# INTRODUCTION

* 1. **Overview of SAOD(Sentiment Analysis On Demonetization) :**

The project is titled as “SENTIMENT ANALYSIS ON DEMONETIZATION” which is an application that does sentiment analysis by categorizing the given text into positive or negative. This project is mainly implemented to create a platform between the citizens and the government of that nation. As there are many projects and policies are getting implemented but, unfortunately the impacts and trends of those on the citizens are not taken into consideration. In order to overcome such complications we have created an application using the latest technologies.

Opinions of people belonging to different organizations and states are collected and this information is analyzed. Suppose if we consider demonetization which has recently occurred in India. The opinions of the common man is not considered by the authorities. Thus, by using this application opinions of people are gathered and analyzed. After analyzing these opinions, visualization is performed. How much impact has taken place is been represented using graph.

Thus, the impact analysis has done with the help of this application.

**Salient Features:**

* Fast and reliable application which reduces time complexity unlike other traditional applications available.
* Reports can be made easily and vividly such that they can be used to notify the impact occurred on the common man..
* Easy to use the application. Manages users opinions and details. Generates a graph showing the impact occurred on the citizen.

# SYSTEM ANALYSIS

**2.1 Problem Specification:**

The following part explains the existing problems in the reality and what we propose to improve the system drastically such that the outcome is simplified and sustainable.

**2.1.1 Existing System**

In the present scenario the main problem with the existing system is that the government takes the decisions without knowing the opinions of the common man. The decisions of the people is not reaching to the higher authorities. Thus, this is the main drawback in the existing system.

**2.1.2 Proposed System**

The main aim of the proposed system is to develop a system that will overcome the problems of the existing system. The system gathers the opinions of the people from different occupations & states. This information is finally represented into a graph. With the help of this graph, one can easily identify the impact occurred on the number of people.

**Key Roles:**

The project is categorized into two case studies. The first case study demonstrates gathering of data through the application. The second case study demonstrates collecting the data from any social network. The following describes about their roles, behavior with respect to the application.

# User:

* The user enters his/her details such as name, age, occupation, state .
* The user then gives his/her opinion regarding the recent change in currency through this application.
* The information entered by the user will help the analysis team to perform sentiment analysis on it.

# Analysis Team:

* The analysis team is responsible for analyzing the data which has been gathered with the help of the application.
* The analysis team copies the file into Hadoop environment, this file consists of the information given by the user and sentiment analysis on this file is performed by the analysis team using Pig.
* After analyzing the data, the analysis team performs visualization to the output generated.

# The following describes the roles, behaviors with respect to the data that has been collected from the social network. We have collected twitter dataset on demonetization.

# Admin:

* The admin gets the dataset from any social network and copies it to Hadoop environment.
* The admin then performs sentiment analysis on the dataset using Pig.
* After getting the count of number of negative and positive comments, the admin performs visualization for it.

**2.2 Feasibility Study:**

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Economic, Operational and Technical feasibilities. The following are its features:

**Economic Feasibility:**

Economic feasibility attempts to weigh the costs of developing and implementing a new system, again the benefits that would accrue from having the new system in place. The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

* The costs conduct a full system investigation.
* The cost of the hardware and software.
* The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it gives an indication of the system is economically possible for development.

**Operational Feasibility:**

Proposed project is beneficial only if it can be turned into information systems that will meet the organizations operating requirements. Simply, stated this test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to implementation? Here are questions that will help test the operation feasibility of a project.

Is there sufficient support for the project from management, from users? If the current system is well liked and used to the extent that persons will not be able to see reasons for change, there may be resistance.

Are the current business methods acceptable to the user? If they are not , Users may welcome a change that will bring about a more operational and useful systems.

Have the user been involved in the planning and development of the project? Early involvement reduces the chances of resistance to the system and increase the likelihood of successful project.

**Technical Feasibility:**

Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at this is point of time , not many detailed design of the system, making it difficult to access like performance, costs on (on account of kind of technology to be deployed etc). A number of issues have to be considered while dealing with technical analysis.

Understand the different technologies involved in the proposed system before commencing the project. We have to be very clear about what are the technologies that are to be required for the development of the new system. Find out whether the organization currently possesses the required technologies. is the current technology is available with your organization.

**2.3 Software Requirement Specification:**

Software Requirements Specification (SRS) is basically an organizational understanding of a customer (or) potential clients and the dependencies at a particular point in time prior to any actual design or developm1ent work. The sole purpose of SRS being developed is for future reference in case of any ambiguity and misunderstanding. SRS provides a detailed description of the requirements, behaviors, constraints and performance of the system.

**2.3.1 Functional Requirements**

The functional requirement of the system defines a software system or its components. A function is described as set of inputs, behavior of a system and output. Timeline should not be disturbed. Fast and efficient.

