

CREDIT CARD FRAUD DETECTION

BY

M. Sakthi Karthick Raja	192132
K. Sivaganesh	192138
K. Thiruppathy	192143
G. Maniraj	202207
M. Ranjith	202212
P. Sibiraj	202214

A Project Report

Submitted to the

DEPARTMENT OF COMPUTER ENGINEERING

In partial fulfillment for the award of

DIPLOMA

In

COMPUTER ENGINEERING

GUIDED BY

Mrs. K.Sindhuja B.E.,



SANKAR POLYTECHNIC COLLEGE (AUTONOMOUS)

SANKAR NAGAR-627 357

APRIL-2022

CREDIT CARD FRAUD DETECTION

BY

M.Sakthi Karthick Raja	192132
K.Sivaganesh	192138
K.Thiruppathy	192143
G.Maniraj	202207
M.Ranjith	202212
P.Sibiraj	202214

A Project Report

Submitted to the

DEPARTMENT OF COMPUTER ENGINEERING

In partial fulfillment for the award of

DIPLOMA

In

COMPUTER ENGINEERING

GUIDED BY

Mrs. K.Sindhuja B.E.,



SANKAR POLYTECHNIC COLLEGE (AUTONOMOUS)

SANKAR NAGAR-627 357

April-2022

SANKAR POLYTECHNIC COLLEGE (AUTONOMOUS)
SANKAR NAGAR-627 357

BONAFIDE CERTIFICATE

Certified that this report titled " **CREDIT CARD FRAUD DETECTION** " for the project is a bonafide work on _____ who carried out the work under my supervision for the partial fulfillment of the requirements for the award of the diploma in Computer Engineering The academic year 2021-2022

Signature of Guide
Mrs. K. Sindhuja B.E.,

Signature of HOD
Mr.S.Ramesh M.E.,

Submitted to the Project viva-voce held on _____

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

First, we would like to express our heartfelt thanks to Almighty God for his perpetual blessings.

We take this opportunity to express our sincere gratitude to our Principal –

Dr. G. SANKARASUBRAMANIAN B.E. M.S., Ph.D., for his encouragement given to carry out this project work

We take this opportunity to express our sincere gratitude to the head of the department –

Mr. S. RAMESH M.E., for his unique holistic approach and his encouragement, thought the year

I would like to express my sincere gratitude and thanks to my project coordinator –

Mr. BALA MURUGEN B.E., and guide – **Mrs. K. Sindhuja B.E.,** for their Motivation. Inspiration and Encouragement to undertake this work

We take this opportunity to thank all the staff members Department of Computer Engineering staff members for their support and suggestions given to our project.

Last but not least we would like to thank our parents and friends for their wholehearted support help and prayers throughout our course work and project work.

CONTENTS

CONTENTS

S.NO	CHAPTER	PAGE NO
1	Synopsis	1
2	Requirement Analysis	3
3	Project Description	5
4	Introduction to HTML5, CSS3, Bootstrap, js	10
5	Introduction to PYTHON, FLASK	14
6	Software Design	22
7	Software Implementation	28
8	Software Testing	30
9	Future Enhancement	34
10	Conclusion	36
11	Appendix1	38
12	Appendix2	46
13	Bibliography	52
14	Project Members	54

SYNOPSIS

SYNOPSIS

Due to rapid advancement in the electronic commerce technology, the use of credit cards has dramatically increased. Since credit card is the most popular mode of payment, the number of fraud cases associated with it is also rising. Thus, in the order to stop these frauds we need a powerful fraud detection system that detects it in an accurate manner. In this paper we have explained the concept of frauds related to credit cards. An extensive review is done on the existing and proposed models for credit card fraud detection and has done comparative study on these techniques.

REQUIREMENT ANALYSIS

SYSTEM SPECIFICATION

HARDWARE REQUIREMENTS

Processor	:	AMD Processor
Processor speed	:	2.50GHZ
RAM	:	4GB
Monitor	:	14' color monitor
Hard Disk	:	100GB

SOFTWARE REQUIREMENTS

Operating System	:	Windows10
Front End	:	HTML5, CSS3, Bootstrap, JS
Back End	:	Python
Framework	:	Flask

PROJECT DESCRIPTION

Project Description:

In recent years, the prevailing data mining concerns people with credit card fraud detection model based on data mining. Since our problem is approached as a classification problem, classical data mining algorithms are not directly applicable. This project is to propose a credit card fraud detection system using supervised learning algorithm. supervised algorithms are evolutionary algorithms which aim at obtaining better solutions as time progresses. Credit card is the most popular mode of payment. As the number of credit card users is rising world-wide, the identity theft is increased, and frauds are also increasing. In the virtual card purchase, only the card information is required such as card number, expiration date, secure code, etc. Such purchases are normally done on the Internet or over telephone. To commit fraud in these types of purchases, a person simply needs to know the card details. The mode of payment for online purchase is mostly done by credit card. The details of credit card should be kept private. To secure credit card privacy, the details should not be leaked. Different ways to steal credit card details are phishing websites, steal/lost credit cards, counterfeit credit cards, theft of card details, intercepted cards etc. For security purpose, the above things should be avoided. In online fraud, the transaction is made remotely and only the card's details are needed. A manual signature, a PIN or a card imprint are not required at the purchase time. In most of the cases the genuine cardholder is not aware that someone else has seen or stolen his/her card information. The simple way to detect this type of fraud is to analyze the spending patterns on every card and to figure out any variation to the "usual" spending patterns. Fraud detection by analyzing the existing data purchase of cardholder is the best way to reduce the rate of successful credit card frauds. As the data sets are not available and also the results are not disclosed to the public. The fraud cases should be detected from the available data sets known as the logged data and user behavior. At present, fraud detection has been implemented by a number of methods such as data mining, statistics, and artificial intelligence. Objective – The objective of this paper is to detect fraudulent credit card transactions over non-fraudulent transactions and to use machine learning algorithms to predict fraud efficiently and accurately. There are different types of credit card fraud based on the nature of fraudulent activities such as card getting stolen, obtaining cards using false information, individuals using credit cards while being unable to pay debts, bank employees stealing card details to use it remotely, individual using skimming devices to hack credit card details, etc.

The modules developed in this application are:

- Data collection
- Data Preprocessing
- Predict Algorithm

Module description:

DATA COLLECTION:

The Dataset contains two-days credit card transactions made in September 2013 by European cardholders. The dataset is highly unbalanced with a low percentage of fraudulent transactions within several records of normal transactions. The positive class (frauds) account for 0.172% (492 frauds out of 284,807 transactions) of all transactions. Features V1, V2, ... V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. Feature 'Class' is the target variable with value 1 in case of fraud and 0 otherwise.

DATA PREPROCESSING:

Pre-processing is the process of important and common steps as follows:

1. Formatting: It is the process of putting the data in a legitimate way that it would be suitable to work with. Format of the data files should be formatted according to the need. Most recommended format is .csv files.

2. Cleaning: Data cleaning is a very important procedure in the path of data science as it constitutes the major part of the work. It includes removing missing data and complexity with naming category and so on. There may be data instances that are incomplete and do not carry the data you believe you need to address the problem. For most of the data scientists, Data Cleaning continues of 80% of work.

3. Sampling: The dataset is unbalanced. Training unbalanced dataset with learning algorithms may lead to misclassification of minority class. Therefore, to compensate for the unbalancedness, we will use ADASYN oversampling method as implemented in imbalancedlearn package. ADASYN (ADaptive SYNthetic) is an oversampling technique that adaptively generates minority data samples according to their distributions using K nearest neighbor.

4. Scaling: It is a step of data Pre Processing which is applied to independent variables or features of data. It basically helps to normalise the data within a particular range. Sometimes, it also helps in speeding up the calculations in an algorithm. It scales the data to have zero mean and unit variance. Standard Scaler is used that standardises the data.

Data Visualisation is the method of representing the data in a graphical and pictorial way, data scientists depict a story by the results they derive from analysing and visualising the data. The best tool used is

Tableau which has many features to play around with data and fetch wonderful results. Graph (Count Plot) is drawn to know the number of frauds and legal transactions. Histograms are used to know the distribution of variables.

PREDICT ALGORITHM :

we will transform the data. By getting rid of missing data and removing some columns. First we will create a list of column names that we want to keep or retain.

Next we drop or remove all columns except for the columns that we want to retain.

Finally we drop or remove the rows that have missing values from the data set.

Split into training and evaluation sets

The principal component analysis is the technique that is used, especially for the reduction of the dimension of the given dataset. The principal component analysis is one of the most efficient and an accurate method for reducing the dimensions of data, and it provides the desired results .

This method reduces the aspects of the given dataset into a desired number of attributes called principal components.

This method takes all the input as the dataset, which is having a high number of attributes so as the dimension of the dataset is very high. This method reduces the size of the dataset by taking the datapoints on the same axis. The data points are shifted on a single axis, and the principal components are carried out.

Accuracy on test set:

We got a accuracy of 99.1% on test set.

Saving the Trained Model:

Once you're confident enough to take your trained and tested model into the production-ready environment, the first step is to save it into a `.h5` or `.pkl` file using a library like `pickle`.

Make sure you have `pickle` installed in your environment.

