

A PRELIMINARY REPORT ON
“TRANSACTION FRAUD DETECTION BASED ON TOTAL ORDER RELATION AND
BEHAVIOUR DIVERSITY”

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

Salunke Neha	71701694D
Salehittal Atharva	71701692H
Naiknavare Shivani	71701611M



DEPARTMENT OF COMPUTER ENGINEERING
PVG's COLLEGE OF ENGINEERING & TECHNOLOGY

44, VIDYANAGARI, PARVATI, PUNE 411009

SAVITRIBAI PHULE PUNE UNIVERSITY
2019-20



CERTIFICATE

This is to certify that the project report entitles

**“TRANSACTION FRAUD DETECTION BASED ON TOTAL ORDER RELATION AND
BEHAVIOUR DIVERSITY”**

Submitted by

STUDENT NAME	Exam No:
Salunke Neha	71701694D
Salehittal Atharva	71701692H
Naiknavare Shivani	71701611M

is a bonafide student of this institute and the work has been carried out by him/her under the supervision of **Prof. A.M.Bhadgale** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering** (Computer Engineering).

(Prof. A.M.Bhadgale)
Guide
Department of Computer Engineering

(Prof M. V. Marathe)
Head,
Department of Computer Engineering

(Dr. Mrs. K J Kulkarni)
I/C Principal,
PVG's College of Engineering & Technology, Pune – 41

Place: Pune

Date:

ACKNOWLEDGEMENT

No volume of words is enough to express our gratitude towards our guide, Prof A.M.BHADGALE, who has been very concerned and have aided for all the help essentials for the preparation of this work. He has helped us explore this vast topic in an organized manner and provided us with all the ideas on how to work towards a research oriented venture.

We wish to express our sincere gratitude towards Project Coordinators Prof U. M. KALSHETTI and reviewers Prof D. D. SAPKAL and Prof A. G. SAWANT for their timely guidance and suggestion. We are also thankful to Prof M. V. MARATHE, Head of Computer Engineering Department for the motivation and inspiration that triggered for the project work.

SALUNKE NEHA

SALEHITTAL ATHARVA

NAIKNAVARE SHIVANI

ABSTRACT

Due to the rise and rapid growth of E-Commerce, use of credit cards for online purchases has dramatically increased and it caused an explosion in the credit card fraud. As credit card becoming the most suitable use for both online and offline payment, frauds associated with it have also increased. In real life, fraudulent transactions are scattered with genuine transactions and simple pattern matching algorithm are not sufficient to detect those fraud correctly.

Year after year the damages inflicted by credit card problem are growing rapidly. In 2014 alone it is estimated the total global monetary loss was 16.31 million dollars. To build a credit card transaction system which would prevent fraud and even if any person does a fraud, he/she would be caught easily and money losses due to fraud can be saved. To solve the problem we have suggested to build an app that would have two way verification i.e. first face recognition and then pin to do payment and if face is not recognized then face of the person is captured and sent to police along with it live location from where the payment is being done is also displayed along with shortest route to follow it. Emergency blocking of account is provided.

This project purpose is to find out the fraud transactions done by fraud users with high accuracy security system. By using user behavior methodology it's easy to detect the pattern transaction to detect the transaction purpose. This system tracks the fraud used using location tracking with Google Map API. Account blocking facility is added in application to avoid fraud transactions. In particular, since there is no limit on the number of features that can be calculated, a system may take too long to make a decision based on the time spent recalculating the features with each new transaction.

Contents

1	INTRODUCTION	8
1.1	Abstract	8
1.2	Objectives	8
1.3	Motivation	8
1.4	Problem Definition	9
2	LITERATURE SURVEY	10
3	SOFTWARE REQUIREMENT SPECIFICATION	15
3.1	Introduction	15
3.1.1	Project Scope	15
3.1.2	User Classes and Characteristics	15
3.1.3	Assumptions and Dependencies	16
3.2	Functional Requirement	17
3.2.1	Face Recognition	17
3.2.2	Live Location Tracking	17
3.2.3	Emergency Account Blocking	17
3.3	External Interface Requirement	18
3.3.1	User Interfaces	18
3.3.2	Hardware Interfaces	18
3.3.3	Software Interfaces	18

3.3.4	Communications Interfaces	18
3.4	Non-Functional Requirement	20
3.4.1	Usability	20
3.4.2	Reliability	20
3.4.3	Performance	20
3.4.4	Supportability	20
3.4.5	Safety	21
3.4.6	Security	21
3.5	System Requirement	22
3.5.1	Database Requirement	22
3.5.2	Software Requirement	22
3.5.3	Hardware Requirement	22
3.6	Analysis Model: SDLC Model to be applied	23
3.7	System Implementation Plan	25
4	SYSTEM DIAGRAM	26
4.1	System Architecture	26
4.2	Data Flow Diagram	27
4.3	UML Diagrams	28
4.3.1	Use Case Diagram	28
4.3.2	Class Diagram	29
4.3.3	Sequence Diagram	30

4.3.4	Deployment Diagram	31
4.3.5	Activity Diagram	32
5	Project Plan	33
5.1	Project Estimate	33
5.1.1	Reconciled Estimate	33
5.1.2	Project Resources	33
5.2	Risk Management	33
5.2.1	Risk Identification	33
5.2.2	Risk Analysis	34
5.3	Project Schedule	35
5.3.1	Project Task Set	35
5.3.2	Task Network	35
5.3.3	Timeline Chart	36
5.4	Team Organization	37
5.4.1	Team structure	37
5.4.2	Management reporting and communication	38
6	Project Implementation	39
6.1	Overview of Project Module	39
6.2	Tools and Technologies Used	39
6.2.1	Technology Description	39
6.2.2	Hardware Specifications	40

6.2.3	Software Specifications	40
6.3	Algorithm Details	40
6.3.1	CNN Algorithm	40
7	Software Testing	42
7.1	Types of Testing	42
7.1.1	Unit Testing	42
7.1.2	Integration Testing	42
7.1.3	Functional Testing	42
7.1.4	White Box Testing	43
7.1.5	Black Box Testing	43
7.1.6	Unit Testing	43
7.1.7	Integration Testing	44
7.1.8	Acceptance Testing	44
7.1.9	Alpha and Beta Testing	44
7.1.10	System Testing	44
7.2	Test Case and Test Results	45
8	Results	50
8.1	Outcomes	50
8.2	Screenshots	51
9	Conclusion and Future Work	57

9.1	Conclusion	57
9.2	Future Work	57
9.3	Applications	57

List of Figures

3.1	SDLC Analysis Model	24
3.2	System Implementation Plan	25
4.1	System Architecture	26
4.2	Data Flow Diagram	27
4.3	Use Case Diagram	28
4.4	Class Diagram	29
4.5	Sequence Diagram	30
4.6	Deployment Diagram	31
4.7	Activity Diagram	32
5.1	Task Network Diagram	35
5.2	Timeline Chart	36
5.3	Team structure	37
5.4	Management reporting and communication	38
6.1	CNN Algorithm	41
8.1	IP Address	51
8.2	Registration	52
8.3	Welcome Page	53
8.4	Google Maps	54
8.5	User Tracking and Blocking	55
8.6	Face Recognition	56

List of Tables

1	Risk Analysis	34
2	Test Cases	47
3	Test Cases	48
4	Test Cases	49

1 INTRODUCTION

1.1 Abstract

Due to the rise and rapid growth of E-Commerce, use of credit cards for on-line purchases has dramatically increased and it caused an explosion in the credit card fraud. As credit card becomes the most popular mode of payment for both online as well as regular purchase, cases of fraud associated with it are also rising. In real life, fraudulent transactions are scattered with genuine transactions and simple pattern matching techniques are not often sufficient to detect those frauds accurately. Implementation of efficient fraud detection systems has thus become imperative for all credit card issuing banks to minimize their losses. Many modern techniques based on Artificial Intelligence, Data mining, Fuzzy logic, Machine learning, Sequence Alignment, Genetic Programming etc., has evolved in detecting various credit card fraudulent transactions. A clear understanding on all these approaches will certainly lead to an efficient credit card fraud detection system. All the above techniques used need some historical data for their detection.

To improve the problem of the historical data we have decided to check each transaction thus saving the cost if transaction first or last if it is anomalous it is removed and thus saving fraud.

1.2 Objectives

To build a credit card transaction system which would prevent fraud and even if any person does a fraud, he/she would be caught easily and the money losses due to fraud can be saved.

1.3 Motivation

The objectives of credit card fraud detection are to reduce losses due to payment fraud for both merchants and issuing banks and increase revenue opportunities for merchants. In recent years, topics such as fraud detection and fraud prevention

have received a lot of attention on the research front, in particular from payment card issuers. The reason for this increase in research activity can be attributed to the huge annual financial losses incurred by card issuers due to fraudulent use of their card products. A successful strategy for dealing with fraud can quite literally mean millions of dollars in savings per year on operational costs.

1.4 Problem Definition

Year after year, the damages inflicted by the credit card fraud problem are growing rapidly. In the year 2014 alone, it was estimated that the total global monetary loss was in 16.31 billion dollars – accumulation of damages from card issuers, acquiring banks and merchants. Financial fraud is an ever growing menace with far reaching consequences in the finance industry, corporate organizations, and government.

2 LITERATURE SURVEY

Classical and Deep Learning Classifiers for Anomaly Detection, Proceedings of 2019 16th International Bhurban Conference on Applied Sciences Technology

Manahil Raza and Usman Qayyum have presented the way machine learning can be used in cyber crime detection. The approach searches for hidden patterns and then discovers it and identifies the fraud. They have used various techniques of ML like SVM, Neural Network and Decision Tree and compared their performances. Overall the paper is good to know the performances and which is the best algorithm to be applied but without the history dataset it is not possible which may be a problem if the first transaction itself may be a fraud.