1. **Output Design:** outputs from computer systems are required primarily to communicate the results of processing to users. They are also to provide a permanent copy of the results for later consultation.
2. **Output Definition:**  outputs should be defined in terms of type, content, format, location, frequency, volume and sequence. For example decimal points need to be inserted.
3. **Output media:** Main considerations are suitability for the device to particular application, need for hard copy, response time required.
4. **Input design:** Main objective is to produce a cost effective method of input, achieve highest possible level of accuracy.
5. **Input Stages:** Input stages are data recording, data transcription, data conversion, data verification, data control, data transmission, data validation, data correction.
6. **Input Types:** Inputs are external, internal, operational, and interactive.
7. **Input media:** considers type of input, flexibility of format, speed, accuracy, verification, rejection rates, ease of correction, security.

**2.3.2 Non-Functional Requirements**

Nonfunctional requirements describe user visible aspects of the system that are not directly related to the functional behavior of the system. Nonfunctional requirements span a number of issues from user interface look and feel to response time requirements to security issues.

Non-functional requirements describe user-visible aspects of the system that are not directly related to functionality of the system.

**2.3.3 System Requirements**

The user is expected to ensure that the minimum requirement for running the product should be satisfied. The hardware and software environment in which this product was developed is specified. It is necessary to make sure that the hardware and software the consumer uses is compatible to the specification given below.

**Hardware Requirements to access the application:**

Processor - Intel, AMD

Speed - 1.1 Ghz

RAM - 4 GB (MIN)

Hard Disk - 20 GB

Key Board - Standard Windows Keyboard

Mouse - Two or Three Button Mouse

Monitor - SVGA

**Hardware Requirements for Hadoop:**

* Intel Core 2 Duo/Quad/hex/Octa or higher end 64 bit processor PC or Laptop (Minimum operating frequency of 2.5GHz)
* Hard Disk capacity of 1- 4TB.
* 64-512 GB RAM.
* 10 Gigabit Ethernet or Bonded Gigabit Ethernet.

**Software Requirements:**

Operating System - Windows, LINUX

Application Server - Apache tomcat 7.0

Front End - AngularJS, HTML, CSS

Script - JavaScript

Server side Script - PHP

Database - MYSQL

Development tools - xampp, VMWare ,Cloudera, Pig

**Apache Pig**  is a high-level platform for creating programs that run on [Apache Hadoop](https://en.wikipedia.org/wiki/Hadoop). The language for this platform is called **Pig Latin**. Pig can execute its Hadoop jobs in MapReduce, Apache Tez, or [Apache Spark](https://en.wikipedia.org/wiki/Apache_Spark). Pig Latin abstracts the programming from the [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) MapReduce idiom into a notation which makes MapReduce programming high level, similar to that of [SQL](https://en.wikipedia.org/wiki/SQL) for [RDBMSs](https://en.wikipedia.org/wiki/RDBMS). Pig Latin can be extended using [User Defined Functions](https://en.wikipedia.org/wiki/User-defined_function) (UDFs) which the user can write in Java, [Python](https://en.wikipedia.org/wiki/Python_(programming_language)), [JavaScript](https://en.wikipedia.org/wiki/JavaScript), [Ruby](https://en.wikipedia.org/wiki/Ruby_(programming_language)) or [Groovy](https://en.wikipedia.org/wiki/Groovy_(programming_language))and then call directly from the language.

**Pig Installation:**

Install, and set up Apache Pig in your system:

Prerequisites

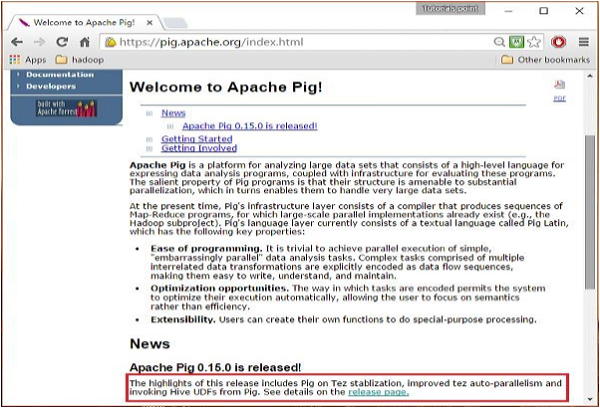
It is essential that you have Hadoop and Java installed on your system before you go for Apache Pig. Therefore, prior to installing Apache Pig, install Hadoop and JAVA.

Download Apache Pig

First of all, download the latest version of Apache Pig from the following website − https://pig.apache.org/

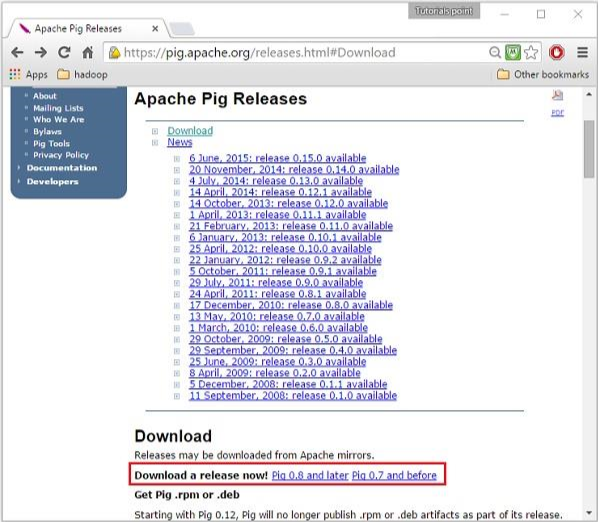
**Step 1**

Open the homepage of Apache Pig website. Under the section News, click on the link release page as shown in the following snapshot.