Next, let's import the module and dump the model into `.pkl` file

INTRODUCTION TO HTML5, CSS3, BOOTSTRAP

HTML5

Introduction:

HTML stands for Hypertext Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within the tag which defines the structure of web pages. HTML 5 is the fifth and current version of HTML. It has improved the markup available for documents and has introduced application programming interfaces (API) and Document Object Model (DOM).

Features:

- ❖ It has introduced new multimedia features which support audio and video controls by using `<audio>` and `<video>` tags.
- ❖ There are new graphics elements including vector graphics and tags.
- ❖ Enrich semantic content by including `<header>` `<footer>`, `<article>`, `<section>` and `<figure>` are added.
- ❖ Drag and Drop- The user can grab an object and drag it further dropping it to a new location.
- ❖ Geo-location services- It helps to locate the geographical location of a client.
- ❖ Web storage facility which provides web application methods to store data on a web browser.
- ❖ Uses SQL database to store data offline.
- ❖ Allows drawing various shapes like triangles, rectangles, circles, etc.
- ❖ Capable of handling incorrect syntax.
- ❖ Easy DOCTYPE declaration i.e. `<!Doctype html>`
- ❖ Easy character encoding i.e. `<meta charset="UTF-8">`

Advantages:

- All browsers are supported.
- More device friendly.
- Easy to use and implement.
- HTML 5 in integration with CSS, JavaScript, etc. can help build beautiful websites.

CSS3

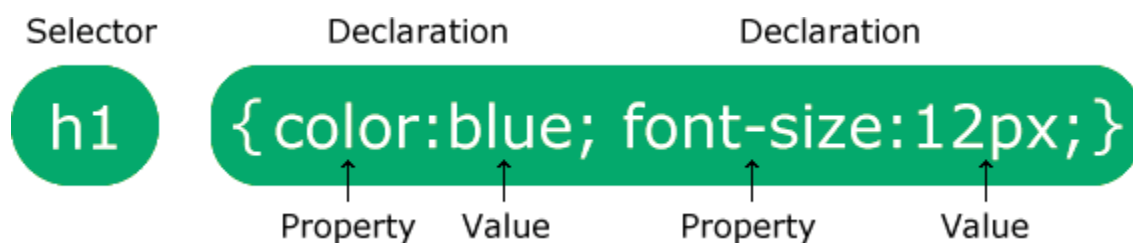
Introduction:

CSS3 stands for Cascading Style Sheet level 3, which is the advanced version of CSS. It is used for structuring, styling, and formatting web pages. Several new features have been added to CSS3 and it is supported by all modern web browsers. The most important feature of CSS3 is the splitting of CSS standards into separate modules that are simpler to learn and use.

New features of CSS3:

1. **Combinatory:** CSS3 has a new General sibling combinatory which matches up with sibling elements via the tilde (~) combinatory.
2. **CSS Selectors:** CSS3 selectors are much more advanced in comparison to simple selectors offered by CSS, and are termed as sequences-to-use use and simple selectors.
3. **Pseudo-elements:** Plenty of new pseudo-elements have been added to CSS3 to give easy styling in depth. Even a new convention of double colons: is also added.
4. **Border Style:** The latest CSS3 also has new border styling features like *border radius*, *image-slice*, *image-source*, and values for “width stretch”, etc. **background style properties:** New features like *background clip*, *size*, *style*, and *origin* properties have been added to CSS3.

CSS SYNTAX:



The selector points to the HTML element you want to style.

The declaration block contains one or more declarations separated by semicolons. Each

declaration includes a CSS property name and a value, separated by a colon.

Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces.

Bootstrap

Introduction:

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains HTML, CSS, and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

As of April 2022, Bootstrap is the eleventh most starred project on GitHub, with over 156,000 stars, behind freeCodeCamp, Vue.js framework, React library, TensorFlow a, and others.

HISTORY OF BOOTSTRAP:

Bootstrap was developed by Mark Otto and Jacob Thornton at Twitter. It was released as an open-source product in August 2011 on GitHub. In June 2014 Bootstrap was the No.1 project on GitHub.

WHY USE BOOTSTRAP:

Following are the main advantage of Bootstrap:

- It is very easy to use. Anybody having basic knowledge of HTML and CSS can use Bootstrap.
- It facilitates users to develop a responsive website.
- It is compatible with most browsers like Chrome, Firefox, Internet Explorer, Safari, Opera, etc.

WHAT BOOTSTRAP PACKAGE CONTAINS:

Scaffolding: Bootstrap provides a basic structure with Grid System, link styles, and background.

CSS: Bootstrap comes with the feature of global CSS settings, fundamental HTML elements styled, and an advanced grid system.

Components: Bootstrap contains a lot of reusable components built to provide iconography, dropdowns, navigation, alerts, pop-overs, and much more.

JavaScript Plugins: Bootstrap also contains a lot of custom jQuery plugins. You can easily include them all, or one by one.

Customize: Bootstrap components are customizable and you can customize Bootstrap's components, fewer variables, and jQuery plugins to get your style.

INTRODUCTION TO PYTHON, FLASK

PYTHON

Introduction:

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** – You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** – Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** – Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python Features

Python's features include –

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly defined syntax.
- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** – Python's source code is fairly easy-to-maintain.

- **A broad standard library** – Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** – Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** – You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** – Python provides interfaces to all major commercial databases.
- **GUI Programming** – Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** – Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below –

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

Getting Python :

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python <https://www.python.org>.

Windows Installation :

Here are the steps to install Python on Windows machine.

- Open a Web browser and go to <https://www.python.org/downloads/>.
- Follow the link for the Windows installer python-XYZ.msifile where XYZ is the version you need to install.
- To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.
- Run the downloaded file. This brings up the Python install wizard, which is really easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are some definite differences between the languages.

First Python Program

Let us execute programs in different modes of programming.

Interactive Mode Programming

Invoking the interpreter without passing a script file as a parameter brings up the following prompt –

```
$ python

Python2.4.3(#1,Nov112010,13:34:43)

[GCC 4.1.220080704(RedHat4.1.2-48)] on linux2

Type "help", "copyright", "credits" or "license" for more information.

>>>
```

Type the following text at the Python prompt and press the Enter –


```
>>>print"Hello, Python!"
```

If you are running new version of Python, then you would need to use print statement with parenthesis as in **print ("Hello, Python!")**;. However in Python version 2.4.3, this produces the following result —

```
Hello, Python!
```

Script Mode Programming

Invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension **.py**. Type the following source code in a test.py file —

```
print"Hello, Python!"
```

We assume that you have Python interpreter set in PATH variable. Now, try to run this program as follows —

```
$ python test.py
```

This produces the following result —

```
Hello, Python!
```

FLASK

Introduction:

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

Http protocol is the foundation of data communication in world wide web. Different methods of data retrieval from specified URL are defined in this protocol.

The following table summarizes different http methods –

Sr.No	Methods & Description
1	GET Sends data in unencrypted form to the server. Most common method.
2	HEAD Same as GET, but without response body
3	POST Used to send HTML form data to server. Data received by POST method is not cached by server.
4	PUT Replaces all current representations of the target resource with the uploaded content.
5	DELETE Removes all current representations of the target resource given by a URL

By default, the Flask route responds to the **GET** requests. However, this preference can be altered by providing methods argument to **route()** decorator.

In order to demonstrate the use of **POST** method in URL routing, first let us create an HTML form and use the **POST** method to send form data to a URL.

Save the following script as login.html

```
<html>

<body>

<formaction="http://localhost:5000/login"method="post">

<p>Enter Name:</p>

<p><inputtype="text"name="nm"/></p>

<p><inputtype="submit"value="submit"/></p>

</form>

</body>

</html>
```