Anomaly Detection: A Survey ACM Computing Surveys, 09

In the paper of Chandola et.al [13], they presented an overview of the algorithms that can handle anomaly detection; utilization of these algorithms is still on a case to case basis. Not only have they reviewed previous related works from other literatures, they have also given some insights on the possible hindrances that the researcher might experience in handling such topic. Moreover, the proponents cited other possible applications of these detection algorithms with their advantages and disadvantages. Overall, the paper is a good general background reference in understanding the available anomaly detection techniques.

Application of Data Mining in Preventing Credit Card Fraud in International Conference on Management and Service Science

A similar research domain was presented by Haiying Ma and Xin Li [14] where they used an advanced data-mining algorithm which will dynamically produce the ideal attribute combination – the Genetic Algorithm. This algorithm has the ability to perform natural selection of the dataset's attributes heuristically with respect to the chosen fitness-function. By applying this algorithm, it was able to generate a 78% identification rate.

Research on Credit Card Fraud Detection Model Based on Distance Sum in International Joint Conference on Artificial Intelligence

Another similar domain literature is from Wen-Fang Yu and Na Wang [9] where they examined the outlier data-mining approach to identify fraudulent data-points from the dataset. In this concept, fraud is manifested as an isolated point in the vector space and it could appear independently or somewhat included in a small group of clustered data-points. Their technique produces 89.4% accuracy when the outlier threshold is set to 12.

Solving Credit Fraud Detection Problem by the New Metaheuristic Migratory Birds Optimization Advances in Computational Intelligence: 12th International Work-Conference on Artificial Neural Networks

Lastly, Duman et al. [15] had successfully implemented a credit card fraud detection system in a Turkish bank using the Migrating Birds Optimization (MBO) algorithm. The paper used the saved limit ratio (SLR) as their primary performance criterion and was fortified by the True Positive Rate (TPR) value. Their MBO algorithm obtained 93.9% for SLR and 91.45% for TPR; the selection of this algorithm was further verified via t-tests. The paper also provided some insights on the current fraud detection techniques, option in handling instance distribution of fraudulent and normal transactions, and observations in using common algorithms in this field. The researcher reviewed the literature pertaining to credit card services from a host of sources.

The important of such literature mainly consists of review of Ph. D. theses, review of committee reports and review of research articles, which are presented below.

Ph. D Theses

Sarma E. J., (1989)¹⁰, used the ‘Lifestyle’ concept to understand the users of credit card psycho graphically in 1989. He concluded that “the users of credit cards in contrast to non-users exhibit a contemporary state of mind and reject conservative traditional concepts, can indicate that the type of purchases that can be made on credit cards could be viewed as ‘luxury or necessity’ based on the lifestyles rather than broadly classifying them for all groups. Therefore the traditional orientation towards cash which can be a major barrier to the use of credit cards must be tackled by card marketing banks through better marketing”. He further concluded that “the major problem with the banks today is not only of popularizing the credit card concept but also of increasing the credit card usage.” Even after two decades much has not changed.

Yeo (1990)¹¹, in his Ph. D. thesis looks at identifying household credit card choice and usage behavior of cardholders. It was found that several factors were significant in determining the choice between the bank and retail credit cards in USA. These were the level of family income, the household age, the number of bank card holdings, the number of store card holding the related interest rate, the relative membership fees and the card preference.

Yeo Jung Sang (1992)¹², in his thesis had focused on the attitude and behavior of credit card holders. The researcher had used dichotomous multi nominal logic to analyze the demographic factors like age, education level, size of the house hold, income level, number of bank card holdings, store card holdings, relative interest rate, and relative membership fee. The results of the study revealed that the effects.

Punjavat (1993)¹³, in his Ph. D. thesis examines Graduate International Students' knowledge, attitude, experience, practices and satisfaction relating to credit card. This study was conducted on 261 students. Finding showed that the respondents' credit card knowledge was low, attitudes were favorable and Pre U.S experience is limited. Since living in the U.S respondents had become regular user with a mean of 3.3 cards and a majority charging more than \$200 per month. They followed commonly recommended practices and eight of ten were satisfied with their credit card use, several statistically significant relationship were found among the credit card variables and between this variables and socio-demographic characteristics such as country of origin and length of U.S. pre-developed and evaluated.

Parimala (2001)¹⁴, in her Ph. D. thesis highlighted the marketing environment of credit cards in Trichirappalli. Her major findings are not sufficient merchant establishment to honoring credit cards, cardholders were not aware of all services offered by the issuers, lack of sufficient advertisement and publicity. Her suggestions to overcome these and reduce charges for penalty, interest, annual charges are a much scope for growth and expansions of credit cards market.

Swarnalatha (2002)¹⁵, in her Ph. D. thesis analyses and the results of the credit cards services. The study based on the perception of selected credit cardholders of various issuing banks in Chennai city. This research finding and conclusion based on the hypothesis that single cardholders are less satisfied than multiple cardholders. Further the results also stated that there are cardholders holding cards both in Indian and foreign banks are more aware of the services and experienced their greater satisfaction.

C. Prakash (2003)¹⁶, in his study had focused on the impact of credit cards on spending pattern to study the conceptual aspects and the evolution of credit cards in India. Interview schedule was given to three hundred credit card holders. Simple Percentage, Ftest, ANOVA and Chi-square test were used to analyze the collected data. It was found out that almost all the respondents irrespective of their income levels had agreed that their purchasing power had increased after they had begun to use their credit cards. The accessibility of the credit cards is limited to very small institutions like airlines, jewelers shop and restaurants.

V. K. Anitha (2003)¹⁷, in her study had focused on the purchase behavior of customers, their satisfaction with regard to the services offered by Citi Bank through its agencies and their functioning. The Merchant Establishments should be encouraged to accept more cards. It was concluded that the service centers must maintain good relationship with their customers and they should solve their problems and clarify doubts regarding settlements.

Murugesan (2007)¹⁸, in his Ph. D. thesis entitled “A study on bank credit card culture in Chennai city” analyses that the credit card gives ample scope for the expansion and growth of business. More innovative and user-friendly schemes should be introduced to make more number of people to have credit cards. To reduce the burden of existing cardholders, reduction should be given in the collection of penal interest and other charges, incentive scheme should be introduced to the account holders of the bank willing to get credit card.

Joji, Alex N (2010)¹⁹, the study attempts to understand the impact of retailing and the credit card environment on materialism, compulsive buying, credit card usage pattern and credit default among shoppers in Kochi Municipal Corporation. The goals of the study are to understand the extent of materialism and compulsive buying among shoppers in Kochi Municipal Corporation. The study further attempts to understand whether the presence of credit cards has in any way changed the pattern of spending (usage) and also refine the understanding of what influences (causes) credit card usage pattern. The final accomplishment of the study is to find the extent of influence of the above said variables on credit default.

Mandeep Kaur (2011)²⁰,, perception of cardholders and member establishments towards plastic money in India is a study that examines the perspective of users as well as member establishments has towards the use of plastic money. It focuses on some vital aspects like challenges experienced by customers and bankers, value attribution to plastic-money adaptation and some factors attracting them to adopt and use plastic money. It also analyses the current status and trends of plastic money in India.

3 SOFTWARE REQUIREMENT SPECIFICATION

3.1 Introduction

Financial fraud is an ever growing menace with far reaching consequences in the finance industry, corporate organizations, and government. 16.31 billion dollar were lost in the year 2014 alone due to credit fraud transactions. The credit card fraud detection features uses user behavior and location scanning to check for unusual patterns. These patterns include user characteristics such as user spending patterns as well as usual user geographic locations to verify his identity. If any unusual pattern is detected, the system requires re-verification. The basic goal is to avoid fraudulent transactions and if any one is detected it can be easily traced out with the help of the location.

3.1.1 Project Scope

Further enhancement can be done by making this system secure with the use of certificates for both merchant and customer and as technology changes new checks can be added to understand the pattern of fraudulent transactions and to alert the respective card holders and bankers when fraud activity is identified. The dataset available on day to day processing may become outdated, it is necessary to have updated data for effective fraud behavior identification. To this extent, the incremental approach is necessary in making the system to learn from past as well as present data and capable of handling the both. Fraudster uses different new techniques that are instantaneously growing along with new technology makes it difficult for detection. Three stage authentication process is done to verify the proper user role. Location tracking is implemented to track the user account if any fraud is happened.

3.1.2 User Classes and Characteristics

- To design system which handle online transaction regarding credit card details for secure money transfer without any intruder interface.

- To create a secure computing environment hierarchy for parallel data processing for end user applications.
- To design and implement a new mining technique for generate new block for each transaction.
- To implement a multilevel security system this can validate each peer on every access request to eliminate different fraud attacks.

3.1.3 Assumptions and Dependencies

The user must take good picture of the face for face recognition. OpenCV and Android Studio must be compatible.

3.2 Functional Requirement

3.2.1 Face Recognition

Face Recognition is used for validation of user as it is a very secure way and along with it if the face is not recognised then the photo of the person is captured and sent to police as a fraud person and also along with it a pin verification is done as a added security feature.

3.2.2 Live Location Tracking

Live location can be tracked of the person from where the payment transaction is being done and shortest route for that location is provided with the help of Google Maps API.

3.2.3 Emergency Account Blocking

SMS is sent to the user when an user tries to do payment, if is a genuine person then it can be ignored otherwise if not then an emergency account block is provided so that the transaction can be suspended and fraud does not take place.

3.3 External Interface Requirement

3.3.1 User Interfaces

In this module, User Interface module is developed using Android system. This module is developed to user to identify the credit card fraud using face recognition, transaction pattern and GPS location. So the user interface must be capable of providing the user to upload the dataset and make manipulations and finally must show the user whether fraud has been detected or not. Only final output will be in admin screen.

3.3.2 Hardware Interfaces

Interface between the software product and hardware components. The device types supported would be:

- A display which shows the users some particular information.
- A camera interface which can detect the face and verify for the face recognition

3.3.3 Software Interfaces

Software components include IDEs which the user may use to program the platform to their particular needs. If the user has little or no knowledge about programming then the user can use the videos of android studio or if the user is an experienced programmer then the user can use Android Studio. The IDEs run on the Android. Other components may include databases, third-party softwares/libraries, networking modules, etc. which the user can install, depending on the user's needs and requirements.