****

**Step 2**

On clicking the specified link, you will be redirected to the Apache Pig Releases page. On this page, under the Download section, you will have two links, namely, Pig 0.8 and later and Pig 0.7 and before. Click on the link Pig 0.8 and later, then you will be redirected to the page having a set of mirrors.



Apache Pig Releases

**Step 3**

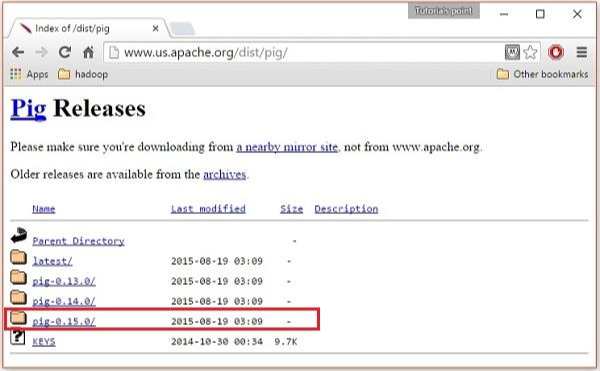
Choose and click any one of these mirrors as shown below.



Click Mirrors

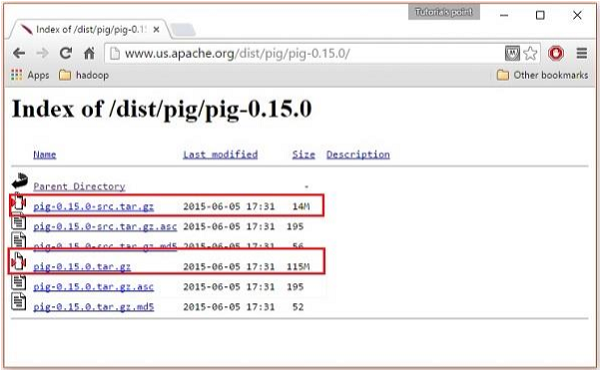
**Step 4**

These mirrors will take you to the Pig Releases page. This page contains various versions of Apache Pig. Click the latest version among them.



**Step 5**

Within these folders, you will have the source and binary files of Apache Pig in various distributions. Download the tar files of the source and binary files of Apache Pig 0.15, pig0.15.0-src.tar.gz and pig-0.15.0.tar.gz.



I*nstall Apache Pig*

After downloading the Apache Pig software, install it in your Linux environment by following the steps given below.

**Step 1**

Create a directory with the name Pig in the same directory where the installation directories of Hadoop, Java, and other software were installed.

$ mkdir Pig

**Step 2**

Extract the downloaded tar files as shown below.

$ cd Downloads/

$ tar zxvf pig-0.15.0-src.tar.gz

$ tar zxvf pig-0.15.0.tar.gz

**Step 3**

Move the content of pig-0.15.0-src.tar.gz file to the Pig directory created earlier as shown below.

$ mv pig-0.15.0-src.tar.gz/\* /usr/local/hadoop/Pig/

Configure Apache Pig

After installing Apache Pig, we have to configure it. To configure, we need to edit two files − bashrc and pig.properties.

.bashrc file

In the .bashrc file, set the following variables −

PIG\_HOME folder to the Apache Pig’s installation folder, PATH environment variable to the bin folder, and PIG\_CLASSPATH environment variable to the etc (configuration) folder of your Hadoop installations (the directory that contains the core-site.xml, hdfs-site.xml and mapred-site.xml files).

export PIG\_HOME = /home/Hadoop/Pig

export PATH = PATH:/home/Hadoop/pig/bin

export PIG\_CLASSPATH = $HADOOP\_HOME/conf

pig.properties file

In the conf folder of Pig, we have a file named pig.properties. In the pig.properties file, you can set various parameters as given below.

pig -h properties

The following properties are supported −

Logging: verbose = true|false; default is false. This property is the same as -v

--------

------

Additionally, any Hadoop property can be specified.

Verifying the Installation

Verify the installation of Apache Pig by typing the version command. If the installation is successful, you will get the version of Apache Pig as shown below.

$ pig –version

Apache Pig version 0.15.0 (r1682971)

compiled Jun 01 2015, 11:44:35

**2.4 System Architecture:**

SAOD system uses MVC (Model View Controller) architecture to develop. It uses ORM (Object Relational Mapping) technique to access or store data from and to database. MVC architecture is achieved by using the struts 2 architecture. It is designed to provide MVC architecture to the application.

Model View Controller or MVC as it is popularly called, is a software design pattern for developing web applications. A Model View Controller pattern is made up of the following three parts.

**Model**  - The lowest level of the pattern which is responsible for maintaining data.

**View**  - This is responsible for displaying all or a portion of the data to the user.

**Controller** - Software Code that controls the interactions between the Model and View.

MVC is popular as it isolates the application logic from the user interface layer and supports separation of concerns. Here the Controller receives all requests for the application and then works with the Model to prepare any data needed by the View. The View then uses the data prepared by the Controller to generate a final presentable response. The MVC abstraction can be graphically represented as follows.

