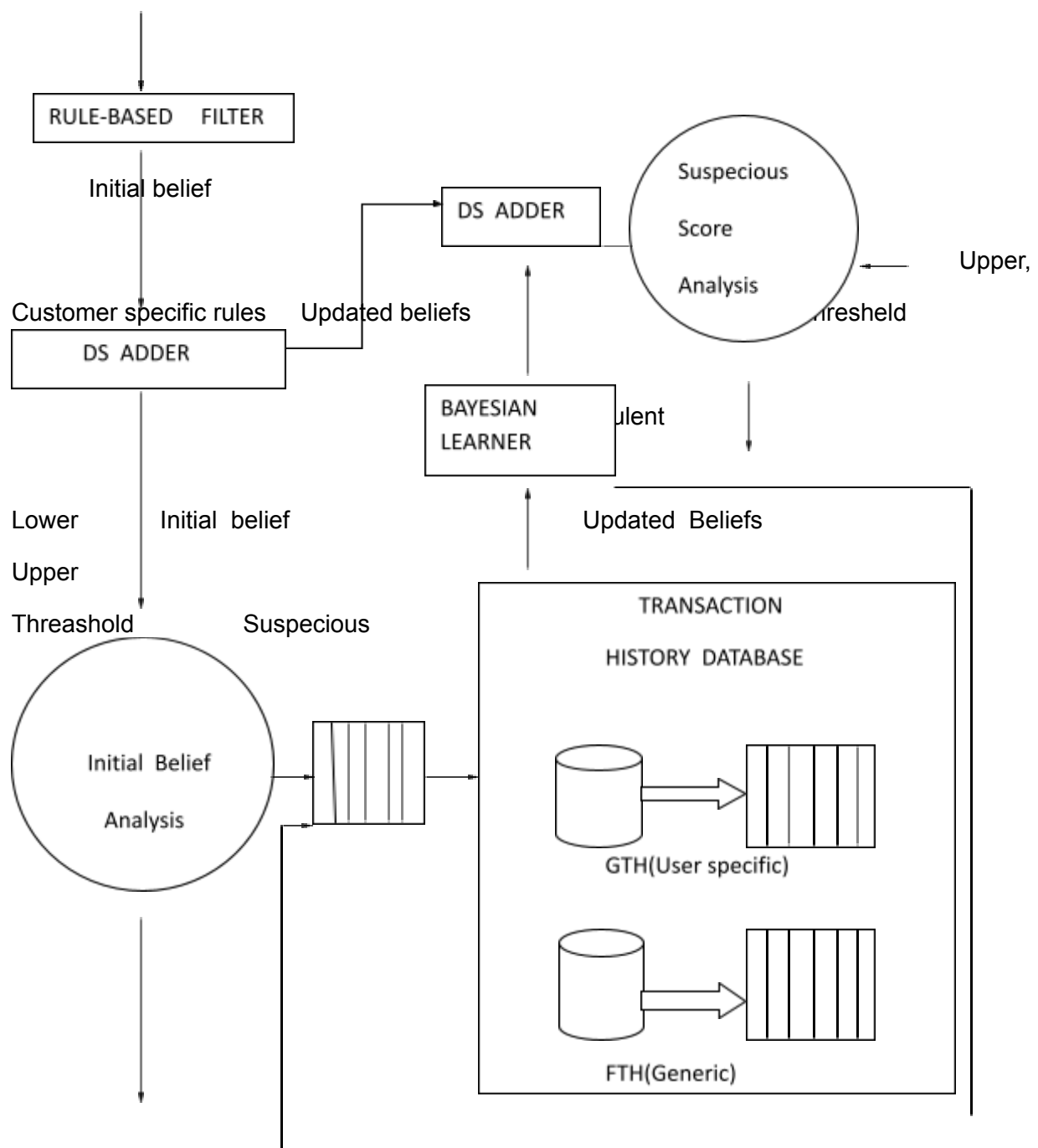
4.4.1 Role of DFD:

- It is a documentation support which is understood by both programmers and nonprogrammers. As DFD postulates only what processes are accomplished not how they are performed.

- A physical DFD postulates where the data flows and who processes the data.

- It permits analyst to isolate areas of interest in the organization and study them by examining the data that enter the process and viewing how they are altered

Incoming Transaction

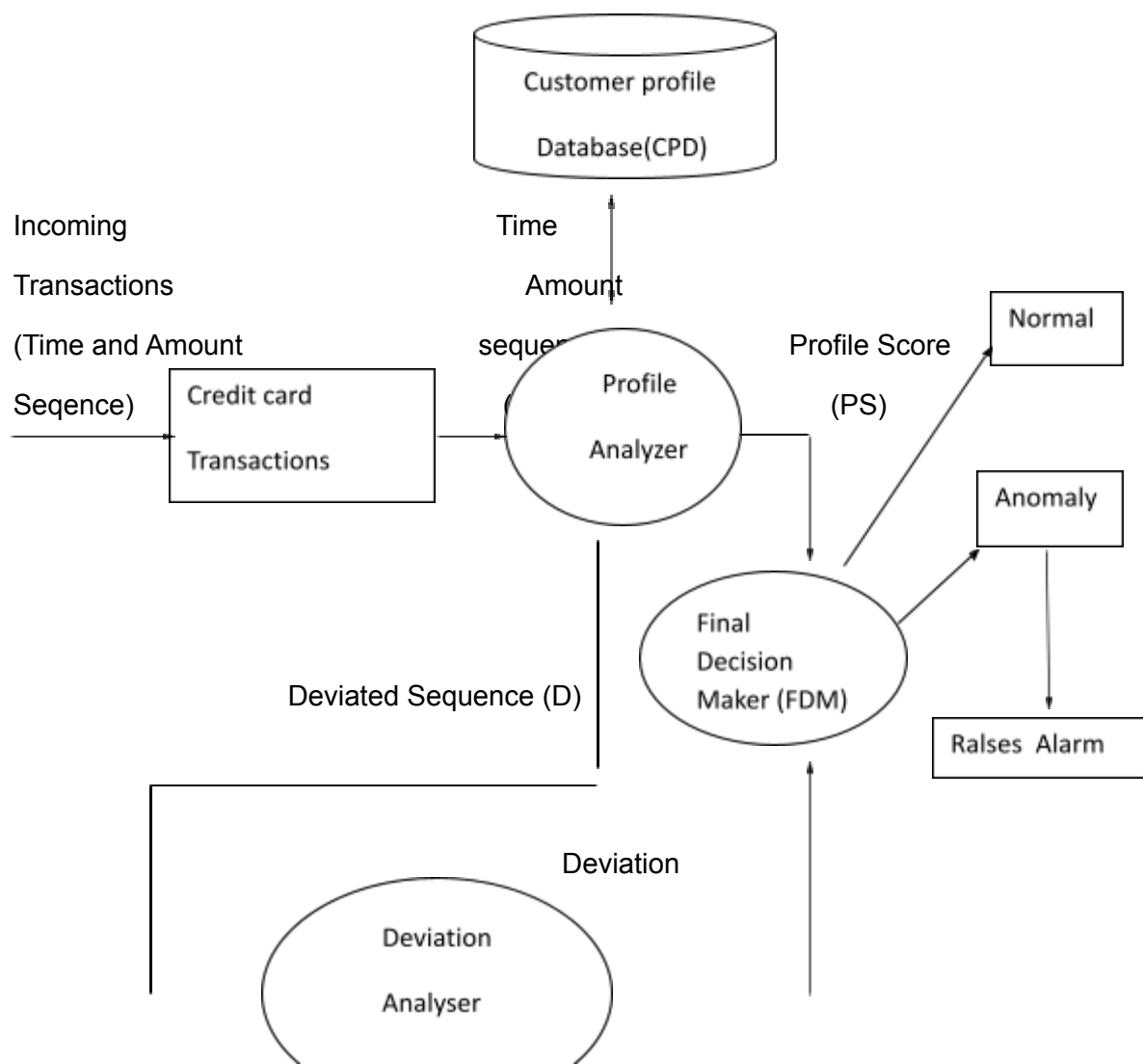


Genuine/Fraudulent

Suspicion score (Current Round)