3.3.4 Communications Interfaces

- Network server communication protocols: Network protocols are formal standards and policies comprised of rules, procedures and formats that define communica-

tion between two or more devices over a network.

- Http server : The communication between client and server takes place using the Hypertext Transfer Protocol (HTTP).
- TCP/IP: The TCP/IP protocol is designed such that each computer or device in a network has a unique "IP Address" (Internet Protocol Address) and each IP address.

3.4 Non-Functional Requirement

The non-functional requirements are divided into usability, reliability, performance, supportability, safety and security.

3.4.1 Usability

The system must be easy to learn for users. The user interface elements will be easy to understand. There will be help pages which will explain how to achieve common tasks. Error messages should give user specific instruction for recovery.

3.4.2 Reliability

The reliability of the system essentially depends on the software tools and how the image of face is taken. The reliability of user interface has also importance. In case user needs help, user interface should provide it any time. If the user interface crashes, it must be repaired within 30 minutes.

3.4.3 Performance

For desired performance, image capturing, image size, speed of connection, response time and processing speed must be considered. The image processing should be optimized so it should take less time.

3.4.4 Supportability

The system should allow system administrator to add additional features. The system needs to be cost effective to maintain. There should be documents in requirements specification, design and implementation and validation steps. Maintainability requirements may cover diverse levels of documentation, such as system documentation, requirements may cover diverse levels of documentations such as system documentation, as well as test documentation e.g. which test cases and test plans will accompany

the system. Diagrams should be provided in the documents in order to improve the understanding of stakeholders and developers.

3.4.5 Safety

In case malfunction, system should shutdown itself and reboots in order to prevent unpredicted results.

3.4.6 Security

System should store user data on database securely and set access permissions to the this data carefully.

3.5 System Requirement

3.5.1 Database Requirement

- It should be MySql database on platform.
- Database must be integrated with key constraints.
- It should be maintain the relational base on RDMS and normalization.
- System will create database backup on periodic basis.
- It will execute all commands like DML, DDL and DCL as well as we required some security measurements for Sql injection.

3.5.2 Software Requirement

- Operating system : Windows/Linux
- Front End : JAVA, Android
- Tool : Android Studio

3.5.3 Hardware Requirement

- SYSTEM : Processor:- Intel Pentium 4 or above
- HARD DISK : 40 GB
- MONITOR : 15 VGA colour
- RAM : 4 MB
- KEYBOARD: 110 keys enhanced.

3.6 Analysis Model: SDLC Model to be applied

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like –

- Planning
- Requirements Analysis
- Analysis and Design
- Implementation
- Unit Testing
- Acceptance Testing

At the end of the iteration, a working product is displayed to the customer and important stakeholders.

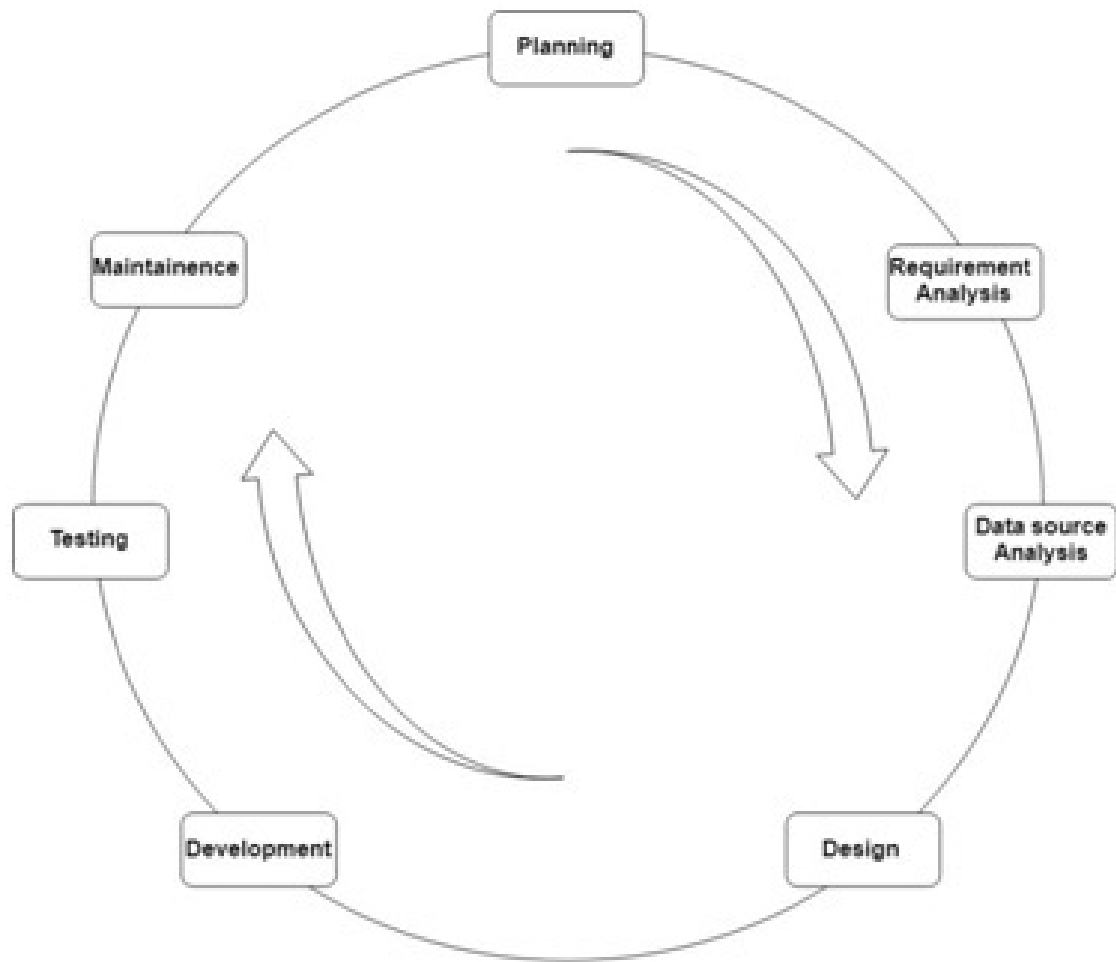


Figure 3.1: SDLC Analysis Model

3.7 System Implementation Plan

Schedule		Date	Project Activity
July	1 st Week	08/07/2019	Project Topic Selection
	2 nd Week	15/07/2019	Synopsis Submission
August	1 st Week	05/08/2019	Presentation of Project Ideas
	2 nd Week	12/08/2019	Submission of Literature Survey
	3 rd Week	19/08/2019	Feasibility Assessment
September	1 st Week	02/09/2019	Mid Sem Preparation
	3 rd Week	16/09/2019	Design of Mathematical Model
	4 th Week	23/09/2019	End Sem Presentation
October	1 st Week	07/10/2019	Report Preparation and Submission
December	3 rd Week	19/12/2019	1 st module presentation
	4 th Week	26/12/2019	Discussion and implementation of 2 nd module
January	1 st Week	02/01/2020	Preparation of cpgon
	2 nd Week	09/01/2020	Study of prequery algorithm
	3 rd Week	16/01/2020	Discussion about modification improved accuracy
	4 th Week	23/01/2020	1 st and 2 nd module presentation
	5 th Week	30/01/2020	Discussion on flow of project designing new module
February	1 st Week	06/02/2020	Modification of modules
	2 nd Week	13/02/2020	Designed test cases for our modules
	3 rd Week	20/02/2020	Work on user interface
March	1 st Week	06/03/2020	Integration of all modules
	3 rd Week	20/03/2020	Final Report and presentation