Hibernate is used to adopt and use ORM technology. It maps the data to the database as objects and maps database as a object.

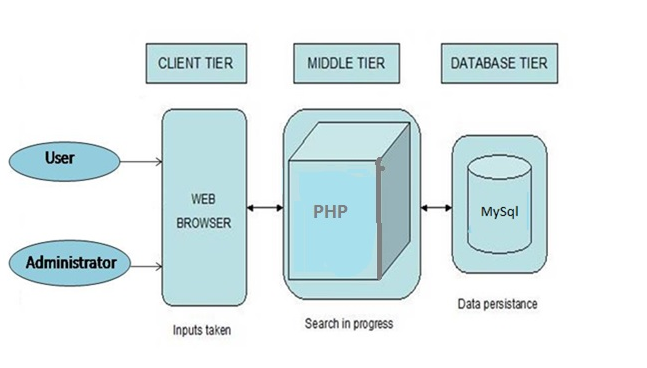


Figure 2.4 System Architecture

**2.5 Use Case Analysis:**

During requirements elicitation phase, describing the purpose of the system. The developers and the users identify the problem area and define a system that addresses the problem. Such a definition is called system specification and serves as a contract between the client and the developer. The result of this activity is a description of the following activities:

* Actors
* Use Cases
* Scenarios
* Participating Objects

**Actors involved in our project are as follows:**

1. **Administrator:** Administrator is a person who analyzes the data taken from any social site and performs visualization.
2. **User:** User gives his/her opinions regarding demonetization that has recently occurred in India with the help of application.
3. **Analysis Team:** Analysis Team performs sentiment analysis on the data gathered using the application and also performs visualization

**Identifying Use cases:**

A use case represents a complete flow of events through the system in the sense that it describes a series of related interactions that result from its initiation. Use cases are used during requirement elicitation and analysis to represent the functionality of the system. Use cases focus on the behavior of the system from the external point of view.

A use case describes a function provided by the system that yields a visible result for an actor. An actor describes an entity that interacts with the system. The identification of actors and use cases results in the definition of the boundary of the system that is in differentiating the tasks accomplished by its environment. The actors are outside the boundary of the system, where as the use cases are inside the boundary of the system.

Actors are the external entities that interact with the system. Use cases describe the behavior of the system as seen from actor’s point of view. Actor’s initiates a use case of the system functionality. When actors and use cases exchange information then they are said to be communicated.

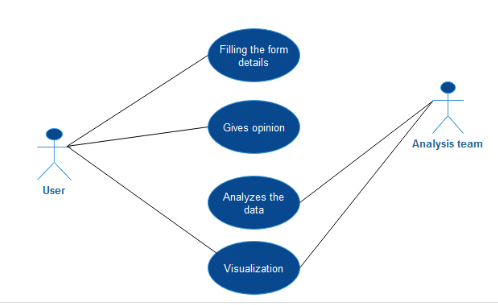
****

Figure 2.5.1 CaseStudy1 UseCase Diagram

**CaseStudy1 Use case:**

|  |  |
| --- | --- |
| Use case Name | CaseStudy1 |
| Participating Actors | User, Analysis Team |
| Entry Condition | User invokes the application |
| Flow of events | 1. Demonetization page is displayed 2. User then clicks on "give your opinion here" button to give his/her opinion regarding the act of demonetization 3. Then the user needs to enter his/her name. 4. Enters age. 5. Selects occupation. 6. Selects state. 7. Gives opinion 8. Submits his/her opinion 9. Analysis Team analyzes the data generated from the application. The analysis team copies the data to hadoop and performs sentiment analysis and also perform visualization on the output generated. 10. The user can also check out the impact occurred on the citizen by clicking on "Gathered" option in the menu bar and selecting either "From Twitter" or "From the application" |
| Exit condition | The message page is displayed. |

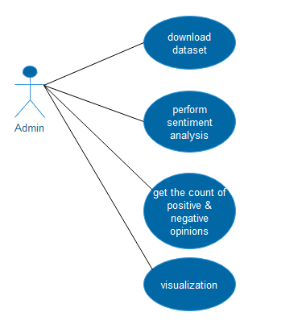
****

Figure 1.5.2 UseCase Diagram for CaseStudy 2

**CaseStudy2 Use case:**

|  |  |
| --- | --- |
| Use case Name | CaseStudy 2 |
| Participating Actors | Admin, System |
| Entry Condition | Admin needs to perform sentiment analysis on the. |
| Flow of events | 1. Admin downloads any social dataset. In this project, the admin downloads twitter dataset. 2. Then the admin copies the dataset to hadoop. 3. Performs sentiment analysis. 4. Gets the count of positive and negative tweets. 5. Performs visualization on the output generated by analyzing. |
| Exit Condition | The message page is displayed |

**Participating Objects:**

Once use cases have been consolidated, developers identify the participating objects for each use case. The participating objects correspond to the main concepts in the application domain. The identification of participating objects results in the initial Analysis model. This will only constitute a first step toward the complete Analysis model.