Fig.4.4.1 Role of DFD:

4.4.2 DFD-1



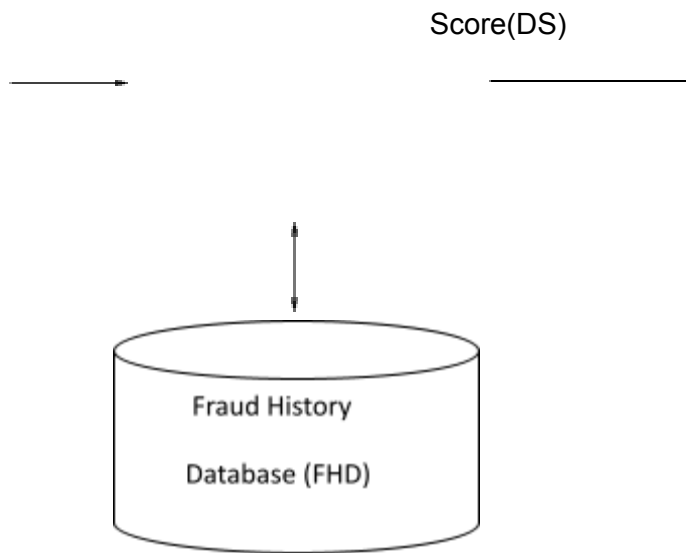


Fig.4.4.2 Data Flow Diagram

4.4.3 DFD-2

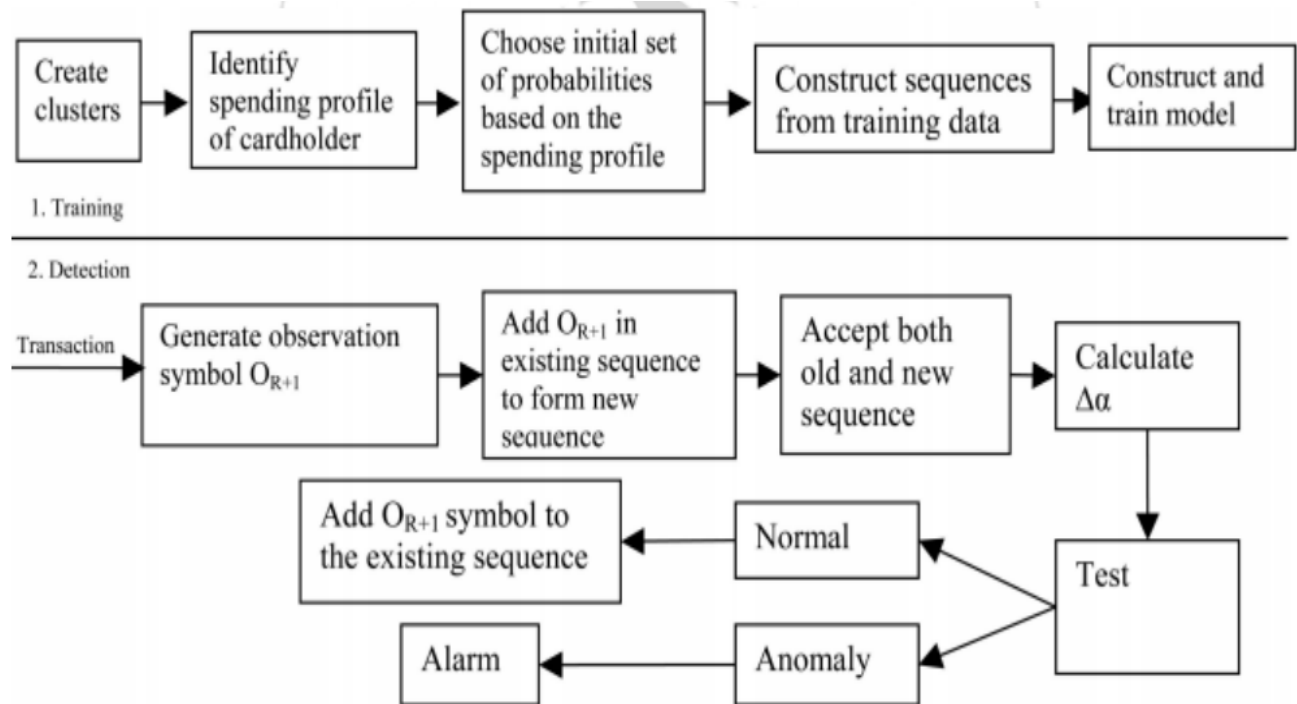


Fig.4.4.3 Data Flow Diagram-2

4.4.4 DFD-3

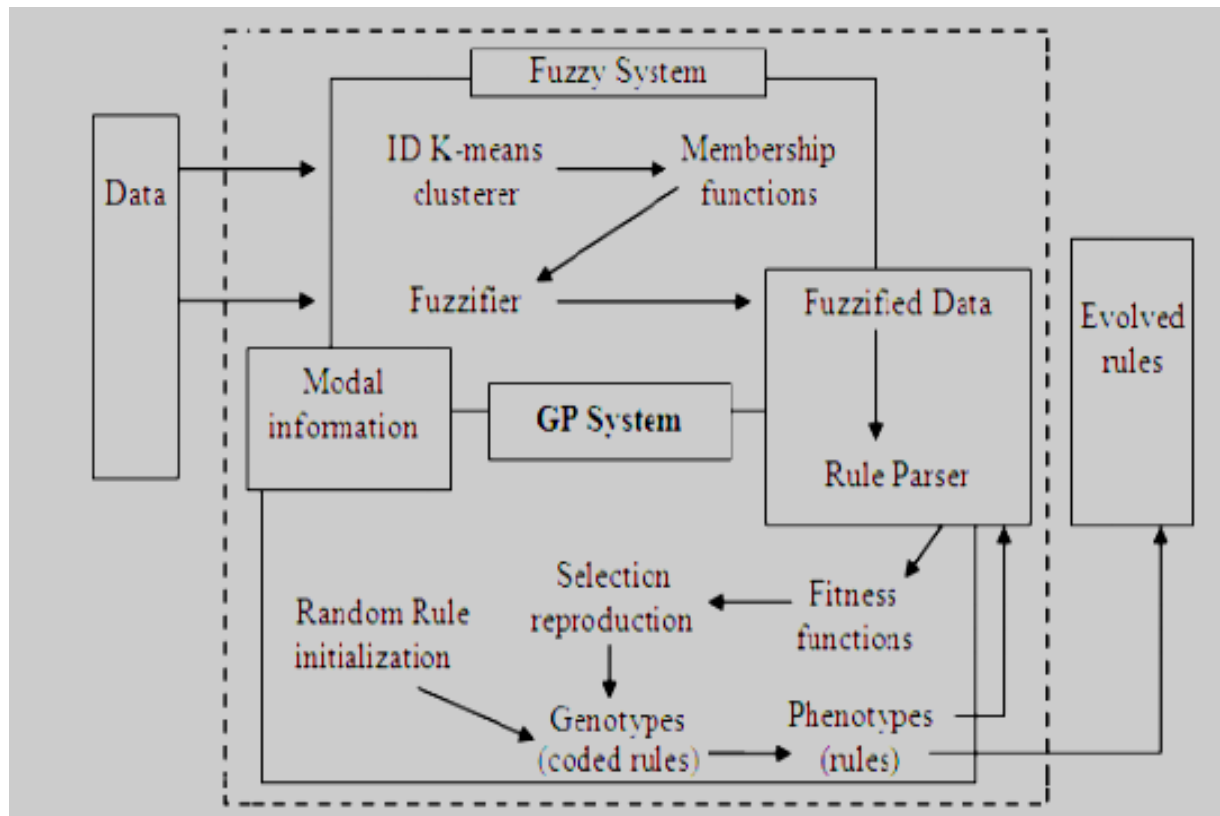


Fig.4.4.4 Data Flow Diagram-3

4.5 MODULES DESCRIPTION

4.5.1 MODULES

- User GUI
- Critical Value Identification
- Fraud Detection using Genetic Algorithm