Now enter the following script in Python shell.

```
from flask import Flask, redirect, url_for, request

app=Flask(__name__)

@app.route('/success/<name>')

def success(name):

return'welcome %s'% name

@app.route('/login',methods=['POST','GET'])

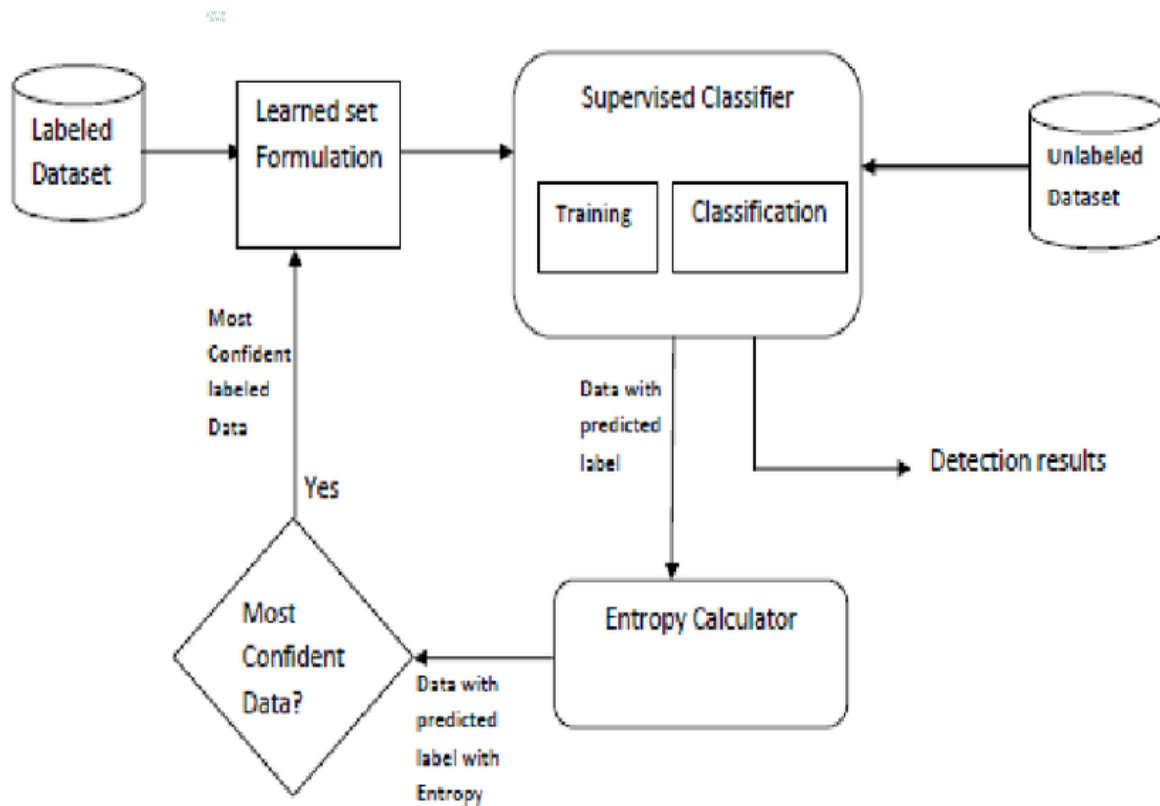
def login():
```

```
if request.method == 'POST':  
  
    user = request.form['nm']  
  
    return redirect(url_for('success', name = user))  
  
else:  
  
    user = request.args.get('nm')  
  
    return redirect(url_for('success', name = user))  
  
if __name__ == '__main__':  
  
    app.run(debug = True)
```

After the development server starts running, open **login.html** in the browser, enter name in the text field and click **Submit**.

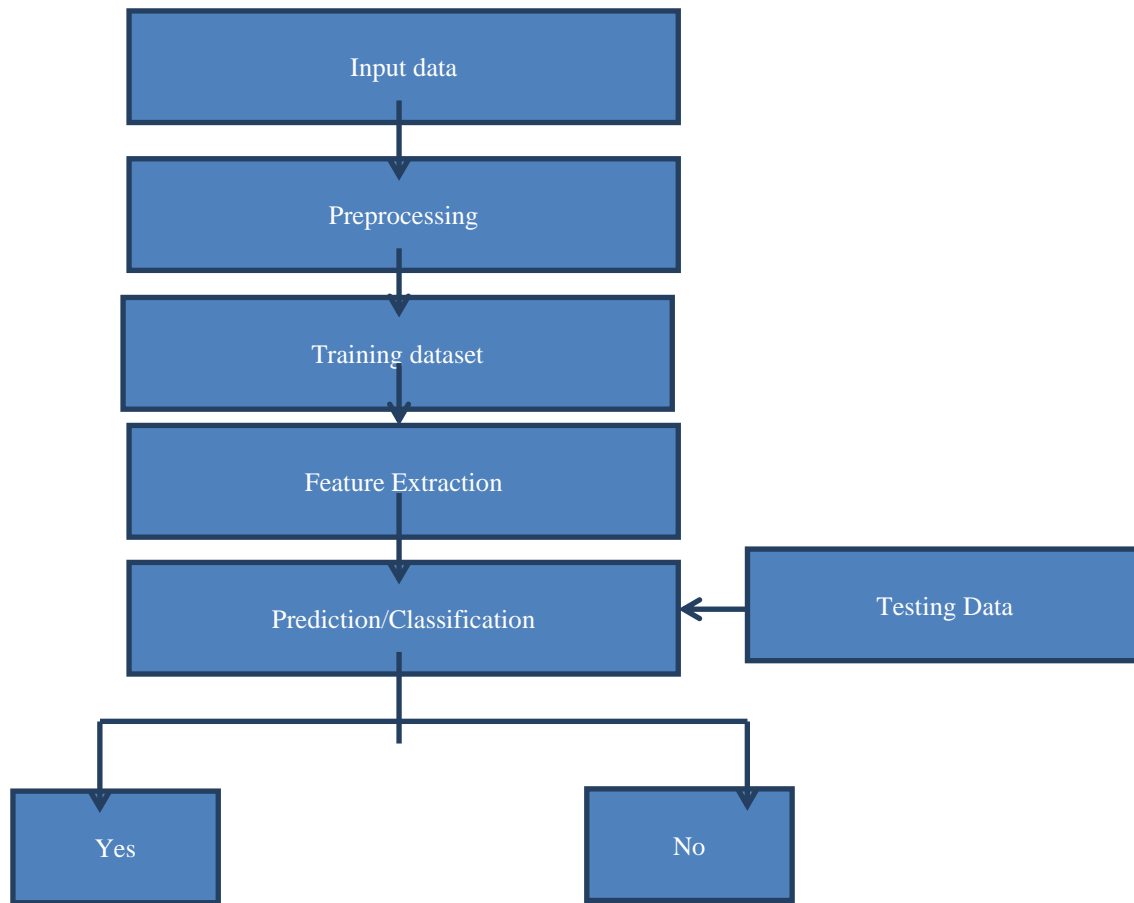
SOFTWARE DESIGN

FLOWCHART TO THE SYSTEM



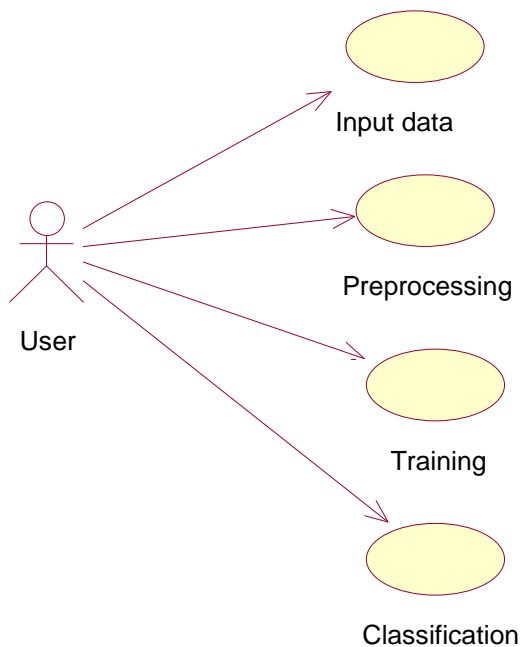
DATA FLOW DIAGRAM :

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.



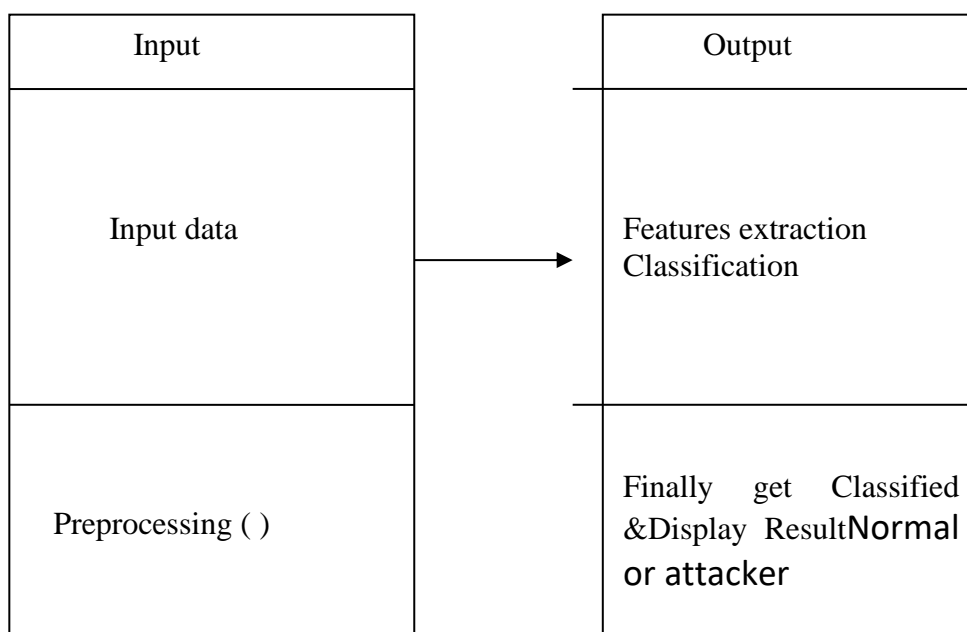
USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



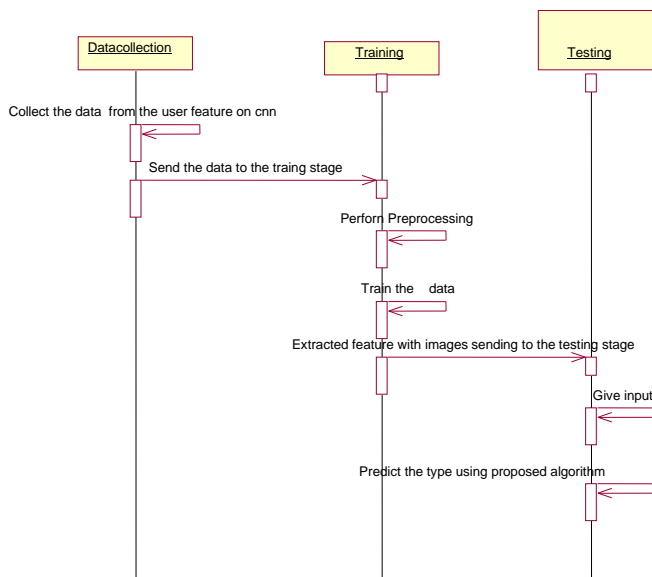
CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