**2.6 Modular Description:**

Sentiment Analysis on Demonetization contains majorly two modules .

* CaseStudy 1
* CaseStudy 2

**CaseStudy 1:**

CaseStudy 1 describes the analysis of data with respect to application. Here, the user gives his/her opinion regarding demonetization that has recently took place in India using the SAOD application.

CaseStudy 1 has the following modules: User and the analysis team.

**User:**

* Filling the form details
* Gives opinion

Filling the form details:

User fills the form by providing the details given in the application

Gives opinions:

User gives his/her opinion regarding the recent changes in currency that took place in India.

**Analysis Team:**

* + This is the main team where each individual's feedback is been analyzed.
  + The user can give feedback without any restrictions
  + The analysis team then performs visualization on the output generated.

CaseStudy 2 has only one module: Admin

**Admin:**

The Admin has the following roles:

* Gets dataset from any social site
* Copies the dataset to Hadoop environment
* Performs sentiment analysis
* Gets the count of positive and negative comments.
* Performs visualization for the count generated.

# SOFTWARE DESIGN

**3.1 User Interface Design:**

This section we are dealing with how the user interface looks to the user and what are the components used to design that user screen.

HTML:

Hypertext Markup Language is used to design the user interface of the Fee Management System.

CSS:

Cascading Style Sheets are used to beautify the user interface more attractive.

AngularJS :

AngularJS is a structural framework for dynamic web applications. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application components clearly and succinctly. Its data binding and dependency injection eliminate much of the code you currently have to write. And it all happens within the browser, making it an ideal partner with any server technology.

These are the files used to create user interface

* Main.html
* opinion.html

**3.2 UML Diagrams:**

**3.2.1. Introduction:**

**UML** is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.

UML was created by **Object Management Group** (OMG) and then the **UML** 1.0 specification draft was proposed to the OMG in January 1997.

• UML stands for Unified Modeling Language.

• UML is different from the other common programming languages like C++, Java,

and COBOL etc.

• UML is a pictorial language used to make software blue prints.

So UML can be described as a general purpose visual modeling language to visualize, specify, construct and document software system. Although UML is generally used to model software systems but it is not limited within this boundary. It is also used to model non-software systems as well like process flow in a manufacturing unit etc.

UML is not a programming language but tools can be used to generate code in various languages using UML diagrams. UML has a direct relation with object oriented analysis and design.

After some standardization UML has become an OMG (Object Management Group) standard.

**3.2.2. Goals:**

A picture is worth a thousand words; this absolutely fits while discussing about UML. Object oriented concepts were introduced much earlier than UML. So at that time there were no standard methodologies to organize and consolidate the object oriented development. At that point of time UML came into picture.

There are a number of goals for developing UML but the most important is to define some general purpose modeling language, which all modelers can use, and also it needs to be made simple to understand and use.

UML diagrams are not only made for developers but also for business users, common people and anybody interested to understand the system. The system can be a software or non-software. So it must be clear that UML is not a development method rather it accompanies with processes to make a successful system.

At the conclusion the goal of UML can be defined as a simple modeling mechanism to model all possible practical systems in today’s complex environment.

**3.2.3. Conceptual Model:**

A conceptual model can be defined as a model, which is made of concepts and their relationships. A conceptual model is the first step before drawing a UML diagram. It helps to understand the entities in the real world and how they interact with each other.

**3.2.4. Object Oriented Approach:**

“**UML** can be described as the successor of object oriented analysis and design.”

An object contains both data and methods that control the data. The data represents the state of the object. A class describes an object and they also form hierarchy to model real world system. The hierarchy is represented as inheritance and the classes can also be associated in different manners as per the requirement.

The objects are the real world entities that exist around us, and the basic concepts like abstraction, encapsulation, inheritance, and polymorphism all can be represented using UML.

So UML is powerful enough to represent all the concepts exists in object oriented analysis and design. UML diagrams are representation of object-oriented concepts only.

Following are some fundamental concepts of object-oriented world:

* **Objects**: Objects represent an entity and the basic building block.
* **Class**: Class is the blue print of an object.
* **Abstraction**: Abstraction represents the behavior of an real world entity.
* **Encapsulation**: Encapsulation is the mechanism of binding the data together and hiding them from outside world.
* **Inheritance**: Inheritance is the mechanism of making new classes from existing one.
* **Polymorphism**: It defines the mechanism to exist in different forms.

**3.2.5. Building Blocks**

The building blocks of UML can be defined as:

* Things
* Relationships
* Diagrams

**Things**

**Things** are the most important building blocks of UML. Things can be classified as :

**1.** **Structural**

**2. Behavioral**

**3. Grouping**

**4. Annotational**

**Structural Things**

The **Structural things** define the static part of the model. They represent physical and conceptual elements. Following are the brief descriptions of the structural things.

## Class:

Class represents set of objects having similar responsibilities.

class

## Interface:

Interface defines a set of operations which specify the responsibility of a class.

Interface

## Collaboration:

Collaboration defines interaction between elements.