Figure 3.2: System Implementation Plan

4 SYSTEM DIAGRAM

4.1 System Architecture

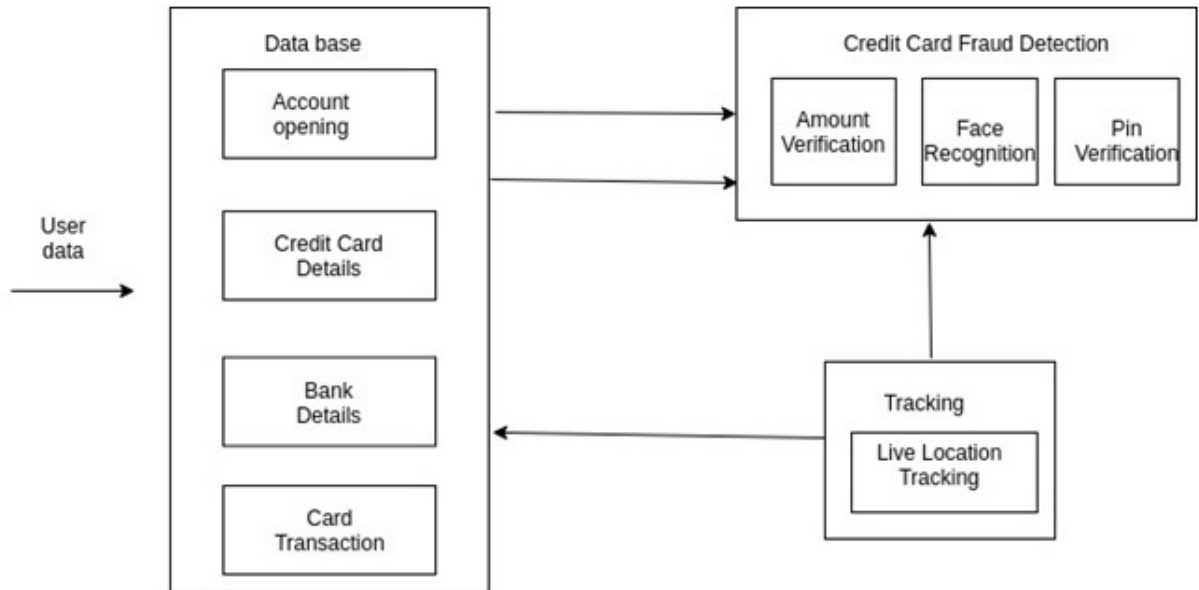


Figure 4.1: System Architecture

4.2 Data Flow Diagram

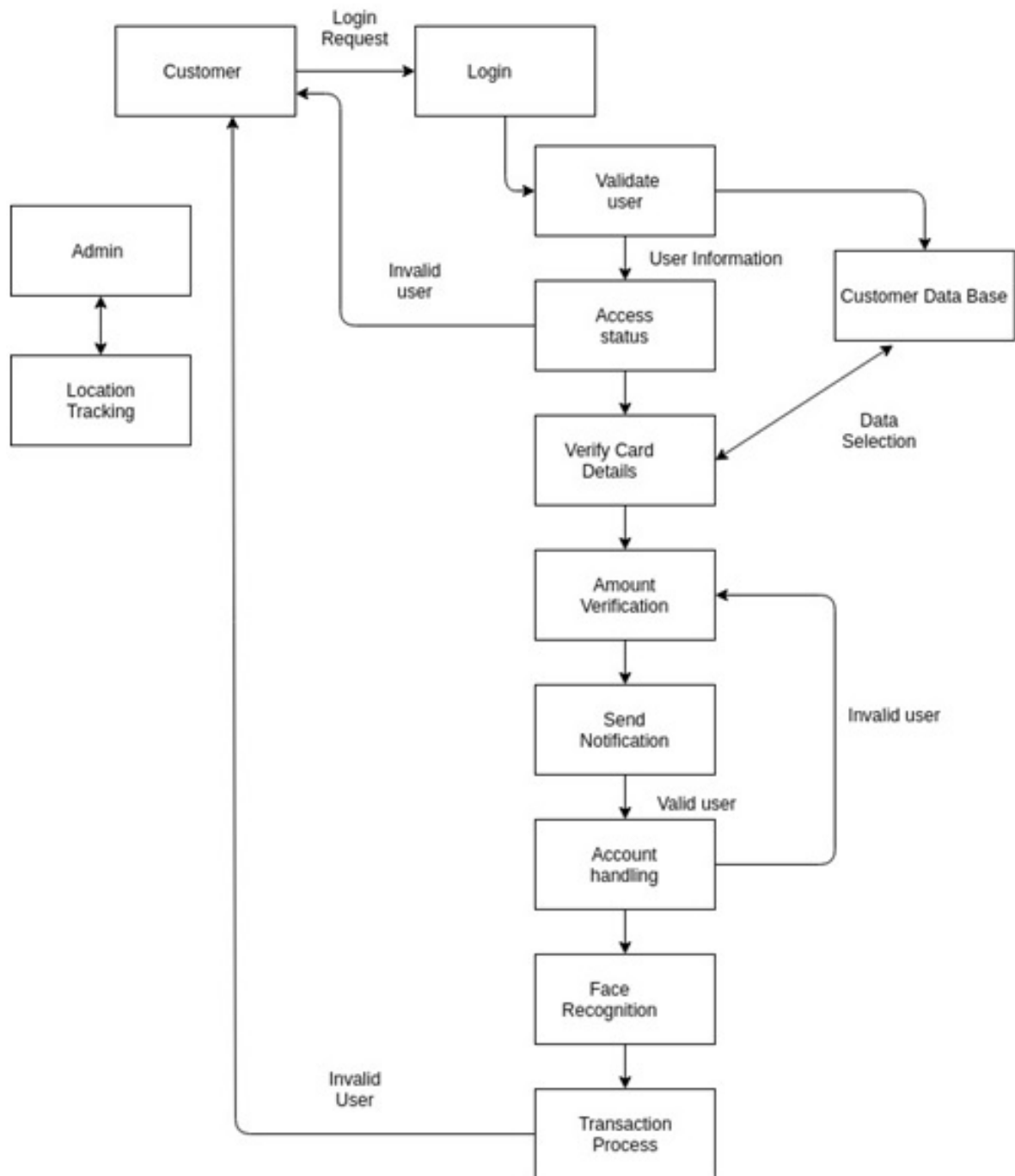


Figure 4.2: Data Flow Diagram

4.3 UML Diagrams

4.3.1 Use Case Diagram

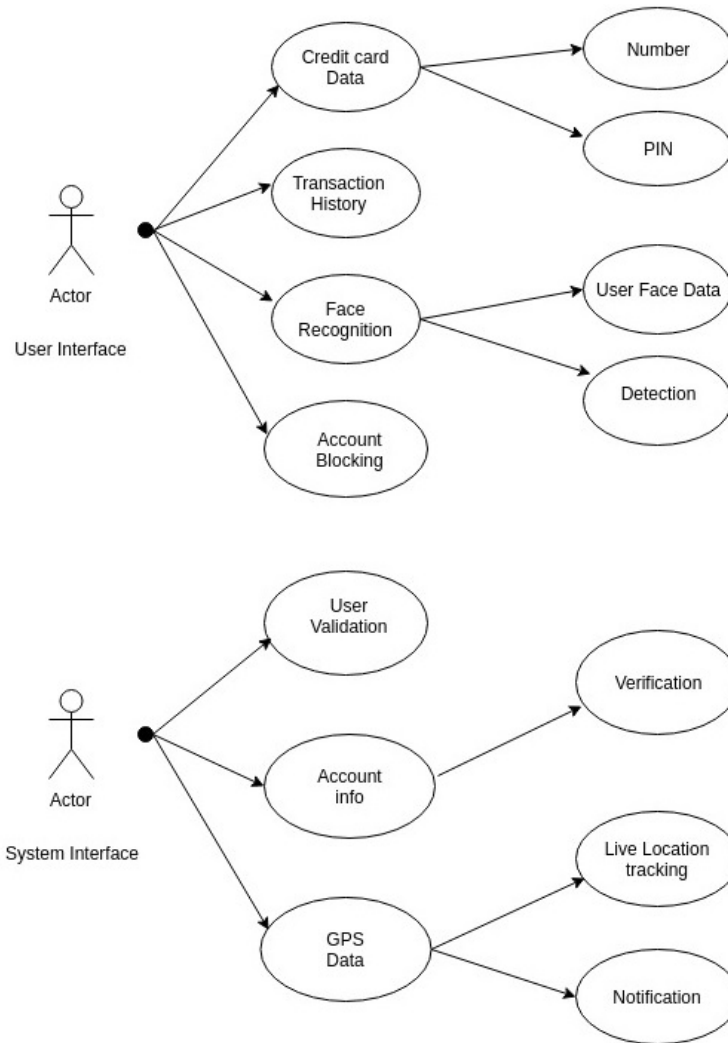


Figure 4.3: Use Case Diagram

A use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform. In this context, a "system" is something being developed or operated, such as a web site. The "actors" are people or entities operating under defined roles within the system.

4.3.2 Class Diagram

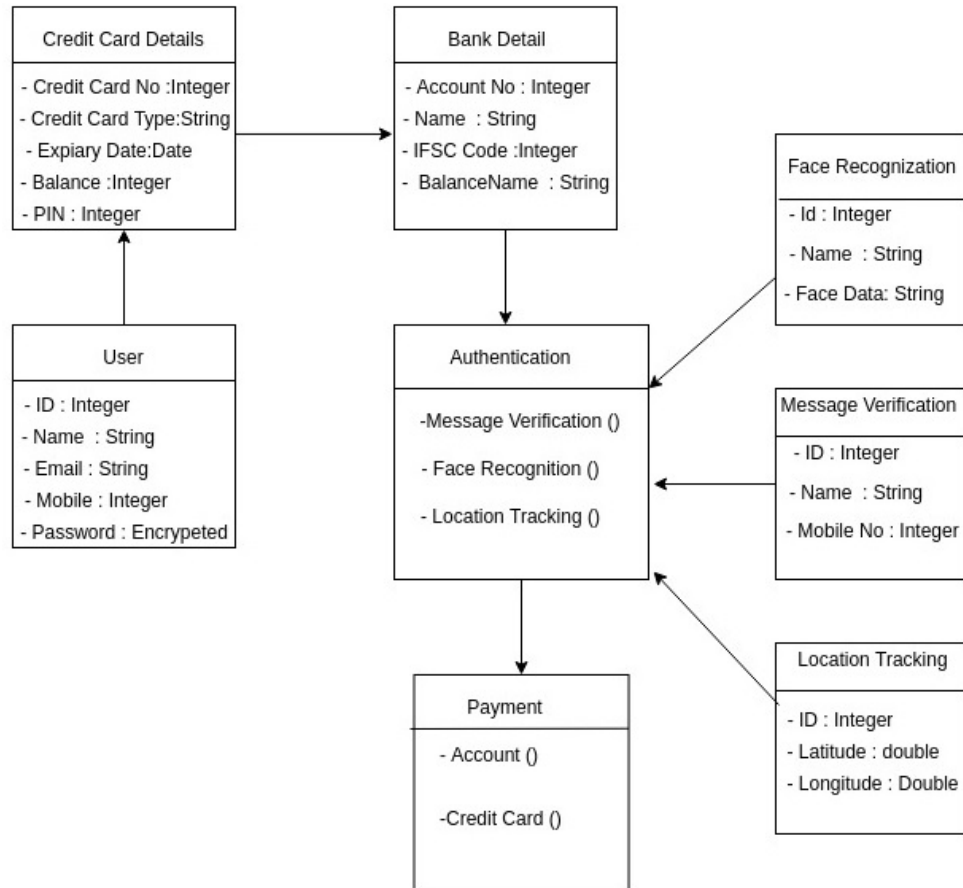


Figure 4.4: Class Diagram

Class Diagram gives the static view of an application. A class diagram describes the types of objects in the system and the different types of relationships that exist among them. This modeling method can run with almost all Object-Oriented Methods. A class can refer to another class. A class can have its objects or may inherit from other classes. UML Class Diagram gives an overview of a software system by displaying classes, attributes, operations, and their relationships. This Diagram includes the class name, attributes, and operation in separate designated compartments.