4.5.1.1 User GUI

In this module, User Interface module is created utilizing Applet Viewer. This module is created to client to recognize the Mastercard misrepresentation utilizing hereditary calculation method. So the UI should be fit for giving the client to transfer the

dataset and make controls lastly should show the client if misrepresentation has been identified. Just last yield will be in applet screen.

4.5.1.2 Critical Value Identification

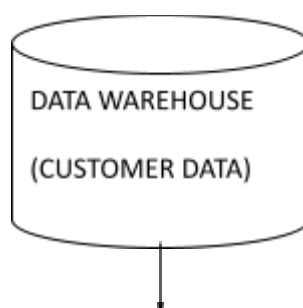
- Based on CC usage Frequency
- Based on CC usage Location
- Based on CC OverDraft
- Based on CC Book Balance
- Based on CC Average Daily Spending

4.5.1.3 Fraud Detection using Generic Algorithm

In this module the system should recognize whether any extortion has been happened in the exchange or not. It should likewise show the client about the outcome.

4.6 SYSTEM ARCHITECTURE

Depicting the general highlights of the product is worried about characterizing the prerequisites and setting up the undeniable level of the framework. During building plan, the different website pages and their interconnections are recognized and planned. The significant programming parts are distinguished and decayed into preparing modules and conceptual information structures and the interconnections among the modules are recognized. The accompanying modules are recognized in the proposed framework..



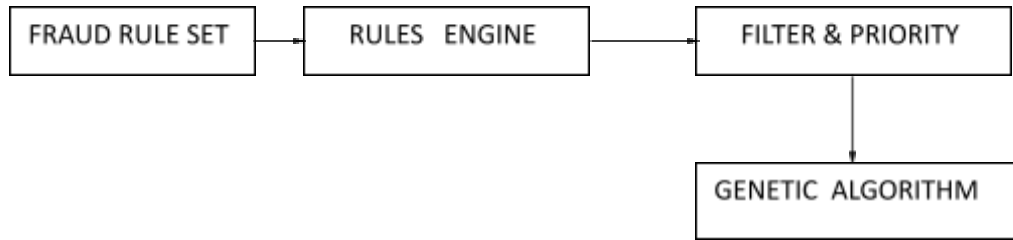


Fig.4.6 SYSTEM ARCHITECTURE

The above architecture describes the work structure of the system.

- The client information in the information distribution center is exposed to the principles motor which comprises of the extortion rule set.
- The channel and need module sets the need for the information and afterward sends it to the hereditary calculation which plays out its capacities and produces the yield.

4.6.1 Problem Statement:

The Credit Card Fraud Detection Problem incorporates displaying past Visa exchanges with the information on the ones that ended up being misrepresentation. This model is then used to distinguish if another exchange is false. Our point here is to identify 100% of the deceitful exchanges while limiting the mistaken misrepresentation arrangements.

4.7 SYSTEM TESTING

4.7.1 Test plan

Programming testing is the cycle of assessment a product thing to recognize contrasts between given info and anticipated yield. Likewise to survey the component of a product thing. Testing evaluates the nature of the item. Programming testing is an interaction that ought to be finished during the advancement cycle. At the end of the day programming testing is a confirmation and approval measure.

4.7.2 Verification

Check is the cycle to ensure the item fulfills the conditions forced toward the beginning of the improvement stage. At the end of the day, to ensure the item carries on the manner in which we need it to.

4.7.3 Validation

Approval is the cycle to ensure the item fulfills the predetermined prerequisites toward the finish of the improvement stage. At the end of the day, to ensure the item is worked according to client prerequisites.

4.7.4 Basics of programming testing

There are two rudiments of programming testing: discovery testing and white box testing.

4.7.4.1 Black box Testing

Discovery testing is a trying procedure that overlooks the inward component of the framework and spotlights on the yield produced against any information and execution of the framework. It is likewise called useful testing.

4.7.4.2 White box Testing

White box testing is a trying strategy that considers the inner component of a framework. It is likewise called underlying testing and glass box testing. Dark box testing is frequently utilized for approval and white box testing is regularly utilized for check.

4.8 Types of testing

There are many types of testing like

- Unit Testing
- Integration Testing
- Functional Testing
- System Testing
- Stress Testing

- Performance Testing
- Usability Testing
- Acceptance Testing
- Regression Testing
- Beta Testing

4.8.1 Unit Testing

Unit testing is the trying of an individual unit or gathering of related units. It falls under the class of white box testing. It is regularly done by the developer to test that the unit he/she has carried out is creating anticipated yield against given information.

4.8.2 Integration Testing

Combination testing will be trying in which a gathering of parts are joined to deliver yield. Additionally, the collaboration among programming and equipment is tried in incorporation testing if programming and equipment segments have any connection. It might fall under both white box testing and discovery testing.

4.8.3 Functional Testing

Utilitarian testing is the trying to guarantee that the predefined usefulness needed in the framework prerequisites works. It falls under the class of discovery testing.

4.8.4 System Testing

Framework testing is the trying to guarantee that by placing the product in various conditions (e.g., Operating Systems) it actually works. Framework testing is finished with full framework execution and climate. It falls under the class of discovery testing.

4.8.5 Stress Testing

Stress testing is the trying to assess how framework carries on under negative conditions. Testing is directed at past constraints of the particulars. It falls under the class of discovery testing.

4.8.6 Performance Testing

Execution testing is the trying to survey the speed and viability of the framework and to ensure it is producing results inside a predetermined time as in execution prerequisites. It falls under the class of discovery testing.

4.8.7 Usability Testing

Convenience testing is performed to the viewpoint of the customer, to assess how the GUI is easy to understand? How effectively can the customer learn? Subsequent to figuring out how to utilize, how capably can the customer perform? How satisfying is it to utilize its plan? This falls under the class of discovery testing.

4.8.8 Acceptance Testing

This testing is regularly done by the client to guarantee that the conveyed item meets the necessities and functions as the client anticipated. It falls under the class of discovery testing.

4.8.9 Regression Testing

This testing is the trying after adjustment of a framework, part, or a gathering of related units to guarantee that the alteration is working effectively and isn't harming or forcing

5.RESULTS AND DISCUSSIONS

5.1.1 HOME PAGE

A credit card home page is available in the page. It has created the existing user can be log in .