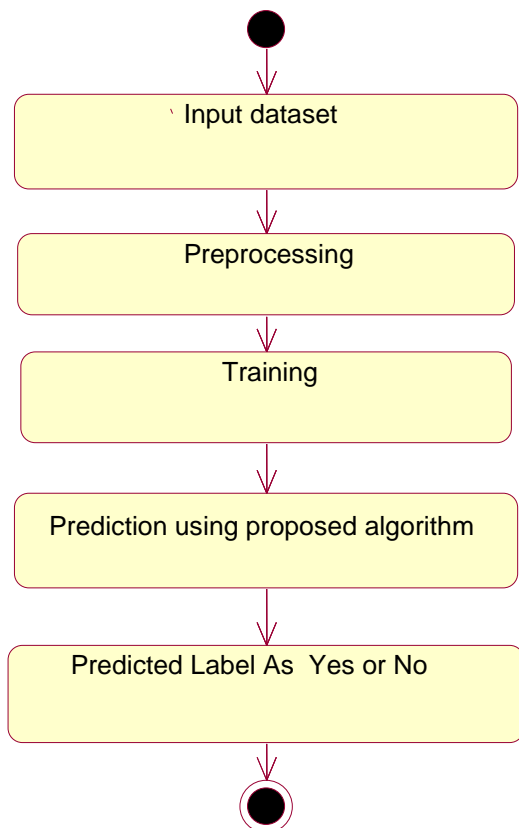
SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



SOFTWARE IMPLEMENTATION

SOFTWARE IMPLEMENTATION

CREDIT CARD FRAUD DETECTION :

This idea is difficult to implement in real life because it requires the cooperation from banks, which aren't willing to share information due to their market competition, and also due to legal reasons and protection of data of their users. Therefore, we looked up some reference papers which followed similar approaches and gathered results. As stated in one of these reference papers: "This technique was applied to a full application data set supplied by a German bank in 2006. For banking confidentiality reasons, only a summary of the results obtained is presented below. After applying this technique, the level 1 list encompasses a few cases but with a high probability of being fraudsters. All individuals mentioned in this list had their cards closed to avoid any risk due to their high-risk profile.

The condition is more complex for the other list. The level 2 list is still restricted adequately to be checked on a case by case basis. Credit and collection officers considered that half of the cases in this list could be considered as suspicious fraudulent behaviour. For the last list and the largest, the work is equitably heavy. Less than a third of them are suspicious.

In order to maximize the time efficiency and the overhead charges, a possibility is to include a new element in the query; this element can be the five first digits of the phone numbers, the email address, and the password, for instance, those new queries can be applied to the level 2 list and level 3 list."

SOFTWARE TESTING

SOFTWARE TESTING :

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner.

There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS :

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.

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 satisfactory, 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.

FUNCTIONAL TEST :

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.

SYSTEM TEST :

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.

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.

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. 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.

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.

TEST STRATEGY AND APPROACH

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

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.

Test Results: All the test cases mentioned above passed successfully. No defects encountered. **6.3**

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.

FUTURE ENHANCEMENT

FUTURE ENHANCEMENT

While we couldn't reach our goal of 100% accuracy in fraud detection, we did end up creating a system that can, with enough time and data, get very close to that goal. As with any such project, there is some room for improvement here.

The very nature of this project allows for multiple algorithms to be integrated together as modules and their results can be combined to increase the accuracy of the final result.

This model can further be improved with the addition of more algorithms into it. However, the output of these algorithms needs to be in the same format as the others. Once that condition is satisfied, the modules are easy to add as done in the code.

This provides a great degree of modularity and versatility to the project. More room for improvement can be found in the dataset. As demonstrated before, the precision of the algorithms increases when the size of dataset is increased. Hence, more data will surely make the model more accurate in detecting frauds and reduce the number of false positives. However, this requires official support from the banks themselves.

CONCLUSION

CONCLUSION

In this project, I have explained the topic “**CREDIT CARD FRAUD DETECTION**” by providing full details on it this project also emphasizes on main ideas related to the topic.

In this paper, we studied applications of machine learning like Naïve Bayes, Logistic regression, Random forest with boosting and shows that it proves accurate in deducting fraudulent transaction and minimizing the number of false alerts.

Supervised learning algorithms are novel one in this literature in terms of application domain. If these algorithms are applied into bank credit card fraud detection system, the probability of fraud transactions can be predicted soon after credit card transactions. And a series of anti-fraud strategies can be adopted to prevent banks from great losses and reduce risks.

The objective of the study was taken differently than the typical classification problems in that we had a variable misclassification cost. Precision, recall, f1-score, support and accuracy are used to evaluate the performance for the proposed system. By comparing all the three methods, we found that random forest classifier with boosting technique is better than the logistic regression and naïve bayes methods

APPENDIX-1

App.py:

```
from flask import Flask, render_template, request
import pickle
import numpy as np
from sklearn.preprocessing import OneHotEncoder
from sklearn.decomposition import PCA
filename = 'CreditCard-prediction-rfc-model.pkl'
classifier = pickle.load(open(filename, 'rb'))
```

```
app = Flask(__name__)
```

```
@app.route('/')
def home():
    return render_template('index.html')
```

```
@app.route('/predict', methods=['POST'])
def predict():
    if request.method == 'POST':
        time = int(request.form['time'])
        cardtype = int(request.form['cardtype'])
        bankname = int(request.form['bankname'])
        cardno = request.form['cardno']
        c1 = int(cardno[0])
        c2 = int(cardno[1])
        c3 = int(cardno[2])
        c4 = int(cardno[3])
        c5 = int(cardno[5])
        c6 = int(cardno[6])
        c7 = int(cardno[7])
        c8 = int(cardno[8])
        c9 = int(cardno[10])
        c10 = int(cardno[11])
        c11 = int(cardno[12])
        c12 = int(cardno[13])
        c13 = int(cardno[15])
        c14 = int(cardno[16])
        c15 = int(cardno[17])
        c16 = int(cardno[18])
        transid = int(request.form['transid'])
        cvvno = request.form['cvvno']
        cvv1 = int(cvvno[0])
        cvv2 = int(cvvno[1])
        cvv3 = int(cvvno[2])
        month = request.form['month']
        m1 = int(month[0])
```

```
m2=int(month[1])
year = request.form['year']

y1=int(year[0])
y2=int(year[1])
amount = float(request.form['amount'])
enc = OneHotEncoder()
X = enc.fit_transform([[0, 0,
bankname,cardtype],[c1,c2,c3,c4],[c5,c6,c7,c8],[c9,c10,c11,c12],[c13,c14,c15,c16],[transid,cvv1,cvv2,cvv3],[m1,
m2,y1,y2]]).toarray()
pca = PCA(n_components=4)
X_pca = pca.fit_transform(X)
data = np.array([[time, X_pca[0][0], X_pca[0][1], X_pca[0][2], X_pca[0][3], X_pca[1][0], X_pca[1][1],
X_pca[1][2], X_pca[1][3], X_pca[2][0], X_pca[2][1], X_pca[2][2], X_pca[2][3], X_pca[3][0], X_pca[3][1], X_pca[3][2],
X_pca[3][3], X_pca[4][0], X_pca[4][1], X_pca[4][2], X_pca[4][3], X_pca[5][0], X_pca[5][1], X_pca[5][2], X_pca[5][3],
X_pca[6][0], X_pca[6][1], X_pca[6][2], X_pca[6][3], amount]])
print(data)
my_prediction = classifier.predict(data)
if cardtype>=3:
    my_prediction=1
if bankname>=3:
    my_prediction=1
if c1==0:
    my_prediction=1
print(my_prediction)

return render_template('result.html', prediction=my_prediction)
#return render_template('result.html')

if __name__ == '__main__':
    app.run(debug=True)
```

HTML :

```
<!DOCTYPE html>
<html>
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>Credit Card Fraud Predictor</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
  <link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
  <link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
  <link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>
  <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</head>
```

```

<body>
  <!-- <header id="header"></header> -->
  <div id="app">
    <div class="title">
      <div class="title-inner">
        <div class="cafe">
          <div class="cafe-inner">Credit Card </div>
        </div>
        <div class="mozart">
          <div class="mozart-inner">Fraud Detection</div>
        </div>
      </div>
    </div>

    <div class="image">
      
    </div>
  </div>
  <nav id="nav">
    <ul>
      <li><a href="#"><i class='fa fa-home' style='color:#eec617'></i>Home</a></li>
      <li><a href="#"><i class='fas fa-hands-helping' style='color:#eec617'></i>Assist</a></li>
      <li><a href="#"><i class="fa fa-cogs" style='color:#eec617'></i>Support</a></li>
    </ul>
  </nav>
  <div class="box1">
    <h1> <i class="fas fa-shield-alt" style='font-size:88px;color:rgb(172, 78, 235)'></i> Your security is our top priority</h1>
    <p id="para2">We treat your information as we'd treat our own — with privacy and care. We do not make unsolicited phone calls and will never call you to offer services over the phone.</p>
    </div> <br> <br> <br>
    <h2>CreditCard Fruads Detecteds : </h2>
    <p id="para">Unauthorized card operations hit an astonishing amount of 16.7 million victims in 2017. Additionally, as reported by the Federal Trade Commission (FTC), the number of credit card fraud claims in 2017 was 40% higher than the previous year's number. There were around 13,000 reported cases in California and 8,000 in Florida, which are the largest states per capita for such type of crime. The amount of money at stake will exceed approximately $30 billion by 2020. Here are some credit card fraud statistics:</p>
    <br> <br>

    
    <br> <br>
    <h2>How Does Credit Card Fraud Happen?</h2>
    <p id="para">Credit card fraud is usually caused either by card owner's negligence with his data or by a breach in a website's security. Here are some examples: </p>
    <ul class="points">
      <li>A consumer reveals his credit card number to unfamiliar individuals.</li>

```


- A card is lost or stolen and someone else uses it.
- Mail is stolen from the intended recipient and used by criminals.
- Business employees copy cards or card numbers of its owner.
- Making a counterfeit credit card.