4.3.3 Sequence Diagram

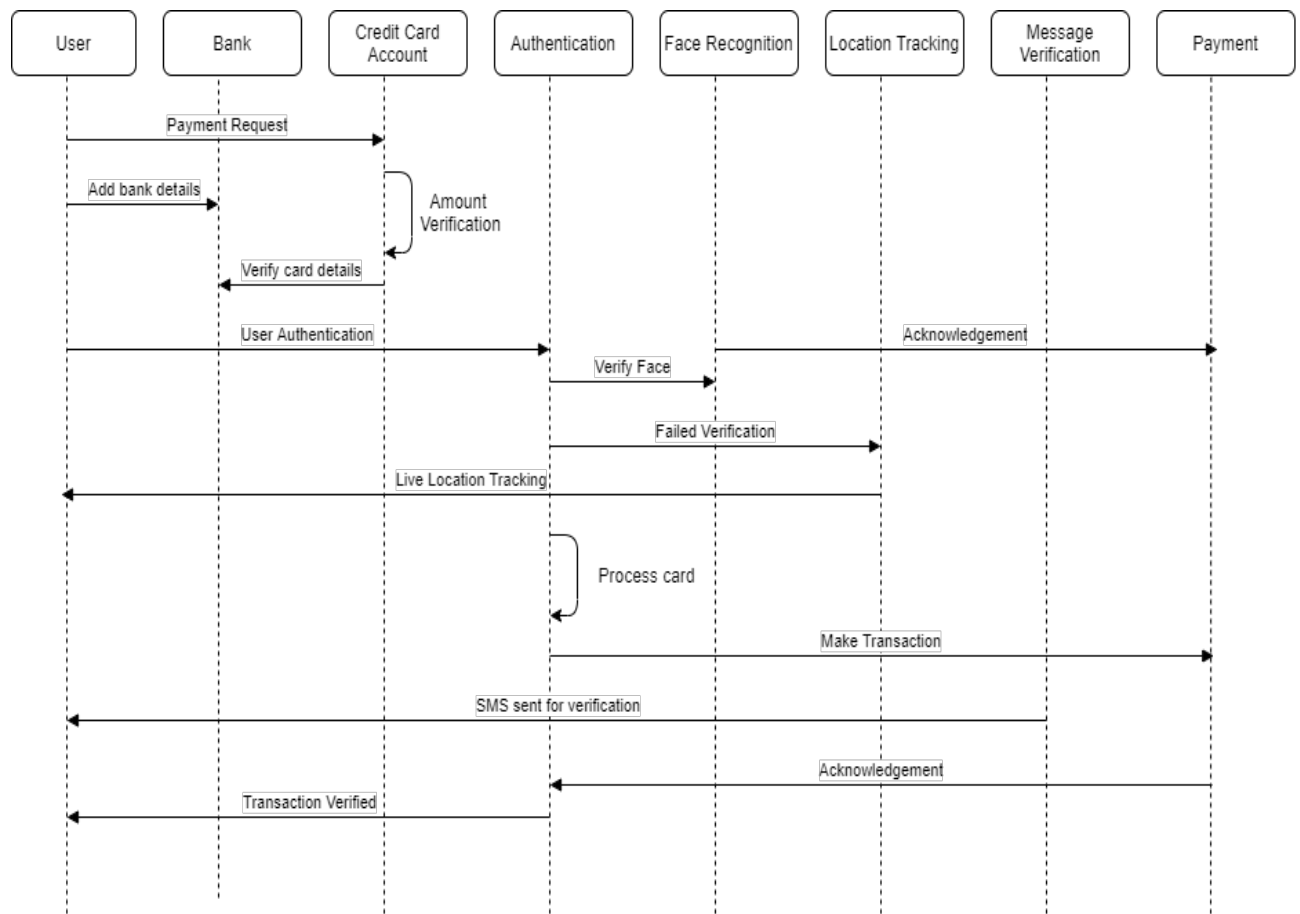


Figure 4.5: Sequence Diagram

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

4.3.4 Deployment Diagram

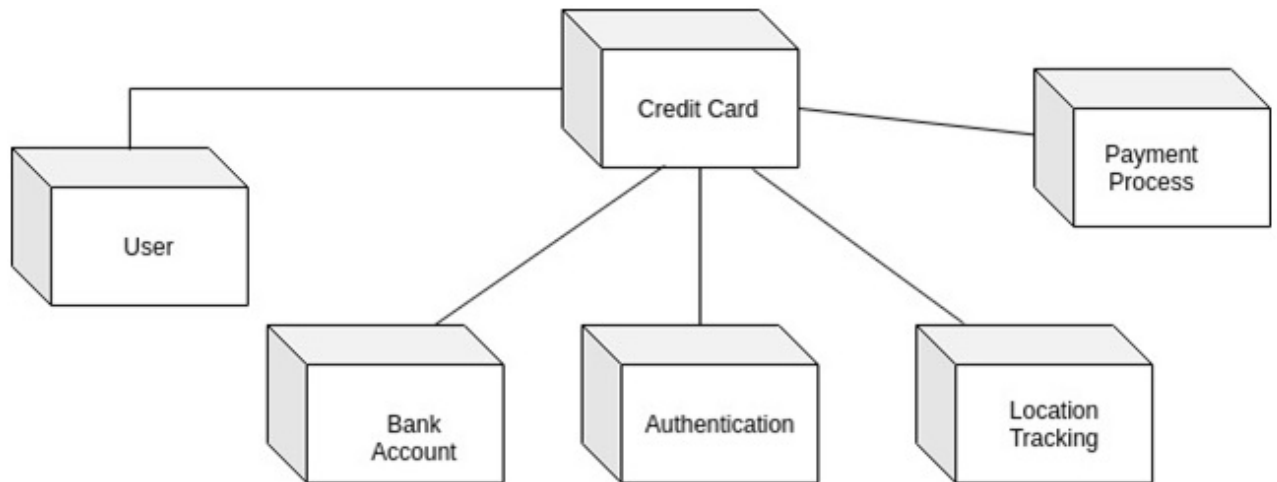


Figure 4.6: Deployment Diagram

Deployment diagram is a diagram that shows the configuration of run time processing nodes and the components that live on them. Deployment diagrams are a kind of structure diagram used in modeling the physical aspects of an object-oriented system. They are often be used to model the static deployment view of a system (topology of the hardware).

4.3.5 Activity Diagram

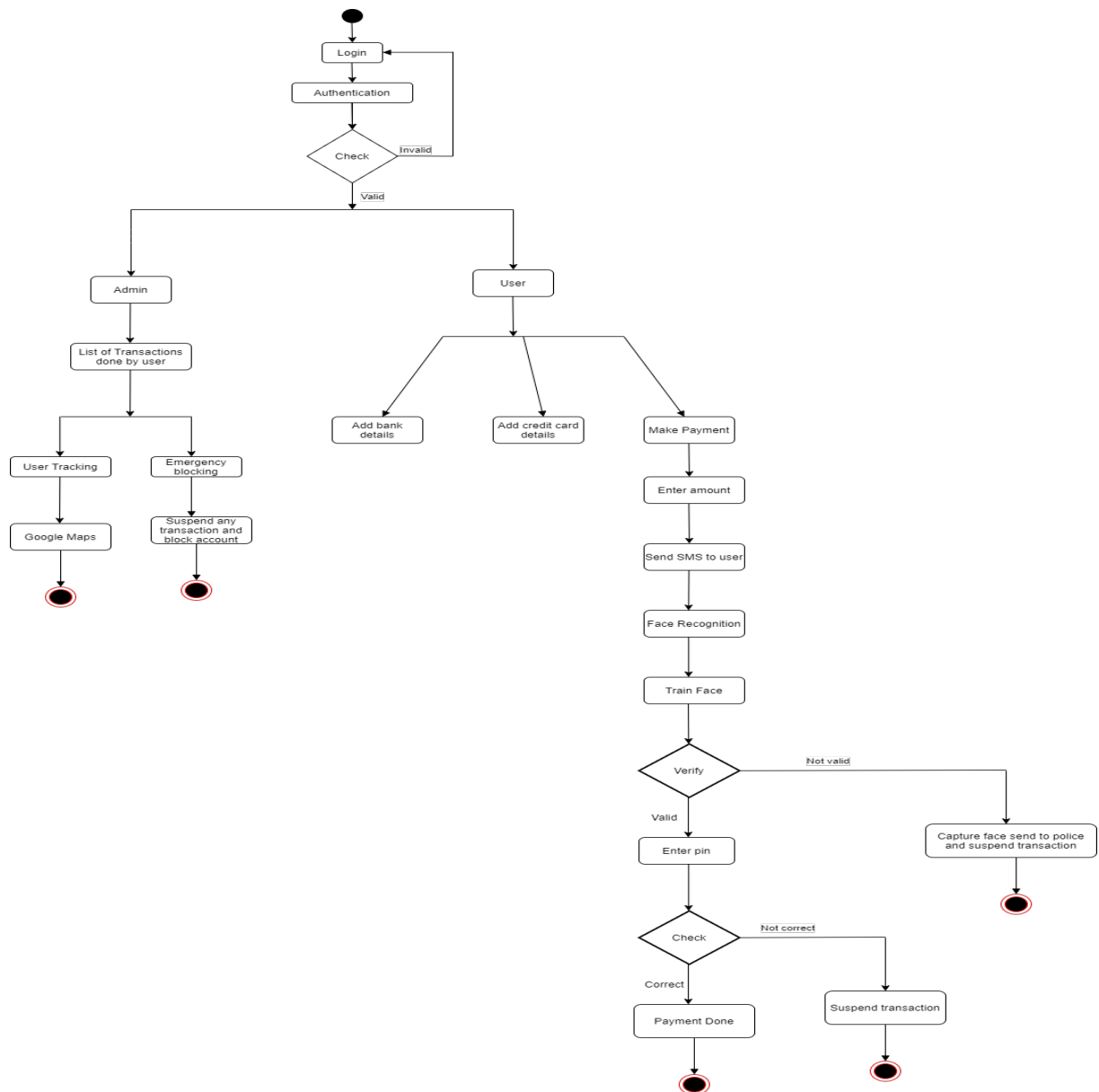


Figure 4.7: Activity Diagram

Activity diagram is defined as a UML diagram that focuses on the execution and flow of the behavior of a system instead of implementation. It is also called object-oriented flowchart. Activity diagrams consist of activities that are made up of actions which apply to behavioral modeling technology.