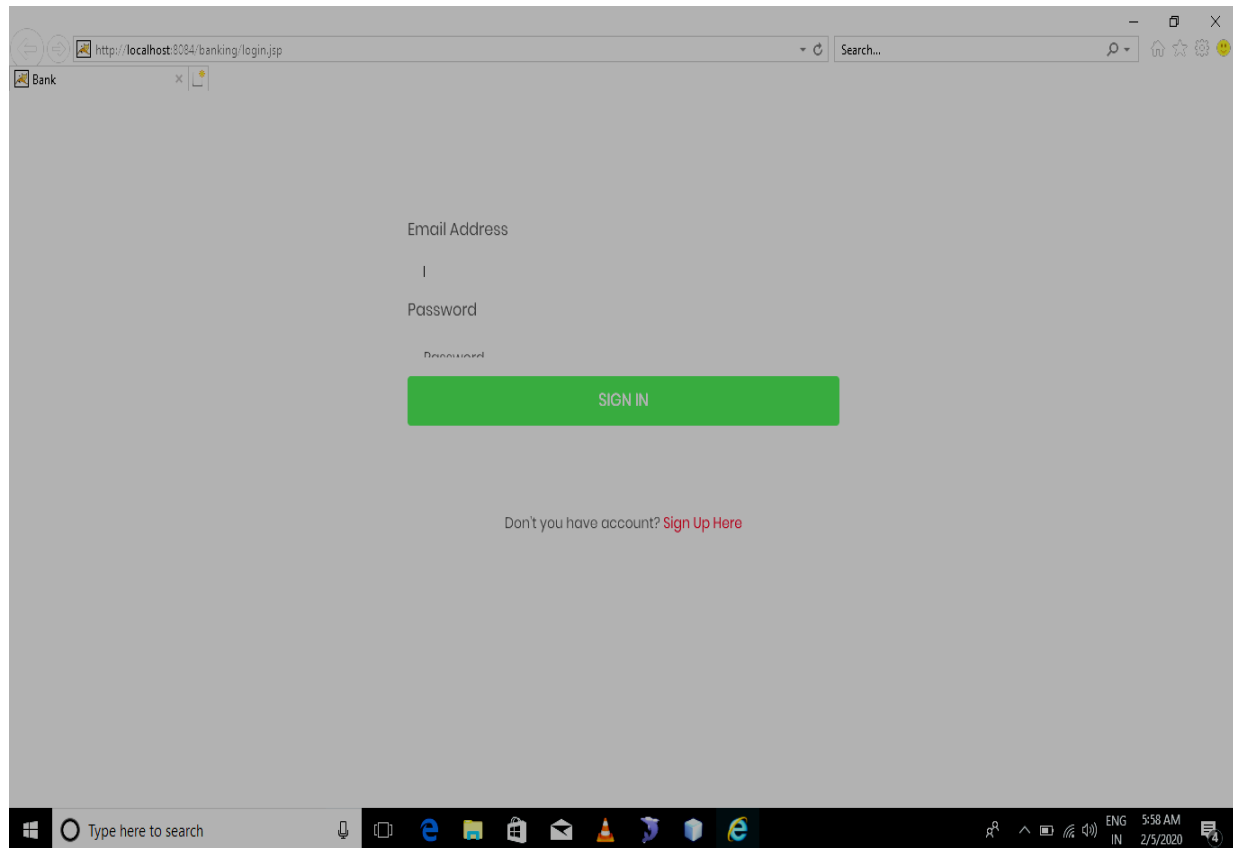


Fig.A.5.1.1 HOME PAGE

5.1.2 ADMIN SIGNUP

In the page has been singin for new user signup the page. There are new user used in there websites.

There number of steps in the web page and name and address are entered to the webpage there new user used.

Bank

http://localhost:8084/banking/register.jsp

Search...

Username
sakthi

Email Address
msakthi@gmail.com

Password
.....

Name
sakthi

Address
chennai

Gender
Male

NEXT PAGE

Type here to search

6:04 AM 2/5/2020

Fig.B 5.1.2 ADMIN SIGNUP

5.1.3 ADMIN BALANCE

In the account user are balance checking to the webpage and used one or more to account.

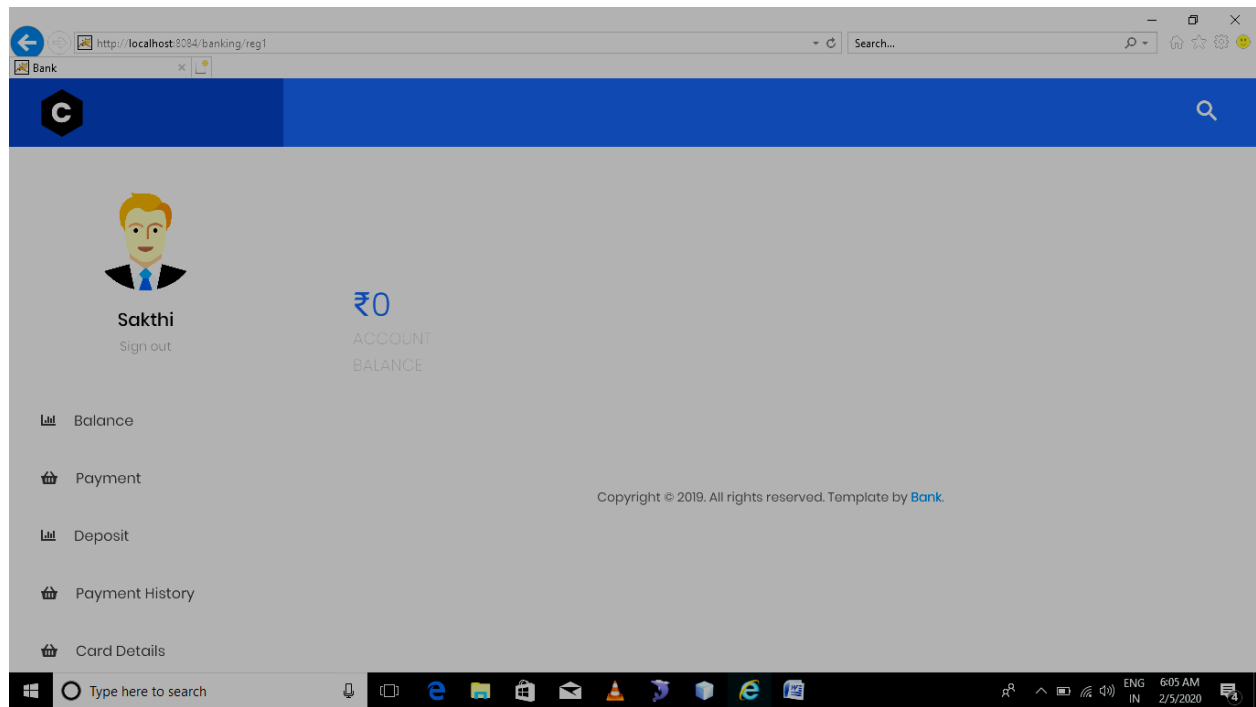


Fig.C 5.1.3 ADMIN BALANCE PAGE

5.1.4 DEPOSIT PAGE

User Deposit money in our external account. Because in you could shopping there money contain in our account. If there zero account balance in our account cannot purchase any shopping, etc.....