` </BR>

<p id="para"> When your card is lost or stolen, an unauthorized charge can happen; in other words, the person who finds it uses it for a purchase. Criminals can also forge your name and use the card or order some goods through a mobile phone or computer. Also, there is the problem of using a counterfeit credit card – a fake card that has the real account information that was stolen from holders. That is especially dangerous because the victims have their real cards, but do not know that someone has copied their card. Such fraudulent cards look quite legitimate and have the logos and encoded magnetic strips of the original one. Fraudulent credit cards are usually destroyed by the criminals after several successful payments, just before a victim realizes the problem and reports it. </p>

<hr>

<div class="gonext">

<button class="button-85" role="button" onclick="opengo()" >Let's Go to Predict</button>

</div>

<div class="login" id="predictform">

<h1>Prediction</h1>

<form action="{{ url_for('predict')}}" method="post">

<input class="form-input" type="text" name="time" placeholder="Enter the Time eg. 0">

<input class="form-input" type="text" name="bankname" placeholder="Enter the Bank

Name eg. 1.SBI, 2.HDFC, 3.IOB">

<input class="form-input" type="text" name="cardtype" placeholder="Enter the Card Type eg. 1.Platinum, 2.Gold, 3.Silver">

<input class="form-input" type="text" name="cardno" placeholder="Enter the Card No eg. XXXX XXXX XXXX XXXX">

<input class="form-input" type="text" name="transid" placeholder="Transaction No eg. 20">

<input class="form-input" type="text" name="cvvno" placeholder="Enter the Card CVV No eg. XXX">

<input class="form-input" type="text" name="month" placeholder="Enter the Card Expiry Month eg. 01">

<input class="form-input" type="text" name="year" placeholder="Enter the Card Expiry Year eg. 22">

<input class="form-input" type="text" name="amount" placeholder="Enter the Amount of Transaction eg.234">

<button type="submit" class="btn btn-primary btn-block btn-large">Predict</button>

</form>

{{ prediction_text }}

</div>

<footer></footer>

```

<script>
  function opengo() {
    document.getElementById("predictform").style.display="block";
  }
  const nav = document.querySelector('#nav');
  let navTop = nav.offsetTop;

function fixedNav() {
  if (window.scrollY >= navTop) {
    nav.classList.add('fixed');
  } else {
    nav.classList.remove('fixed');
  }
}

window.addEventListener('scroll', fixedNav);
</script>
</body>
</html>

```

Result.html :

```

<!DOCTYPE html>

<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Credit Card Predictor Result</title>
    <style>
      @import url(https://fonts.googleapis.com/css?family=Open+Sans);
      .btn {
        display: inline-block;
        *display: inline;
        *zoom: 1;
        padding: 4px 10px 4px;
        margin-bottom: 0;
        font-size: 13px;
        line-height: 18px;
        color: #333333;
        text-align: center;
        text-shadow: 0 1px 1px rgba(255, 255, 255, 0.75);
        vertical-align: middle;
        background-color: #f5f5f5;
        background-image: -moz-linear-gradient(top, #ffffff, #e6e6e6);
        background-image: -ms-linear-gradient(top, #ffffff, #e6e6e6);

```

```

background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#ffffff), to(#e6e6e6));
background-image: -webkit-linear-gradient(top, #ffffff, #e6e6e6);
background-image: -o-linear-gradient(top, #ffffff, #e6e6e6);
background-image: linear-gradient(top, #ffffff, #e6e6e6);
background-repeat: repeat-x;
filter: progid:dximagetransform.microsoft.gradient(startColorstr=#ffffff, endColorstr=#e6e6e6,
GradientType=0);
border-color: #e6e6e6 #e6e6e6 #e6e6e6;
border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1)
rgba(0, 0, 0, 0.25); border: 1px solid #e6e6e6;
-webkit-border-radius: 4px;
-moz-border-radius: 4px;
border-radius: 4px;
-webkit-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
-moz-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
cursor: pointer; *margin-left: .3em;
}
.btn:hover, .btn:active, .btn.active, .btn.disabled, .btn[disabled] { background-color: #e6e6e6; }
.btn-large { padding: 9px 14px; font-size: 15px; line-height: normal; -webkit-border-radius: 5px; -moz-
border-radius: 5px; border-radius: 5px; }
.btn:hover { color: #333333; text-decoration: none; background-color: #e6e6e6; background-position: 0 -
15px; -webkit-transition: background-position 0.1s linear; -moz-transition: background-position 0.1s linear; -ms-
transition: background-position 0.1s linear; -o-transition: background-position 0.1s linear; transition: background-
position 0.1s linear; }
.btn-primary, .btn-primary:hover { text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25); color: #ffffff; }
.btn-primary.active { color: rgba(255, 255, 255, 0.75); }
.btn-primary { background-color: #4a77d4; background-image: -moz-linear-gradient(top, #6eb6de,
#4a77d4); background-image: -ms-linear-gradient(top, #6eb6de, #4a77d4); background-image: -webkit-
gradient(linear, 0 0, 0 100%, from(#6eb6de), to(#4a77d4)); background-image: -webkit-linear-gradient(top,
#6eb6de, #4a77d4); background-image: -o-linear-gradient(top, #6eb6de, #4a77d4); background-image: linear-
gradient(top, #6eb6de, #4a77d4); background-repeat: repeat-x; filter:
progid:dximagetransform.microsoft.gradient(startColorstr=#6eb6de, endColorstr=#4a77d4, GradientType=0);
border: 1px solid #3762bc; text-shadow: 1px 1px 1px rgba(0,0,0,0.4); box-shadow: inset 0 1px 0 rgba(255, 255, 255,
0.2), 0 1px 2px rgba(0, 0, 0, 0.5); }
.btn-primary:hover, .btn-primary:active, .btn-primary.active, .btn-primary.disabled, .btn-primary[disabled]
{ filter: none; background-color: #4a77d4; }
.btn-block { width: 100%; display: block; }
body {
width: 100%;
height: auto;
font-family: 'Open Sans', sans-serif;
color: #fff;
font-size: 18px;
text-align: center;
letter-spacing: 1.2px;
background-image: url("../static/wallx.png");