Collaboration

## Use case:

Use case represents a set of actions performed by a system for a specific goal.

Use case

## Component:

Component describes physical part of a system.

Component

## Node:

A node can be defined as a physical element that exists at run time.



# Behavioral things

# A behavioral thing consists of the dynamic parts of UML models. Following are the behavioral things:

## Interaction:

Interaction is defined as a behavior that consists of a group of messages exchanged among elements to accomplish a specific task.

Interaction

## State machine:

State machine is useful when the state of an object in its life cycle is important. It defines the sequence of states an object goes through in response to events. Events are external factors responsible for state change.

# Grouping things:

Grouping things can be defined as a mechanism to group elements of a UML model together. There is only one grouping thing available:

## Package:

Package is the only one grouping thing available for gathering structural and behavioral things.



# Annotational things:

# Annotational things can be defined as a mechanism to capture remarks, descriptions, and comments of UML model elements. Note is the only one Annotational thing available.

## Note:

## A note is used to render comments, constraints etc of an UML element.

Note

# (2) Relationship:

Relationship is another most important building block of UML. It shows how elements are associated with each other and this association describes the functionality of an application.

There are four kinds of relationships available.

## Dependency:

Dependency is a relationship between two things in which change in one element also affects the other one.

Dependency

## Association:

Association is basically a set of links that connects elements of an UML model. It also describes how many objects are taking part in that relationship.

Association

## 

## Generalization:

Generalization can be defined as a relationship which connects a specialized element with a generalized element. It basically describes inheritance relationship in the world of objects.

Generalization

## Realization:

Realization can be defined as a relationship in which two elements are connected. One element describes some responsibility which is not implemented and the other one implements them. This relationship exists in case of interfaces.

Realization

**Diagrams**

UML diagrams are the ultimate output of the entire discussion. All the elements, relationships are used to make a complete UML diagram and the diagram represents a system.

**Class Diagram**

Class diagrams are the most common diagrams used in UML. Class diagram consists of classes, interfaces, associations and collaboration. Class diagrams basically represent the object oriented view of a system, which is static in nature. Active class is used in a class diagram to represent the concurrency of the system.

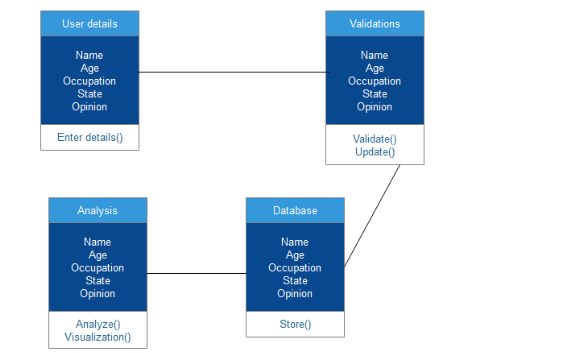


Figure 3.2 Class Diagram for CaseStudy 1

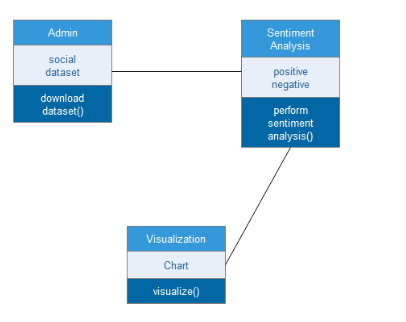
****

Figure 2.2.1 Class Diagram for CaseStudy 2

**Component Diagram**

Component diagrams represent a set of components and their relationships. These components consist of classes, interfaces or collaborations. Component diagram represent the implementation view of a system.

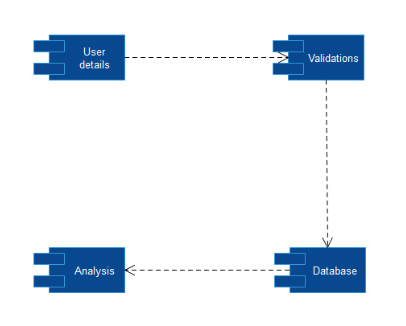


Figure 3.2.2 Component Diagram for CaseStudy 1

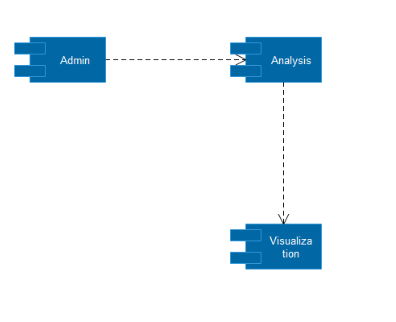


Figure 3.2.3 Component Diagram for CaseStudy 2

**Deployment Diagram**

Deployment diagrams are a set of nodes and their relationships. These nodes are physical entities where the components are deployed. Deployment diagrams are used for visualizing deployment view of a system. The deployment team generally uses this.