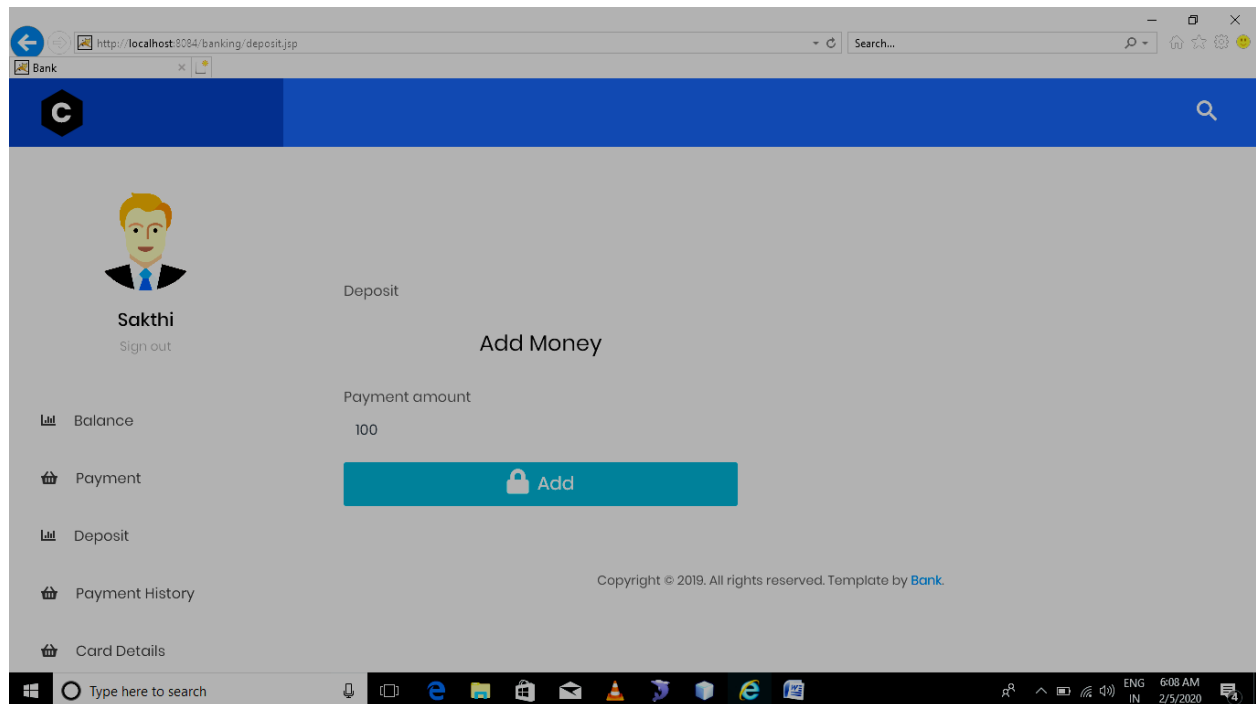


Fig.D 5.1.4 DEPOSIT PAGE

5.1.5 AVAILABLE BALANCE

In there user will be deposit some , amount for using shopping and Balance will be increased

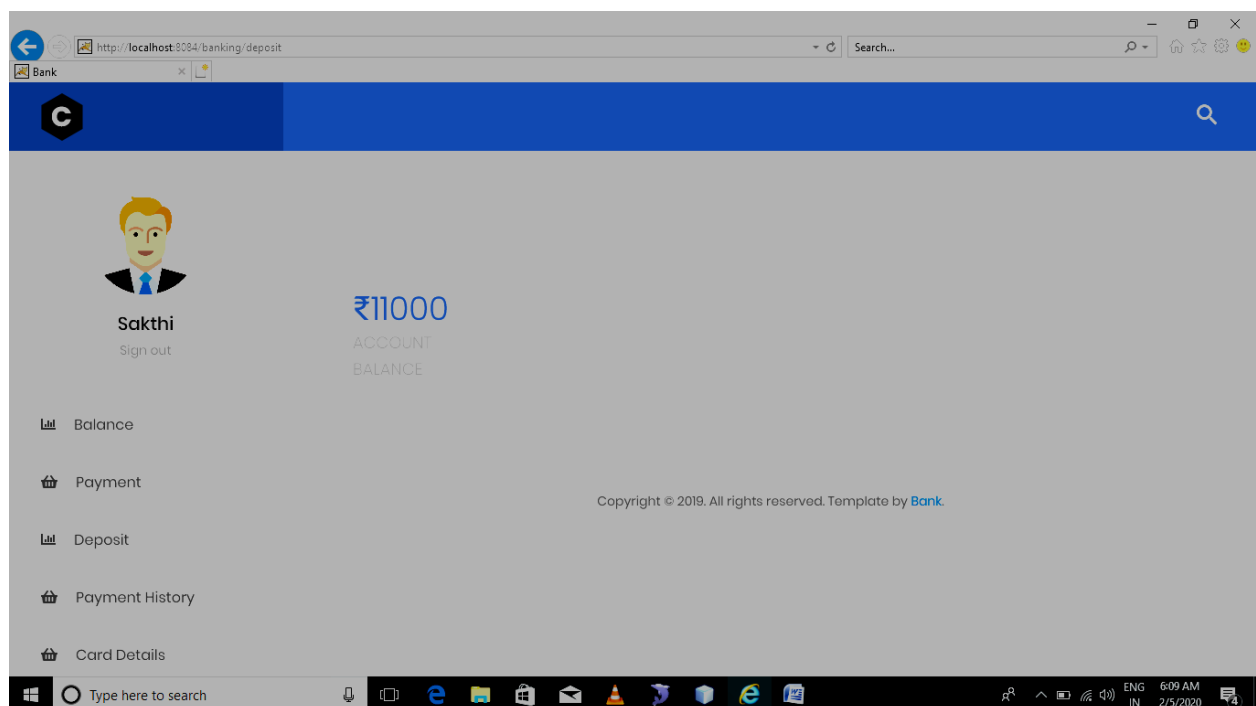


Fig.E 5.1.5 AVAILABLE BALANCE

5.1.6 PAYMENT TRANSACTION

User will be added our information. Some example of payment for shopping Purpose. And amount of money to be added for purchase. And input to the credit an Card number to be purchase. In the payment number name ,and expiry date and Security code.

Bank

Pay Invoice

 **Sakthi**
Sign out

Balance

Payment

Deposit

Payment History

Card Details

Payment For
shopping

Payment amount
100.00

Name on card
sakthi

Card number
435464354353635

Expiration
15/25

Security code
3232

Pay

Fig.F 5.1.6 PAYMENT TRANSACTION

5.1.7 ADMIN QUESTION PAGE

Then ask one question has been confirmation of user. And the determined the user will be many more question will be generated , the questions and answer will be stored automatically in database server, for automatically stored. User will be answer to correct questions there are valid user will be generated and open.