```

```

}

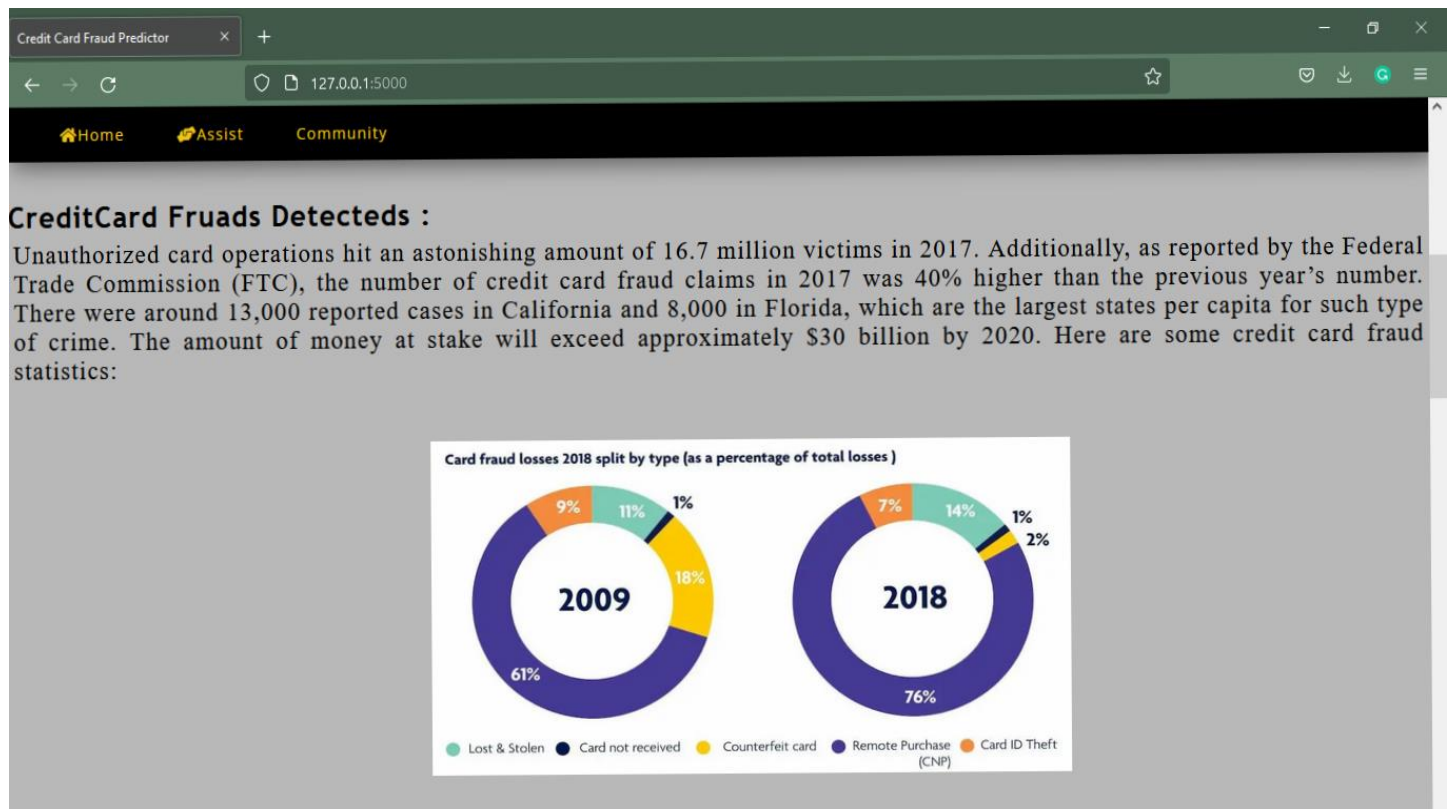
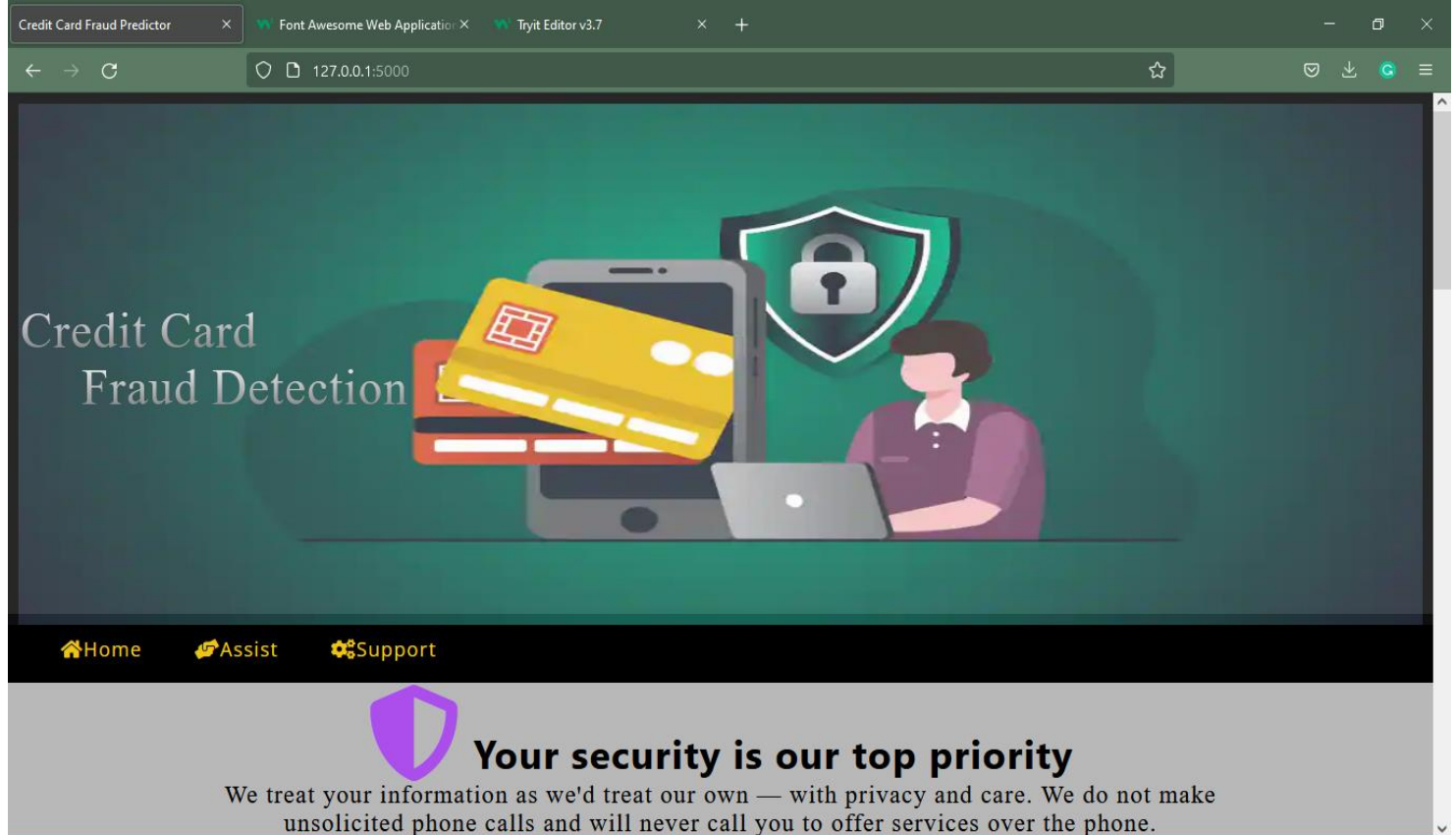
.results{
margin-top:150px;
}
</style>
</head>
<body>

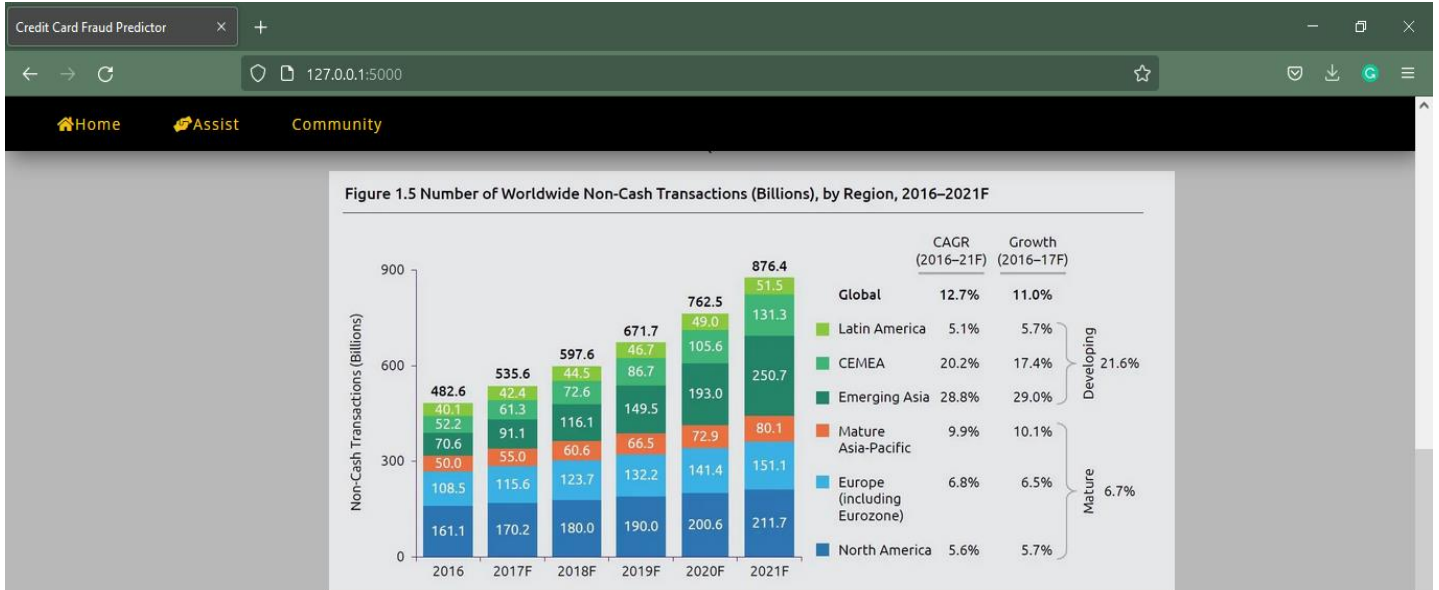
    <!-- Result -->
    <div class="results">
        {% if prediction==1 %}
            <h1>Prediction: <span class='danger'>Opps! You have Fraud Transaction in Credit
Card.</span></h1>
            <h1>Recommendation:      <span      class='danger'>Verify      Your      Bank
Soon.....</span></h1>
        {% elif prediction==0 %}
            <h1>Prediction: <span class='safe'>Hurrah !!! Your Transcation is Secure...No Fraud
Detected.</span></h1>
        {% endif %}
    </div>

</body>
</html>