5 Project Plan

5.1 Project Estimate

5.1.1 Reconciled Estimate

Cost Estimate

Cost will estimate after completing the project that depend on time to complete the project and also efforts required to complete the project.

Time Estimate

Time will depend on modules of project and also project plan of execution.

5.1.2 Project Resources

1. Hardware Resources Required

System : Processor:- Intel Pentium 4 or above, Hard Disk : 40GB, Monitor : 15 VGA colour, Ram : 4 MB, Keyboard : 110 keys enhanced.

2. Software Resources Required

Operating System: Windows/Linux, Front End: JAVA,Android Tool: Android Studio.

5.2 Risk Management

5.2.1 Risk Identification

For risks identification, review of scope document, requirements specifications and schedule is done. Answers to questionnaire revealed some risks.Each risk is categorized as per the categories mentioned in [?]. Please refer the table ?? for all the risks. Following questions can be referred

Description	Low	High
Login	No	Yes
Internet Connection Slow	Yes	No
Face Recognition	Yes	No

Table 1: Risk Analysis

1. Have top software and customer managers formally committed to support the project?

Answer: Yes , top software and customer managers are formally committed to support the project.

2. Are end-users enthusiastically committed to the project and the system/product to be built?

Answer: Yes, end-users enthusiastically committed to the project and the system/product to be built.

3. Are requirements fully understood by the software engineering team and its customers?

Answer: Yes, Are requirements fully understood by the software engineering team and its customers.

4. Have customers been involved fully in the definition of requirements?

Answer: Yes, customers been involved fully in the definition of requirements.

5. Are project requirements stable?

Answer: All project requirements are stable.

6. Is the number of people on the project team adequate to do the job?

Answer: Yes, the number of people on the project team adequate to do the job.

5.2.2 Risk Analysis

Following are the details for each risk.

Risk ID 1

Description 1: While login to system validation is there so user must follow rules and proper entries.

Risk ID 2

Description 2: While registering to system internet connection should be fast for registration to get users data online.

Risk ID 3

Description 3: While taking face recognition the room must be well lit.

5.3 Project Schedule

5.3.1 Project Task Set

Major Tasks in Project Stages are :

- Task 1 : Requirement Analysis(Base Paper Explanation).
- Task 2 : Project Specification.
- Task 3 : Technology Study and Design.
- Task 4 : Coding and Implementation(Module Development).

5.3.2 Task Network

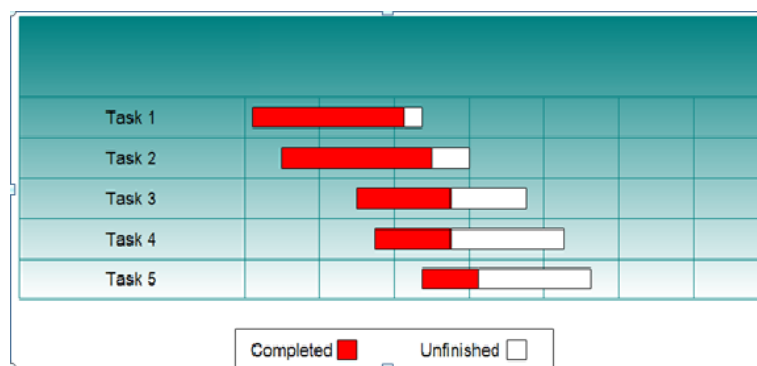


Figure 5.1: Task Network Diagram

5.3.3 Timeline Chart

Activity	I Week	II Week	III Week	IV Week	V Week	VI Week	VII Week	VIII Week	IX Week	X Week
	Aug 4 th	Sep 11 th	Oct 18 th	Nov 25 th	Dec 1 st	Jan 8 th	Jan 15 th	Feb 22 nd	March 29 th	April 10 th
Initiate the project										
Communication										
Literature survey										
Define scope										
Define SRS										
Plan the project										
Design mathematical model										
Feasibility Analysis										
Develop work breakdown structure										
Planning project schedule										
Design UML and other diagrams										
Design test plan										
Design risk management plan										
Deployment of project										
Testing of project										
Documentation										

Figure 5.2: Timeline Chart

5.4 Team Organization

5.4.1 Team structure

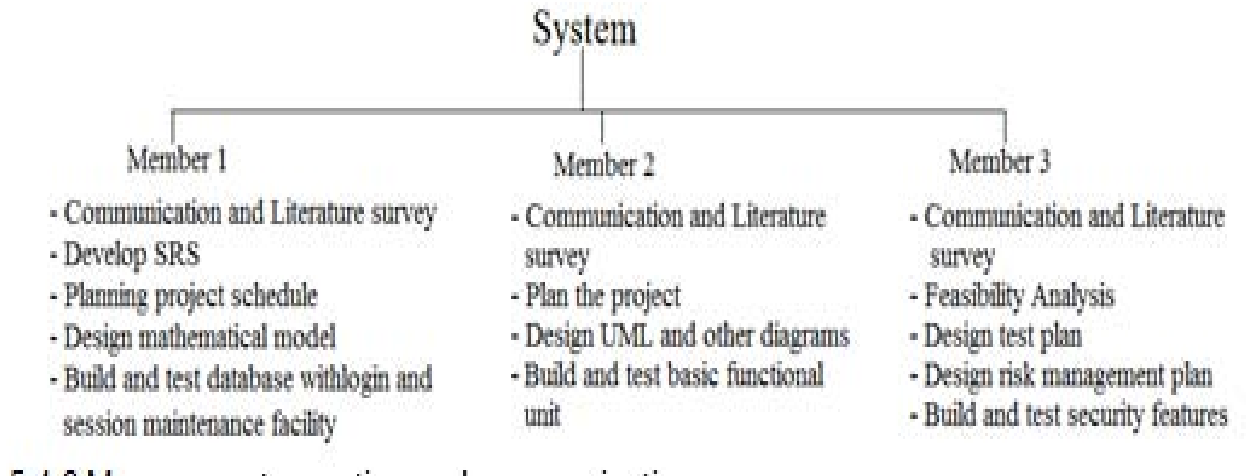


Figure 5.3: Team structure

5.4.2 Management reporting and communication

Sr No.	Month	Description
1.	June	Discussion with guide regarding domain. Searching for IEEE paper for domain.
2.	July	Short listing of IEEE papers within domain. Selection of IEEE paper.
3.	August	Deciding project name. Submission of synopsis
4.	September	Requirement Analysis Designing of models
5.	October	Report preparation Stage-I report submission
6.	November	1 st module presentation Discussion and implementation of 2 nd module
7.	December	Preparation of codes Study of prequery algorithm
8.	January	Discussion about modification to improve accuracy Discussion on flow of project and designing new module
9.	February	Modification of modules Designing test cases for our modules
10.	March	Work on user interface
11.	April	Integration of all modules Final Report and presentation

Figure 5.4: Management reporting and communication

6 Project Implementation

6.1 Overview of Project Module

1. Admin Module:

Role of admin is to check the live payment transaction and if any suspicious is found then the live location of the place from where the transaction took place can be traced with the help of google maps. Also if it is truly suspicious then the account can be blocked so that the transaction can be suspended then and there only.

2. User Module:

Role of user is to save the bank details and also set a pin for the transaction. Face should be trained for the transaction and face recognition can be used by the user after that the pin which can be used so it is a two way verification and then payment is done. SMS is also sent during the once the payment amount is put before face recognition so to check fraud transaction or not.

3. Bank Module:

A bank has been set to maintain all the details of the account of the bank and to check the deduction in the amount entered during the payment and all the bank related details like IFSC code,etc are maintained to show the working of the bank.

6.2 Tools and Technologies Used

6.2.1 Technology Description

Android Studio is the official integrated development environment (IDE) for Android application development. To support application development within the Android operating system, Android Studio uses a Gradle-based build system, emulator, code templates, and Github integration. It has an inbuilt Kotlin and Java language for the development. XML is also an inbuilt in Android Studio used for the designing of the application.

PHP is a popular general-purpose scripting language that is especially suited to web development. PHP includes various free and open-source libraries in its source distribution, or uses them in resulting PHP binary builds. PHP is fundamentally an Internet-aware system with built-in modules for accessing File Transfer Protocol (FTP) servers and many database servers, including PostgreSQL, MySQL, Microsoft SQL Server and SQLite (which is an embedded database), LDAP servers, and others.

6.2.2 Hardware Specifications

- SYSTEM : Processor:- Intel Pentium 4 or above
- HARD DISK : 40 GB
- MONITOR : 15 VGA colour
- RAM : 4 MB
- KEYBOARD: 110 keys enhanced.

6.2.3 Software Specifications

- Operating system : Windows/Linux
- Front End : JAVA, Android
- Back End : MySql 5.0
- Tool : Android Studio

6.3 Algorithm Details

6.3.1 CNN Algorithm

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to

other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.

The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlap to cover the entire visual area.