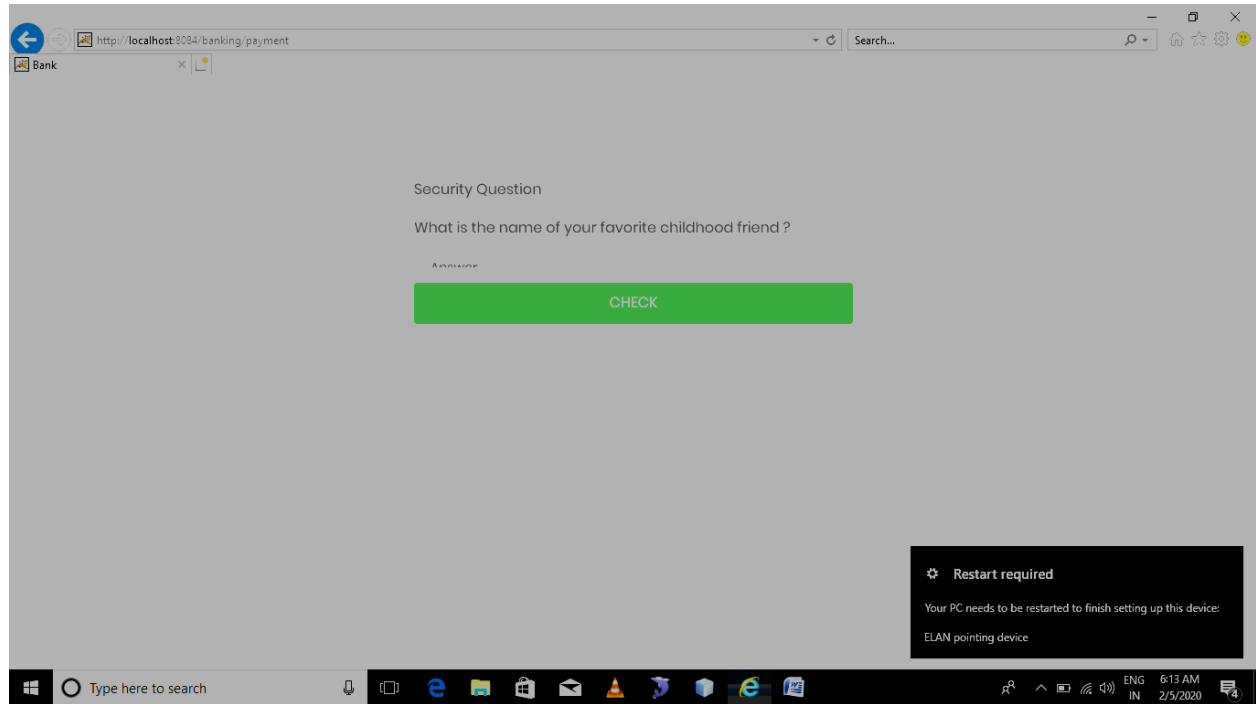


Fig.G 5.1.7 ADMIN QUESTION PAGE

5.1.8 CHECKING VALID USER

Next to if the user will ask correct questions there will be open to valid user.

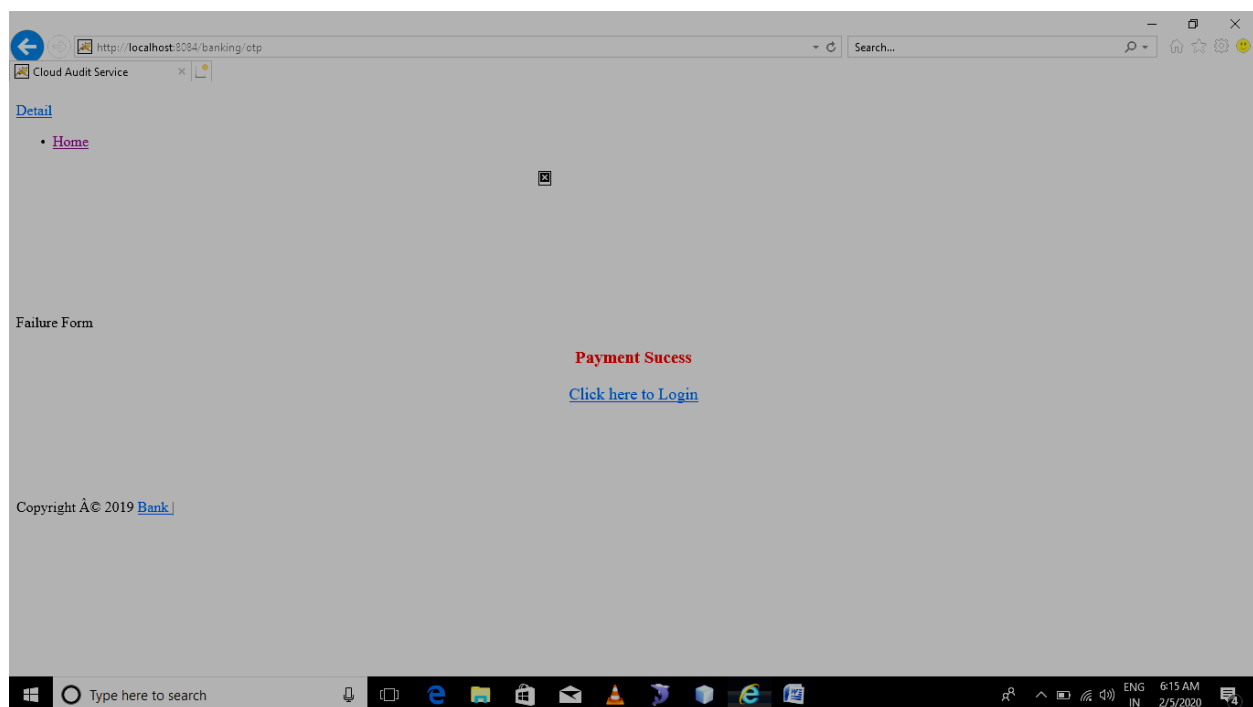


Fig.H 5.1.8 CHECKING VALID USER

5.1.9 PAYMENT TRANSACTION HISTORY

There will be all of the payment details in the page. In the viewed for user purpose.

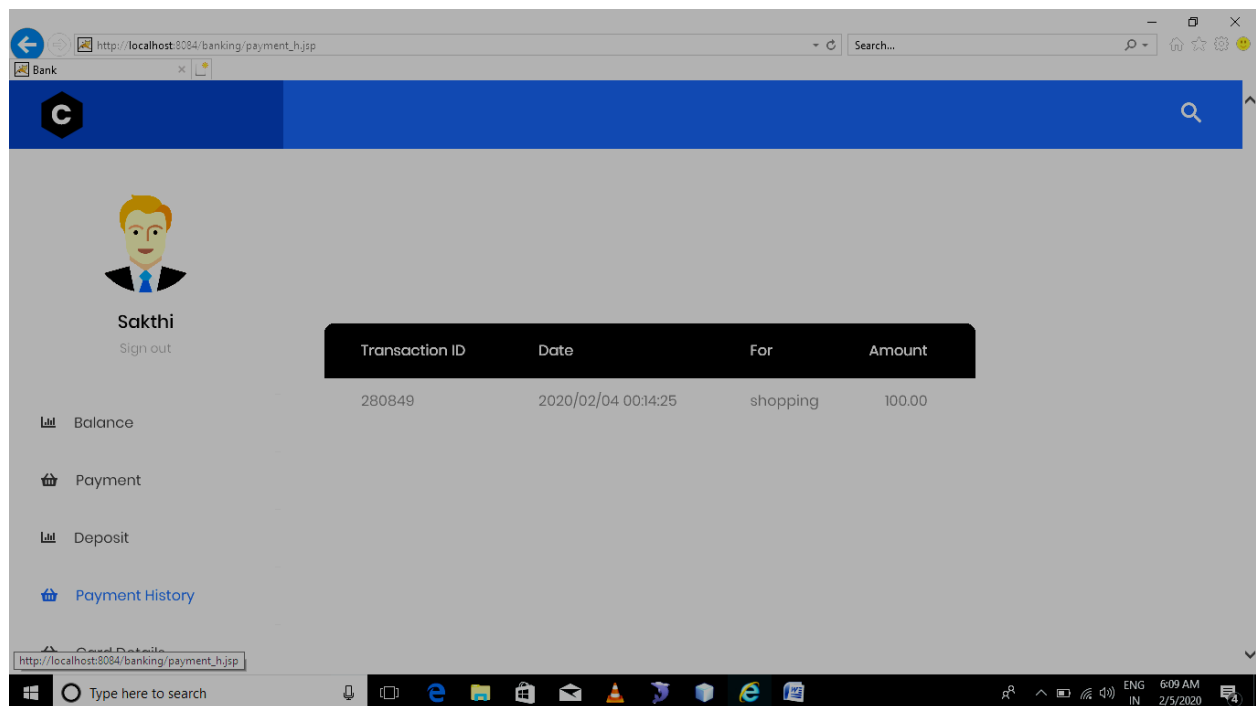


Fig.I 5.1.9 PAYMENT TRANSACTION HISTORY

5.1.10 DATA STORED IN DATABASE SERVER

This all transaction is stored in the database server and there is all user details can be stored in the database

SQLyog Enterprise - MySQL GUI - [New Connection - root@localhost]

File Edit Favorites DB Table Objects Tools Powertools Window Help

root@localhost

- information_schema
- bank
 - Tables
 - account
 - card
 - question
 - transaction
 - Views
 - Stored Procs
 - Functions
 - Triggers
- mysql
- test

Query

Autocomplete: [Tab]->Next Tag, [Ctrl+Space]->List Matching Tags, [Ctrl+Enter]->List All Tags.