```

APPENDIX-2





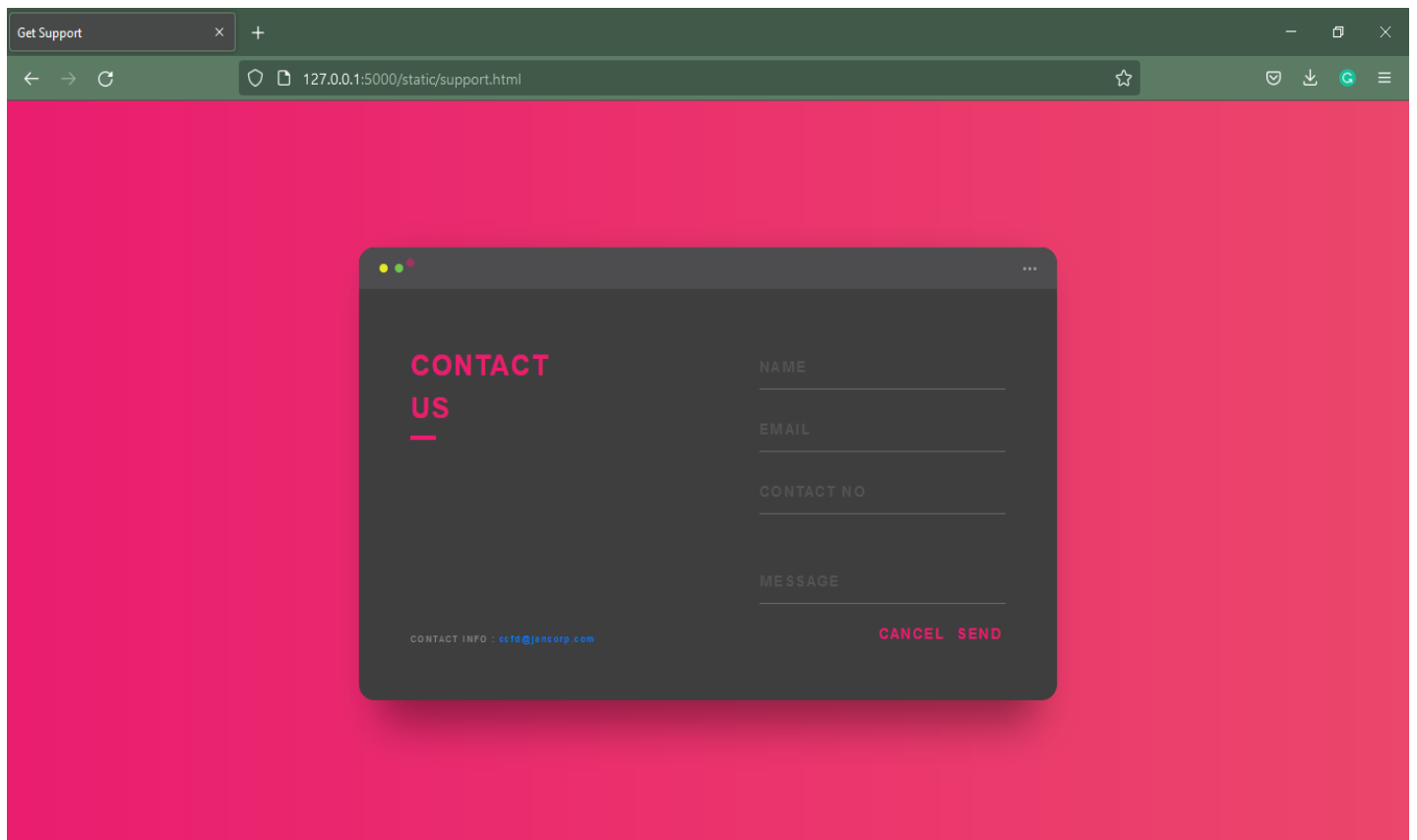
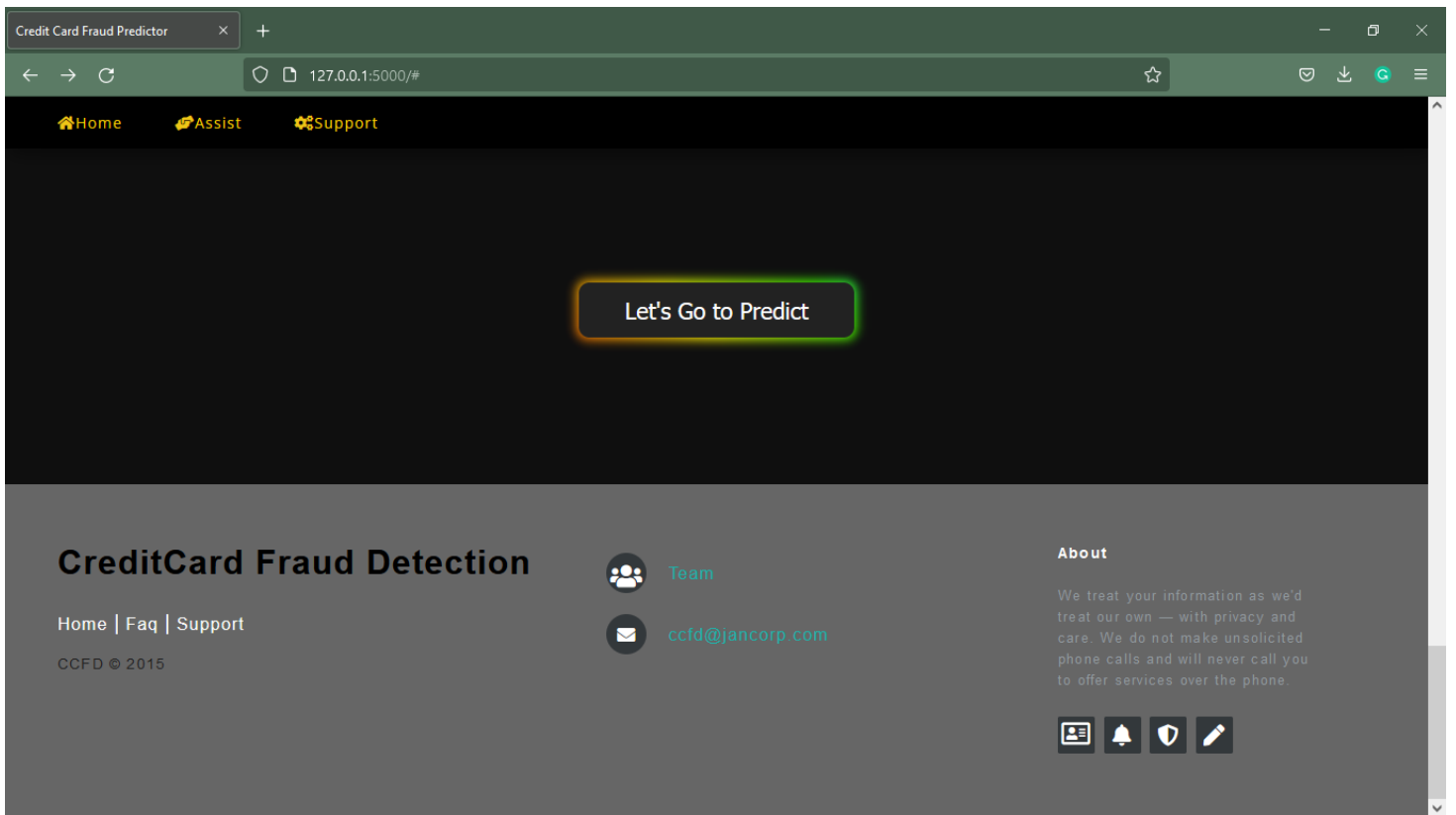
When your card is lost or stolen, an unauthorized charge can happen; in other words, the person who finds it uses it for a purchase. Criminals can also forge your name and use the card or order some goods through a mobile phone or computer. Also, there is the problem of using a counterfeit credit card – a fake card that has the real account information that was stolen from holders. That is especially dangerous because the victims have their real cards, but do not know that someone has copied their card. Such fraudulent cards look quite legitimate and have the logos and encoded magnetic strips of the original one. Fraudulent credit cards are usually destroyed by the criminals after several successful payments, just before a victim realizes the problem and reports it.

Credit Card Fraud Predictor 127.0.0.1:5000

Home Assist Community

When your card is lost or stolen, an unauthorized charge can happen; in other words, the person who finds it uses it for a purchase. Criminals can also forge your name and use the card or order some goods through a mobile phone or computer. Also, there is the problem of using a counterfeit credit card – a fake card that has the real account information that was stolen from holders. That is especially dangerous because the victims have their real cards, but do not know that someone has copied their card. Such fraudulent cards look quite legitimate and have the logos and encoded magnetic strips of the original one. Fraudulent credit cards are usually destroyed by the criminals after several successful payments, just before a victim realizes the problem and reports it.