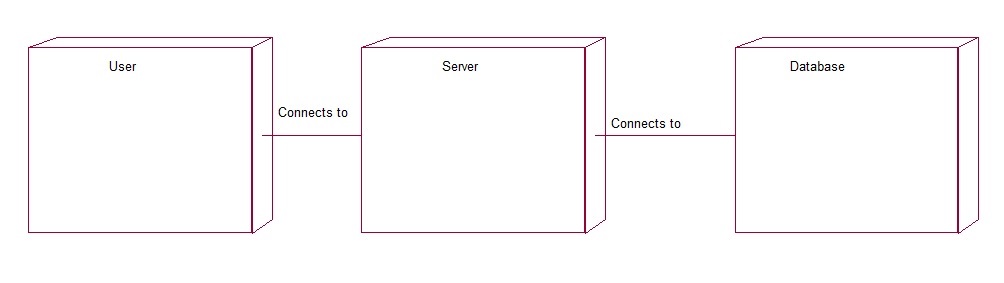


Figure 3.2.4 Deployment Diagram

**Interaction Diagrams**

Interaction diagrams are models, which show interaction among set of objects and their relationships including messages. Interaction diagram comes in two forms.

• Sequence Diagram

• Collaboration Diagram

**Sequence Diagram**

A sequence diagram is an interaction diagram. From the name it is clear that the diagram deals with some sequences, which are the sequence of messages flowing from one object to another. Interaction among the components of a system is very important from implementation and execution perspective.

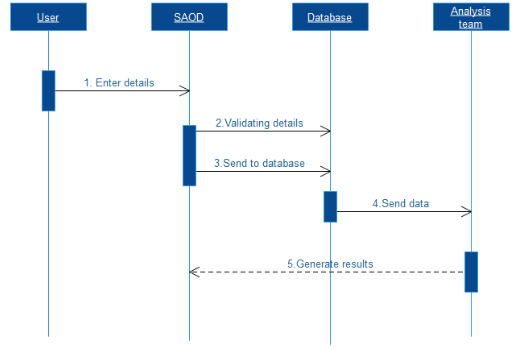
****

Figure 3.2.5 Sequence Diagram for CaseStudy 1

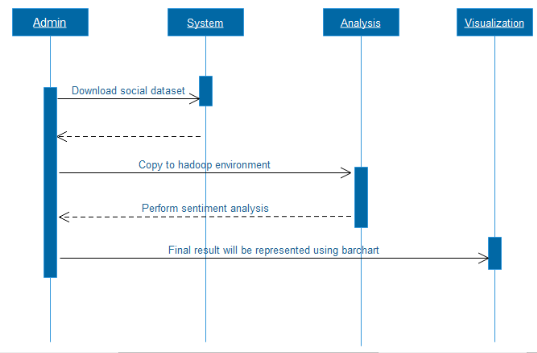
****

Figure 3.2.6 Sequence Diagram for CaseStudy 2

**Collaboration Diagram:**

Collaboration diagram is another form of interaction diagram. It represents the structural organization of a system and the messages sent/received. Structural organization consists of objects and links.

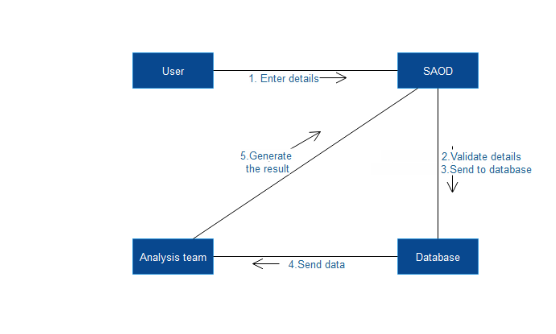
****

Figure 3.2.7 Collaboration Diagram for CaseStudy1

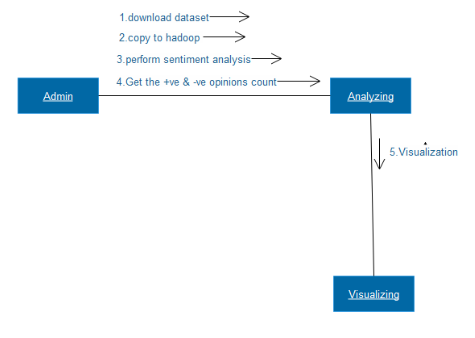


Figure 3.2.7 Collaboration Diagram for CaseStudy 2

**Activity Diagrams**

Activity diagram describes the flow of control in a system. So it consists of activities and links. The flow can be sequential, concurrent or branched. Activities are nothing but the functions of a system. Numbers of activity diagrams are prepared to capture the entire flow in a system.