1

1 Result 2 Messages 3 Table Data 4 Objects 5 History

Show All Or Limit 0 50 Refresh

ac	uname	mail	pwd	name1	address1	gender	pan	aadhar
(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)
637994586149	benin	beniny.gtt@gmail.com	benin	benin	chennai	Male	BKN12464	123456789907456
978458129064	jessy	jasintha.gtt@gmail.com	12345	jasintha	chennai	Female	ABCDE1234H	123456785623
681646379389	tamil	tamil.gtt@gmail.com	tamil	tamilselvi	chennai	Female	1234567890	789456123012
504301708955	kishan	kishankrky@gmail.com	12345	Kishan Kumar	Chennai	Male	ABCDE1234K	123478956325
483906124452	tamilraji	gtsscse@gmail.com	tamilraji	tamil	chennai	Female	ABCDE456ID	123456789012
342164205836	kumar	kumar@gmail.com	kumar	kumar	chennai	Male	BKN12464	123456789907456
221616905517	sakthi	sakthi@gmail.com	sakthi	sakthi	chennai	Male	BKN12464	123456789907456
716449919646	sakthi	msakthi@gmail.com	sakthi	sakthi	chennai	Male	3453535	35353535

Rebuilding tag file completed

0 ms 9 row(s) Connections: 1 Registered To: mazuki

Type here to search

ENG 2:50 PM 2/10/2020

Fig.J 5.1.10 DATA STORED IN DATABASE SERVER

CHAPTER - 6

CONCLUSION& FUTURE ENHANCEMENT

6.1 CONCLUSION

In this paper, we have talked about that How Hidden Markov Model will be valuable to recognize false online exchange through Mastercard. The proposed Fraud Detection System is additionally adaptable for taking care of immense volumes of exchanges information preparing. The HMM based charge card misrepresentation location framework isn't having complex cycle to perform extortion check like the current framework. Proposed Fraud identification framework gives certified and quick outcome than existing framework. The Hidden Markov Model makes the handling of recognition simple and attempts to eliminate the intricacy.

6.2 FUTURE ENHANCEMENT

We have examined a utilization of HMM in charge card extortion recognition. The various strides in Mastercard exchange preparing are addressed as the hidden stochastic cycle of a HMM. We have study the scopes of exchange sum as the perception images, though the sorts of thing have been viewed as conditions of the HMM. We have study the proposed a technique for discovering the spending profile of cardholders, just as use of this information in choosing the estimation of perception images and beginning assessment of the model boundaries. It has likewise been clarified how the HMM can identify if an approaching exchange is fake. The framework is likewise versatile for taking care of enormous volumes of exchanges

REFERENCES

[1] M. A. Al-Shabi, "Credit card extortion location utilizing auto encoder model in uneven datasets," J. Adv. Math. Comput. Sci., vol. 33, no. 5, pp. 1–16, 2019.

[2] U. Fiore, A. D. Santis, F. Perla, P. Zanetti, and F. Palmieri, "Using generative ill-disposed organizations for improving grouping viability in charge card misrepresentation identification," Inf. Sci., vol. 479, pp. 448–455, Apr. 2019.

[3] S. V. S. S. Lakshmi and S. D. Kavilla, "Machine learning for Mastercard misrepresentation identification framework," Int. J. Appl. Eng. Res., vol. 13, no. 24, pp. 16819–16824, 2018.

[4] I. Mekterović, L. Brkić, and M. Baranovi, "A precise survey of information mining ways to deal with Visa misrepresentation identification," WSEAS Trans. Transport. Econ., vol. 15, p. 437, Jan. 2018.

[5] N. Carneiro, G. Figueira, and M. Costa, "An information digging based framework for charge card

misrepresentation identification in e-tail," Decis. Backing Syst., vol. 95, pp. 91–101, Mar. 2017.

[6] "Credit Card Fraud Detection-by Ishu Trivedi, Monika, Mrigya, Mridush" published by international Journal of Advanced Research in Computer and Communication Engineering Vo. 5, Issue 1, January 2016.

[7] J. West and M. Bhattacharya, "Smart monetary extortion location: a far reaching survey," Computers and Security, vol. 57, pp. 47-66, 2016

[8] F. Ghobadi and M. Rohani, "cost sensitive modeling of credit card fraud using neural network strategy", 2016 2nd International Conference of Signal Processing and intelligent systems (ICSPIS), Tehran, 2016, pp. 1-5.

[9] "Survey Paper on Credit Card Fraud Detection by Suman", Research Scholar, GJUS&T Hisar HCE, Sonapat published by International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3 Issue 3, March 2014.

[10] E. W. T. Nagai. Y. Hu, Y. H. Wong, Y. Chen, and X. Sun, "The application of data mining techniques in financial fraud detection a classification framework and academic review of literature", Decision support Systems, vol. 50. no. 3, pp. 559-569, 2011.

[12] Chang C. & Chang S. (2010). The design of E-Travels check with efficiency and Mutual Authentication, Journal of Networks, Vol. 5. No. 3, pp. 275-282.

[13] Pearl Pu and Li Che, "Trust Building with Explanation Interfaces". ICWE 2010, Vienna, Austria, July 5-9, 2010. Proceedings, 2010, pp. 11.

[14] H. Abdou, J. Pointon, and A. El-Masry, Neural nets versus conventional techniques in credit scoring in Egyptian banking.. Expert System with Applications, 35, 1275-1292, 2008.

[15] T.H.T. Dinh, and S. Kleimeier, A credit scoring model for Vietnam's retail banking market. Global Review of Financial Analysis, 16, 471-495, 2007.

APPENDIX

A.SOURCE CODE

`/*`

`* To change this license header, choose License Headers in Project Properties.`

`* To change this template file, choose Tools | Templates`

`* and open the template in the editor.`

`*/`

`import java.io.IOException;`

`import java.io.PrintWriter;`

`import javax.servlet.ServletException;`

`import javax.servlet.annotation.WebServlet;`

`import javax.servlet.http.HttpServlet;`

`import javax.servlet.http.HttpServletRequest;`

```

import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author Satheeshkumar S
 */

@WebServlet(urlPatterns = {"/card"})

public class card extends HttpServlet { /**
    * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
    * methods.
    * @param request servlet request
    * @param response servlet response
    * @throws ServletException if a servlet-specific error occurs
    * @throws IOException if an I/O error occurs
    */

    protected void processRequest(HttpServletRequest request, HttpServletResponse
response)

        throws ServletException, IOException {

        response.setContentType("text/html;charset=UTF-8");

        try (PrintWriter out = response.getWriter()) {

```

```
/* TODO output your page here. You may use following sample code. */

out.println("<!DOCTYPE html>");

out.println("<html>");

out.println("<head>");

out.println("<title>Servlet card</title>");

out.println("</head>");

out.println("<body>");

out.println("<h1>Servlet card at " + request.getContextPath() + "</h1>");


out.println("</body>");

out.println("</html>");

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign
on the left to edit the code.">

/**

 * Handles the HTTP <code>GET</code> method.

 *

 * @param request servlet request
```

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

* Handles the HTTP <code>POST</code> method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

```
protected void doPost(HttpServletRequest request, HttpServletResponse response)
```

```
    throws ServletException, IOException {
```

```
        processRequest(request, response);
```

```
    }
```

```
/**
```

```
 * Returns a short description of the servlet.
```

```
 *
```

```
 * @return a String containing servlet description
```

```
 */
```

```
@Override
```

```
public String getServletInfo() {
```

```
    return "Short description";
```

```
}// </editor-fold>
```

```
}
```

```
/*
```

```
 * To change this license header, choose License Headers in Project Properties.
```

```
 * To change this template file, choose Tools | Templates
```

* and open the template in the editor.