Let's Go to Predict



Credit Card Fraud Predicti XOur Team20 Creative Footer CSSFooter Example 4இணையம் மக்கள்Ranjithm041 (RANJITH)New Tab

127.0.0.1:5000/#

HomeAssistSupport

Prediction

Enter the Time eg. 0

Enter the Bank Name eg. 1.SBI, 2.HDFC, 3.IOB

Enter the Card Type eg. 1.Platinum, 2.Gold, 3.Silver

Enter the Card No eg. XXXX XXXX XXXX XXXX

Transaction No eg. 20

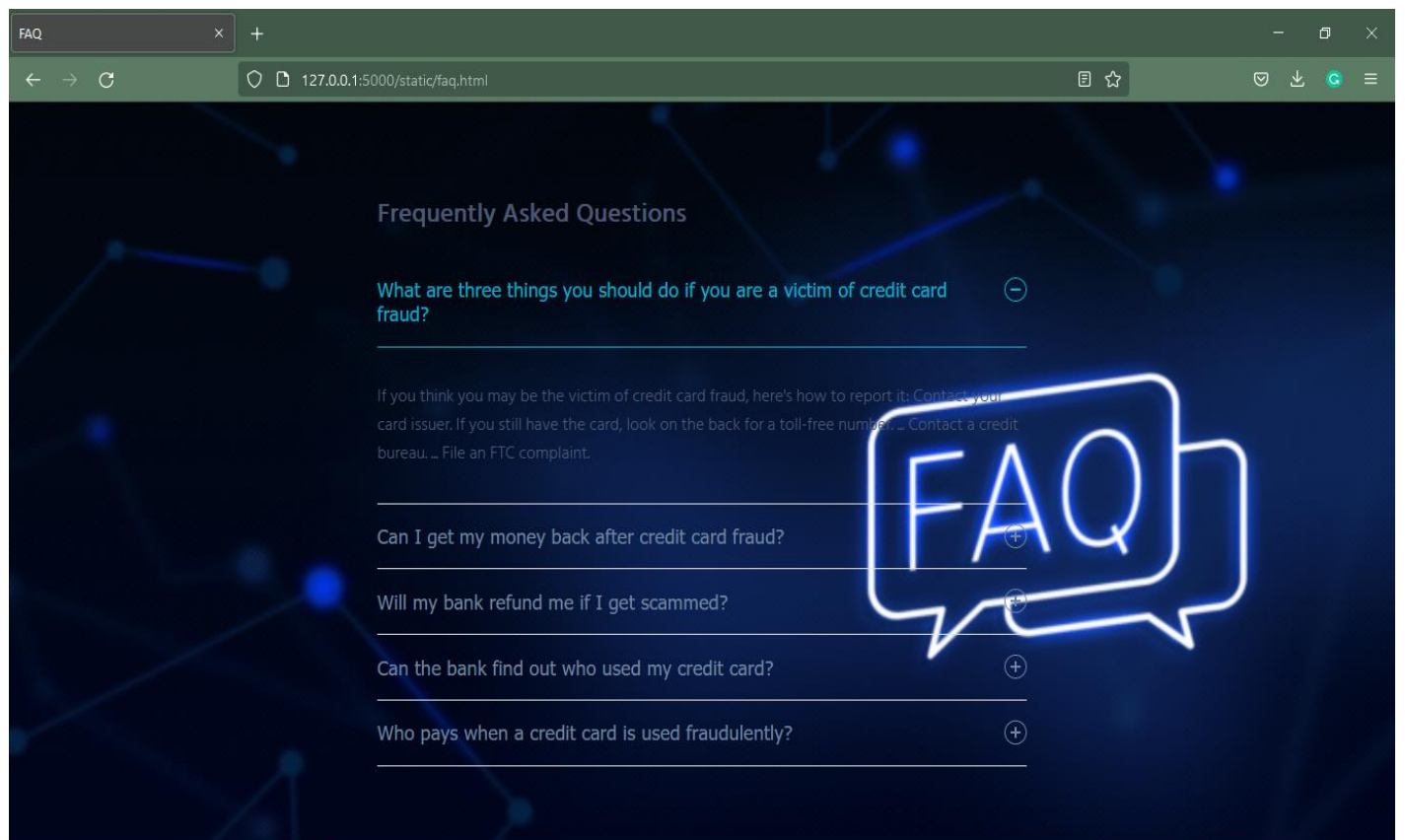
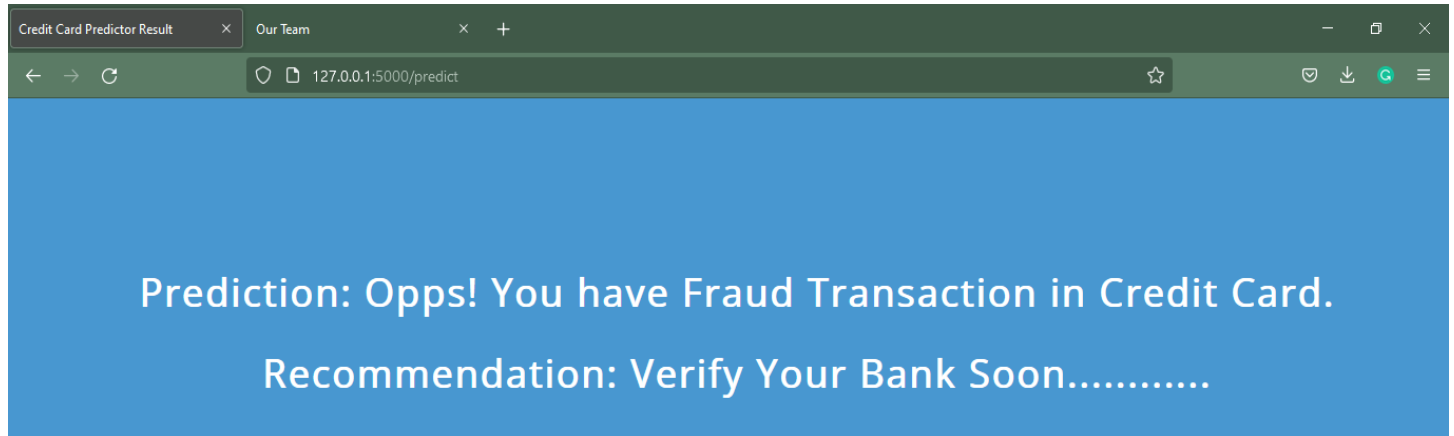
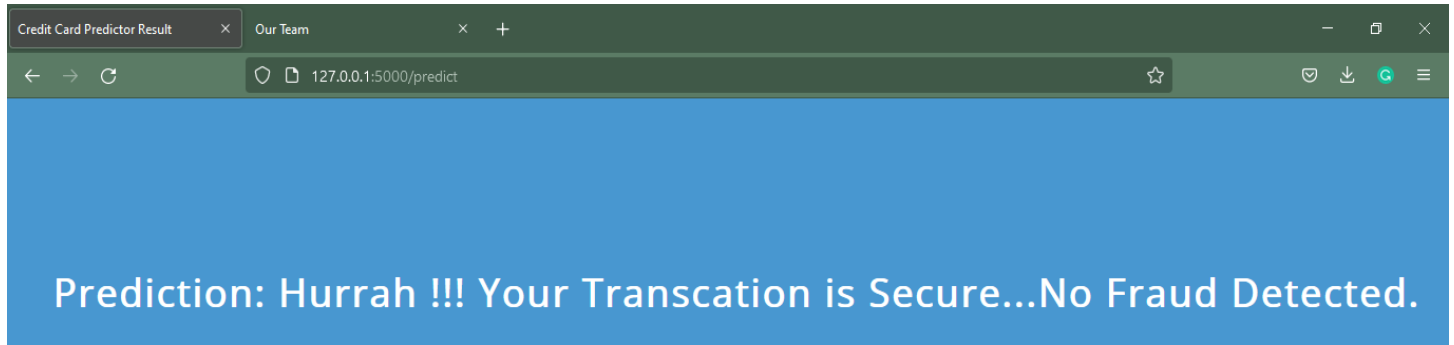
Enter the Card CVV No eg. XXX

Enter the Card Expiry Month eg. 01

Enter the Card Expiry Year eg. 22

Enter the Amount of Transaction eg.234

Predict



BIBLIOGRAPHY

BIBLIOGRAPHY

REFERENCES WEBSITES:

- www.studentproject.com
- www.codeproject.com
- www.ijert.org
- www.phpprojects.com
- www.onlinetutorials.com
- www.w3schools.com
- www.stackoverflow.com
- www.python.com
- www.codepen.com
- www.fontawesome.com

PROJECT MEMBERS

PROJECT MEMBERS:

Name:	M. SakthiKarthickraja
Reg No:	192132
Mobile No:	6383908568
Email Id:	sakthikarthikraja2004@gmail.com
Address:	315,Sakarankovil main road,rastha-627201.

Name:	K. sivaganesh
Reg No:	192138
Mobile No:	9344080136
Email Id:	ganeshsiva056@gmail.com
Address:	5/101,Pillaiyarkovil street, patta varthi

Name:	K. Thiruppathy
Reg No:	192143
Mobile No:	9385682213
Email Id:	thiruppathytitan270@gmail.com
Address:	51/f,Matha west street,Town-Tirunelveli.

Name:	G. Maniraj
Reg No:	202207
Mobile No:	9025019839
Email Id:	manirajgomathi579@gmail.com
Address:	109/D, v.v.k Street,Pettai

Name:	M.Ranjith
Reg No:	202212
Mobile No:	6385481845
Email Id:	mrnjith901@gmail.com
Address:	11/161, Gandhi puram street , Mathalamparai,Tenkasi-627814

Name:	P. Sibiraj
Reg No:	202214
Mobile No:	6383598799
Email Id:	sibiraj2704@gmail.com
Address:	4/140, soosaipar kovil street,keelanattarkulam,Tuticorin-628809