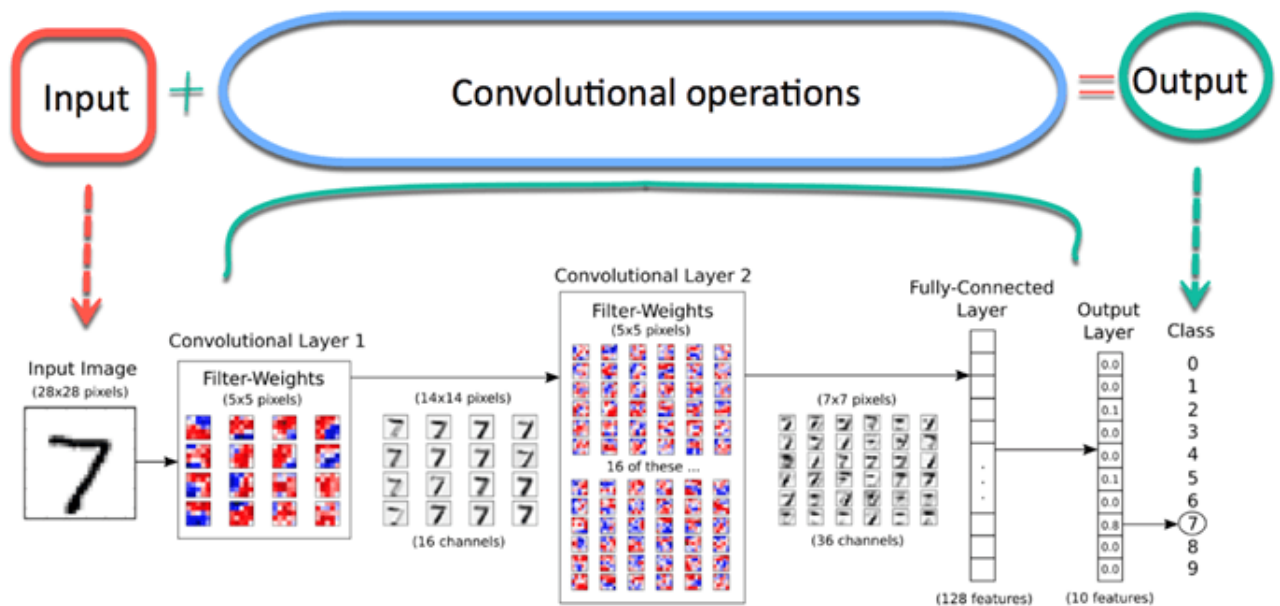


Figure 6.1: CNN Algorithm

7 Software Testing

7.1 Types of Testing

7.1.1 Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

7.1.2 Integration Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

7.1.3 Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

7.1.4 White Box Testing

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

7.1.5 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box you cannot see into it. The test provides inputs and responds to outputs without considering how the software works.

7.1.6 Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

7.1.7 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or one step up software applications at the company level interact without error.

7.1.8 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements. Test Results: All the test cases mentioned above passed successfully.No defects encountered.

7.1.9 Alpha and Beta Testing

Alpha Testing is a type of software testing performed to identify bugs before releasing the product to real users or to the public. Alpha Testing is one of the user acceptance testing.Beta Testing is performed by real users of the software application in a real environment. Beta testing is one of the type of User Acceptance Testing.

7.1.10 System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

7.2 Test Case and Test Results

Testing of project problem statement using generated test data (using mathematical models, GUI, Function testing principles, if any) selection and appropriate use of testing tools, testing of UML diagrams reliability.

Module-ID:-01

Modules to be tested:- Registration

1. Enter the case insensitive Username click on Submit button.
Expected: It should display an error.
2. Enter the case sensitive Username click on Submit button.
Expected: It should accept.
3. Enter the case insensitive password click on Submit button.
Expected: It should display an error.
4. Enter the case sensitive password click on Submit button.
Expected: It should accept.
5. Enter the case insensitive mobile number click on Submit button.
Expected: It should display an error.
6. Enter the case sensitive mobile number click on Submit button.
Expected: It should accept.

Test Case ID	Description	Test Case I/P	Expected Result	Actual Result	Test case criteria P/F
001	Enter the case insensitive Username click on Submit button	UserName	Error should come	Error comes	P

002	Enter the case sensitive User-name click on Submit button	UserName	It should accept	Accepted	P
003	Enter the case insensitive password click on Submit button	Password	Error should come	Error comes	P
004	Enter the case sensitive password click on Submit button	Password	It should accept	Accepted	P
005	Enter the case insensitive mobile number click on Submit button	Mobile No	Error should come	Error comes	P

006	Enter the case sensitive mobile number click on Submit button	Mobile No	It should be accepted	Accepted	P
-----	---	-----------	-----------------------	----------	---

Table 2: Test Cases

Module ID:-2

Modules to be tested:- Login

1. Enter the correct username and wrong password click on Submit button.
Expected : It should display error.
2. Enter the wrong username and correct password and click on Submit button.
Expected : It should display error.
3. Enter the correct username and password and click on Login button.
Expected : It should display welcome page
4. After login with valid credentials click on back button.
Expected : The page should expire.
5. Check the password with Lower case and upper case.
Expected : Password should be case sensitive.

Test Case ID	Description	Test Case I/P	Expected Result	Actual Result	Test case criteria P/F
--------------	-------------	---------------	-----------------	---------------	------------------------

001	Enter the correct username and wrong password click on Submit button	Username Password	Error should come	Error comes	P
002	Enter the wrong username and correct password and click on Submit button	Username Password	Error should come	Error comes	P
003	Enter the case insensitive password click on Submit button	Password	It should accept	Accepted	P

Table 3: Test Cases

Module ID:- 3

Modules to be tested : Face Recognition

1. Correct Face in well lit area
Expected : Face Recognition is done
2. Correct Face in dark area
Expected : Face not recognised
3. Face with rashes after training face
Expected : Face should be recognised

Test Case ID	Description	Test Case I/P	Expected Result	Actual Result	Test case criteria P/F
001	Correct Face in well lit area	Face	Face should be accepted	Face Accepted	P
002	Correct Face in dark area	Face	Error should come	Error comes	P

Table 4: Test Cases

8 Results

8.1 Outcomes

An app developed for the payment transaction during online payment or shopping. We employ the real time location of the transaction as well as the face recognition is good and also the account can be blocked on real time basis. Also if face recognition is overcome then there is a pin verification is done. We also show the shortest route to the place from where the live transaction is going on. We recommend this app for payment.

8.2 Screenshots

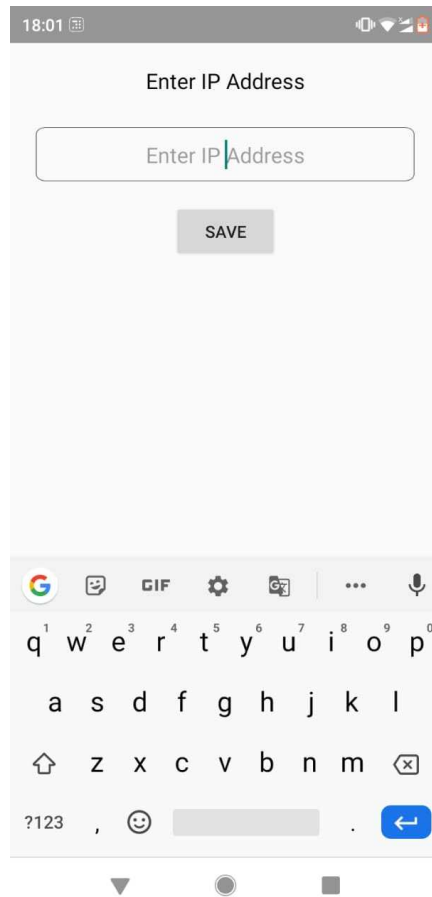
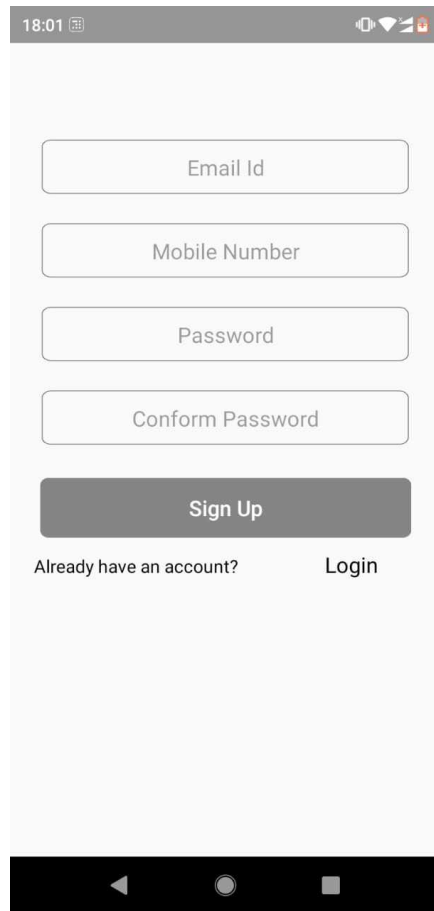


Figure 8.1: IP Address



A mobile application registration screen. At the top, a status bar shows the time 18:01 and various icons. The main area contains four white input fields with rounded corners, stacked vertically, labeled "Email Id", "Mobile Number", "Password", and "Conform Password". Below these is a dark grey button with the text "Sign Up" in white. Under the button, the text "Already have an account?" is followed by a "Login" link. The bottom of the screen features a black navigation bar with three white icons: a back arrow, a home circle, and a recent apps square.