*/

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

/**

*

* @author Satheeshkumar S

```
*/
```

```
@WebServlet(urlPatterns = {"/deposit"})
```

```
public class deposit extends HttpServlet {
```

```
String username="";
```

```
String password="";
```

```
String type1="";
```

```
Connection con=null;
```

```
Statement st=null,st1=null;
```

```
ResultSet rs=null;
```

```
/**
```

```
 * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
```

```
 * methods.
```

```
 *
```

```
 * @param request servlet request
```

```
 * @param response servlet response
```

```
 * @throws ServletException if a servlet-specific error occurs
```

```
 * @throws IOException if an I/O error occurs
```

```
*/
```

```
protected void processRequest(HttpServletRequest request, HttpServletResponse  
response)
```

```
    throws ServletException, IOException {
```

```
    response.setContentType("text/html;charset=UTF-8");
```

```
    try (PrintWriter out = response.getWriter()) {
```

```
        /* TODO output your page here. You may use following sample code. */
```

```
        out.println("<!DOCTYPE html>");
```

```
        out.println("<html>");
```

```
        out.println("<head>");
```

```
        out.println("<title>Servlet deposit</title>");
```

```
        out.println("</head>");
```

```
        out.println("<body>");
```

```
        out.println("<h1>Servlet deposit at " + request.getContextPath() + "</h1>");
```

```
        out.println("</body>");
```

```
        out.println("</html>");
```

```
    }
```

```
}
```

```
// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign  
on the left to edit the code.">
```

```
/**
```

* Handles the HTTP `GET` method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

 processRequest(request, response);

}

/**

* Handles the HTTP `POST` method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse res)

throws ServletException, IOException {

username = req.getParameter("cc-payment");

// password = req.getParameter("pwd");

//type1 = req.getParameter("type1");

HttpSession sn = req.getSession(true);

//sn.setAttribute("username",username);

String username1= sn.getAttribute("username").toString();

System.out.println("username : " +username);

System.out.println("otp : " +password);

System.out.println("type1 : " +type1);

RequestDispatcher rd = null;

try {

Class.forName("com.mysql.jdbc.Driver");

```

con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/bank","root","root");

st1 = con.createStatement();

rs = st1.executeQuery("select amount from account where
uname='"+username1+"'");

if(rs.next()) {

String aa=rs.getString(1);

String bal=String.valueOf(Integer.parseInt(aa)+Integer.parseInt(username));

int add=st1.executeUpdate("update account set amount='"+bal+"' where
uname='"+username1+"'");


sn.setAttribute("balance",bal);

rd=req.getRequestDispatcher("balance.jsp");


//sn.setAttribute("dpm",department);

} else {

rd=req.getRequestDispatcher("failure.jsp");

}

rd.forward(req,res);

```

```
    }

    catch(Exception e2)

    {

        System.out.println("Exception : "+e2.toString());

    }

}
```

```
/**

 * Returns a short description of the servlet.

 *

 * @return a String containing servlet description

 */

@Override

public String getServletInfo() {

    return "Short description";

} // </editor-fold>

}
```

```
import java.util.Date;

import javax.activation.DataHandler;

import javax.activation.DataSource;

import javax.activation.FileDataSource;

import javax.mail.BodyPart;

import javax.mail.Message;

import javax.mail.MessagingException;

import javax.mail.Multipart;

import javax.mail.Session;

import javax.mail.Transport;

import javax.mail.internet.InternetAddress;

import javax.mail.internet.MimeBodyPart;

import javax.mail.internet.MimeMessage;

import javax.mail.internet.MimeMultipart;
```

```
public class EmailUtil {
```

```
    /**
```

```
     * Utility method to send simple HTML email
```

* @param session

* @param toEmail

* @param subject

* @param body

*/

```
public static void sendEmail(Session session, String toEmail, String subject, String  
body){
```

```
    try
```

```
    {
```

```
    }
```

```
        return macAddressBuilder.toString();
```

```
    }
```

```
public static void main(String[] args) throws UnknownHostException, SocketException
```

```
{
```

```
    Sy
```

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

/* TODO output your page here. You may use following sample code. */

out.println("<!DOCTYPE html>");

out.println("<html>");

out.println("<head>");

out.println("<title>Servlet deposit</title>");

out.println("</head>");

out.println("<body>");

out.println("<h1>Servlet deposit at " + request.getContextPath() + "</h1>");

out.println("</body>");

```

        out.println("</html>");

    }

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign
on the left to edit the code.">

/**

 * Handles the HTTP <code>GET</code> method.

 *

 * @param request servlet request

 * @param response servlet response

 * @throws ServletException if a servlet-specific error occurs

 * @throws IOException if an I/O error occurs

 */

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

```

```
    processRequest(request, response);  
}
```

```
/**
```

```
 * Handles the HTTP <code>POST</code> method.
```

```
 *
```

```
 * @param request servlet request
```

```
 * @param response servlet response
```

```
 * @throws ServletException if a servlet-specific error occurs
```

```
 * @throws IOException if an I/O error occurs
```

```
 */
```

```
@Override
```

```
protected void doPost(HttpServletRequest req, HttpServletResponse res)
```

```
    throws ServletException, IOException {
```

```
    username = req.getParameter("cc-payment");
```

```
    // password = req.getParameter("pwd");
```

```
    //type1 = req.getParameter("type1");
```

```
    HttpSession sn = req.getSession(true);
```

```

//sn.setAttribute("username",username);

String username1= sn.getAttribute("username").toString();

System.out.println("username : " +username);

System.out.println("otp : " +password);

System.out.println("type1 : " +type1);

        RequestDispatcher rd = null;

    try {

        Class.forName("com.mysql.jdbc.Driver");

        con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/bank","root","root");

        st1 = con.createStatement();

        rs = st1.executeQuery("select amount from account where
uname='"+username1+"'");

        if(rs.next()) {

            String aa=rs.getString(1);

            String bal=String.valueOf(Integer.parseInt(aa)+Integer.parseInt(username));

            int add=st1.executeUpdate("update account set amount='"+bal+"' where
uname='"+username1+"'");

            sn.setAttribute("balance",bal);

```

```
rd=req.getRequestDispatcher("balance.jsp");

//sn.setAttribute("dpm",department);

    } else {

        rd=req.getRequestDispatcher("failure.jsp");

        }

        rd.forward(req,res);

    }

catch(Exception e2)

{

    System.out.println("Exception : "+e2.toString());

}

}

/**
```

* Returns a short description of the servlet.

*

* @return a String containing servlet description

*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

import java.util.Date;

}

/**

* Returns a short description of the servlet.

*

* @return a String containing servlet description

*/

@Override

public String getServletInfo() {

```
return "Short description";
```

```
}// </editor-fold>
```

```
}
```

```
int add=st1.executeUpdate("update account set amount='"+bal+"' where  
uname='"+username1+"'");
```

```
sn.setAttribute("balance",bal);
```

```
rd=req.getRequestDispatcher("balance.jsp");
```

```
//sn.setAttribute("dpm",department);
```

```
} else {
```

```
rd=req.getRequestDispatcher("failure.jsp");
```

```
}
```

```
rd.forward(req,res);
```

```
}
```

```
catch(Exception e2)
```

```
}}
```

B.PUBLICATION