Figure 8.2: Registration

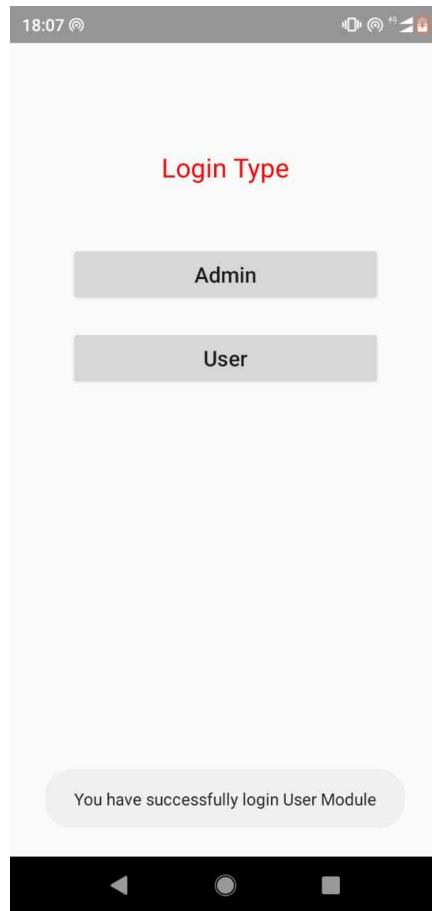


Figure 8.3: Welcome Page

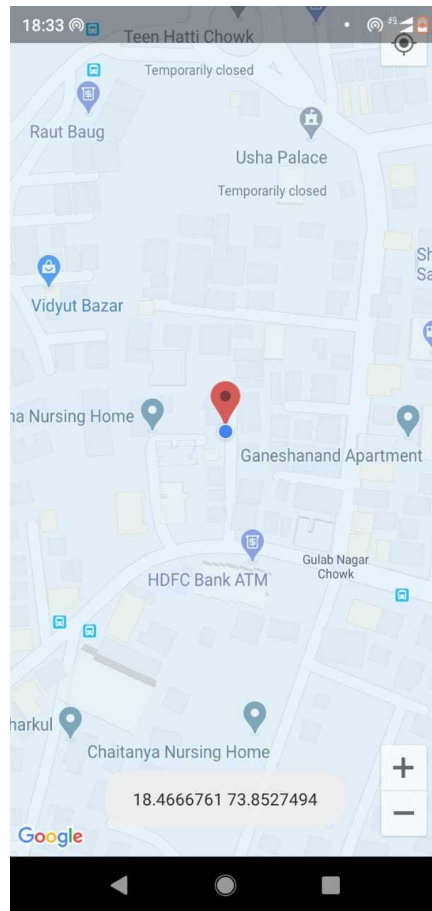


Figure 8.4: Google Maps

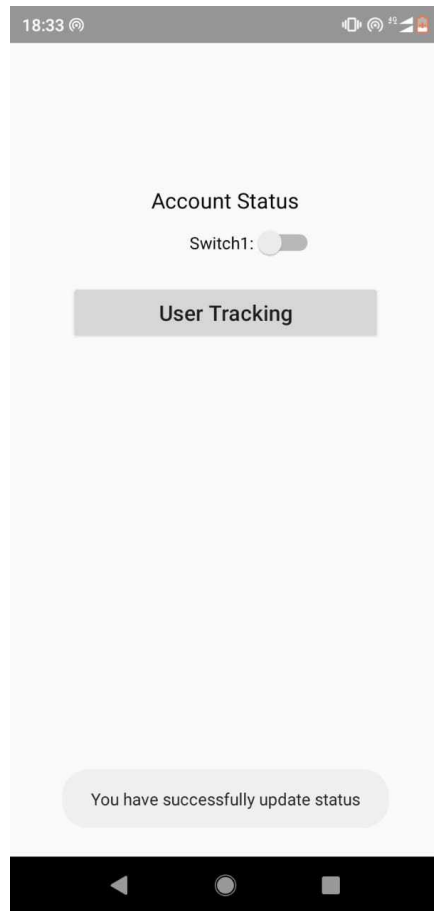


Figure 8.5: User Tracking and Blocking

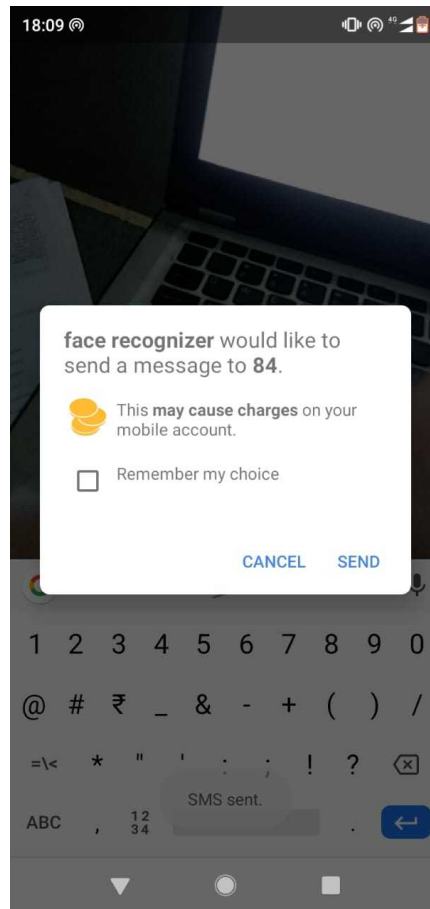


Figure 8.6: Face Recognition

9 Conclusion and Future Work

9.1 Conclusion

This project purpose is to find out the fraud transactions done by fraud users with high accuracy security system. By using the user behavior methodology it's easy to detect the pattern of transaction to detect the transaction purpose. Make every online transaction more and more secure with multiple authentication system. This System track the fraud used using location tracking system with Google map API. Account blocking facility is added in application to avoid fraud transactions. In particular, since there is no limit on the number of features that can be calculated, a system may take too long to make a decision based on the time spent recalculating the features with each new transaction.

9.2 Future Work

From the above comparative analysis of the credit card fraud detection techniques it is clear that multiple authentication system is used best in this scenario. In order to minimize their expense we need to create a hybrid of neural network with some optimisation technique. Optimisation techniques that could be successfully paired with Neural Network are Genetic Algorithm, Artificial Immune System, Case Based Reasoning and any other similar optimisation technique. Genetic Algorithm helps by selecting the optimised weight of the edges in neural network. Artificial Immune System reduces the cost by eliminating the weights that cause the maximum error and Case Based Reasoning first tries to predict the outcome on the basis of a direct match with the user's profile.

9.3 Applications

1. Online Transactions
2. Banking
3. Mobile UPI system

Appendix A

Feasibility Assessment

Operational Feasibility

Operational feasibility is a measure of how well a proposed system solves the identified problems, and takes advantage of the opportunities identified in the scope of research. The system should also satisfy the requirements identified in the requirement analysis phase of system development. Since the software is intended at making operations easy faster, the intended user must have the fundamental knowledge of the computer, and operating systems environment. The proposed system is intended at providing a high level of abstraction to the user so that even any person with average knowledge of concrete construction and user interfaces should be compatible to use it and can analyze fraud efficiently. Moreover, the part of proposed system works as a background operation during the user's usage of the interface, makes user's interaction to a minimum level. Since all these requirements are easy and affordable, the random number generator system is operationally feasible.

Technical Feasibility

The feasibility study is an influencing factor that contributes to the analysis of system implementation. The consideration of whether to design and implement a particular system depends on the feasibility study on a targeted system. Technical feasibility study is carried out to determine whether the proposed system has the capability, in terms of software, hardware, personnel, and expertise to handle the completion of a task.

Economic Feasibility

Economic Feasibility is concerned with the cost incurred for development and implementation of the proposed system, the maintenance of the system, and the benefits derived from it. The software required for the implementation of the proposed system is open source and for the implementation we are taking already available data set. So cost incurred is comparatively less.

There are two types of problems:

Polynomial (P): The Proposed system accepts input, and we get the output in fixed polynomial time, the input format is fixed so that our system takes fixed computational time. Applications of polynomial type are rare. One such example is hash table. The hash table finds index for a data to be inserted in fixed amount of time because it uses hash function to find index. So, for finding index 1 or 100 the time is fixed which is not in case of sequential search for index.

Non-Polynomial (NP): There are two sub-types of NP Problems-

NP-Hard: The system accepts input, but there is no guarantee that we will get the output. Such systems do not exist because no one will use the system if there is no guarantee the system works for any inputs. Hence our application is again not of NP-Hard type because we want to build a system that never fails and guarantees output. Example: Turing Machine Halting problem.

NP-Complete: The system accepts input, and we get the output in variable non-polynomial time. Almost all or maximum systems are of NP Complete type our application is not NP complete type because it guarantees output but in fixed amount of time. The time required for the output is directly proportional to the size of the input. So, our output time doesn't vary with the input.

Problem Type

The system is of NP complete type. The proposed system guarantees removal of fraud detection and even if the fraud is found in the face recognition and then the image is captured and the location is sent to police. Outcome of the system will be achieved in fixed time period. From above paragraph we can say that the problem is of polynomial type.

Appendix C

Plagiarism Report

1. 4,557 words.
2. 35,060 characters
3. 96% unique.
4. Plagiarism check done by <https://www.plagiarizechecker.com>.

References

1. W. van der Aalst, T. Weijters and L. Maruster, "Workflow mining: Discovering process models from event logs," *IEEE Trans. Knowl. Data Eng.*, vol. 16, no. 9, pp. 1128–1142, Sep. 2004.
2. *IEEE Spectrum* (2014, August 7). Black Hat 2014: A New Smartcard Hack. Available: <http://spectrum.ieee.org/riskfactor/telecom/security/black-hat-2014-howto-hack-smartcards-and-terms-of-service>.
3. Steven J. Murdoch, Saar Drimer, Ross Anderson, and Mike Bond, "Chip and PIN is Broken" in *IEEE Symposium on Security and Privacy*, 2010 ©. DOI 10.1109/SP.2010.33. pp 433 – 446
4. Haiying Ma Xin Li, "Application of Data Mining in Preventing Credit Card Fraud" in *International Conference on Management and Service Science*, 2009 ©, DOI: 10.1109/ICMSS.2009.5304330, pp 1 – 6
5. Wen-Fang Yu Na Wang, "Research on Credit Card Fraud Detection Model Based on Distance Sum" in *International Joint Conference on Artificial Intelligence*, 2009 ©, DOI: 10.1109/JCAI.2009.146, pp 353 – 356.
6. A. Abdallah, M. A. Maarof, and A. Zainal, "Fraud detection system: A survey," *J. Netw. Comput. Appl.*, vol. 68, pp. 90–113, Jun. 2016.
7. N. Abdelhamid, A. Ayesh, and F. Thabtah, "Phishing detection based associative classification data mining," *Expert Syst. Appl.*, vol. 41, no. 13, pp. 5948–5959, 2014.
8. Manahil Raza Usman Qayyum Classical and Deep Learning Classifiers for Anomaly Detection, *Proceedings of 2019 16th International Bhurban Conference on Applied Sciences Technology (IBCAST)*.