**Activity Diagram for CaseStudy 1:**

Activity diagram for CaseStudy1 is illustrated below:

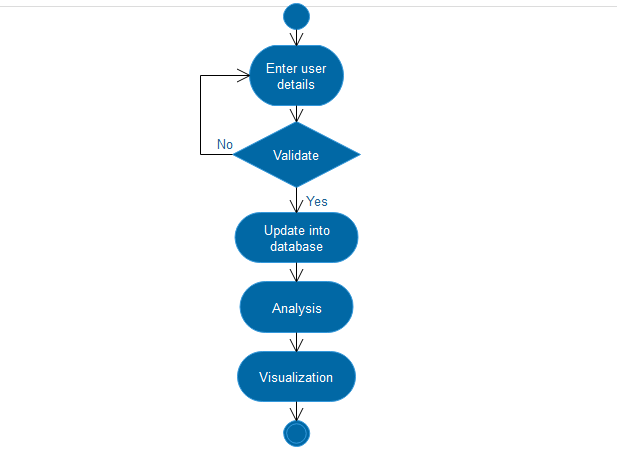


Figure 3.2.8 Activity diagram for CaseStudy 1

**Activity Diagram for CaseStudy2:**

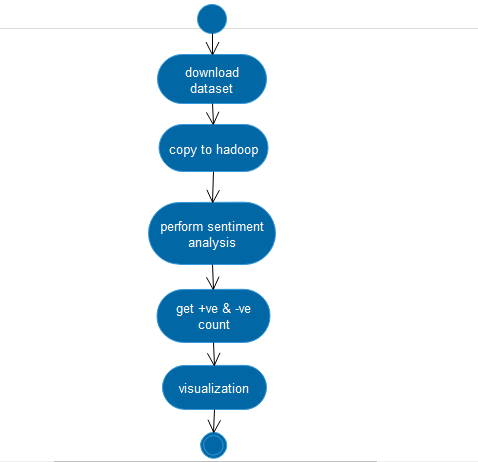
**-**

Figure 3.2.9 Activity Diagram for CaseStudy 2

**3.3 Database Design:**

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

Normalization:

Normalization is a systematic approach of decomposing tables to eliminate data redundancy and undesirable characteristics like Insertion, Updation and Deletion Anomalies. It is a two-step process that puts data into tabular form by removing duplicated data from the relation tables.

Normalization is used for mainly two purposes,

• Eliminating redundant (useless) data

• Ensuring data dependencies make sense i.e. data is logically stored

We should normalize to remove anomalies like,

• Updating Anomaly

• Insertion Anomaly

• Deletion Anomaly

Normalization rules are divided into following normal forms,

1. First Normal Form

2. Second Normal Form

3. Third Normal Form

Below table illustrates the number of columns and datatypes used in this application. There is only a single table 'fem' is used in this application.

**fem**

|  |  |  |
| --- | --- | --- |
| Name | NOT NULL | VARCHAR(30) |
| Age1 | NOT NULL | VARCHAR(30) |
| Occu | NOT NULL | VARCHAR(30) |
| State | NOT NULL | VARCHAR(30) |
| Opinion | NOT NULL | VARCHAR(200) |

**CODING**

**4.1 Sample Code:**

**main.html**

<!DOCTYPE html>

<html class="no-js" lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta name="description" content="Demonetization is a act of defining the facts..."/>

<meta property="og:title" content="Demonetization : It is necessary whenever there is a change of national currency." />

<meta property="og:type" content="website" />

<meta property="og:description" content="Demonetization

Demonetization is necessary whenever there is a change of national currency..

for knowning the opinions" />

<title>Demonetization : It is necessary whenever there is a change of national currency.</title>

<link href="https://fonts.googleapis.com/css?family=Roboto:100,300,300i,400,500,700,900" rel="stylesheet">

<link rel="stylesheet" href="https://assets.awwwards.com/css/8f2abde.css?v29" />

<link rel="stylesheet" type="text/css" media="screen" href="https://assets.awwwards.com/css/854e0f3.css?v=1" />

<style>

div.bsap a {

display: inline-block!important;

}

</style>

</head>

<body class=" ">

<div class = "wrapper ">

<nav class="nav-main" id="nav-main">

<div class="top">

<div class="header">

<div class="pull-left">

<span class="link-1 js-nav-menu">ENGLISH</span>

</div>

<div class="pull-right">

<div class="bt-close js-nav-main">CLOSE</div>

</div>

</div>

<ul class="menu active" data-menu-name="BACK" id="menu-main">

<li class="active"><a href="/" class="item">Home</a></li>

<li class="dropdown">

<span class="item">Gallery</span>

<ul>

<li><a href="photos" class="item">photos for the demonetization</a></li>

<li><a href="news" class="item">news for the demonetization</a></li>

<li><a href="videos" class="item">videos for the demonetization</a></li>

</ul>

</li>

<li class="dropdown">

<span class="item">Gathered</span>

<ul>

<li><a href="https://app.powerbi.com/view?r=eyJrIjoiNzMwNzdkNDAtZTQ1OC00NjFiLTljMzktYjcxYzhmNjNlZjRiIiwidCI6IjliYzI2NGQyLWI0YTgtNDJiMS04YWFhLWQ5ZDdhYTY1NjQ4YyIsImMiOjEwfQ%3D%3D" class="item">From Twitter</a></li>

<li><a href="https://app.powerbi.com/view?r=eyJrIjoiYzQ4OGNlOWYtYWE4Zi00MWM0LWFjMWQtM2JlZjRmYzcyMDRjIiwidCI6IjliYzI2NGQyLWI0YTgtNDJiMS04YWFhLWQ5ZDdhYTY1NjQ4YyIsImMiOjEwfQ%3D%3D" class="item">From the application</a></li>

</ul>

</li>

<li class="dropdown">

<span class="item">About demonetisation</span>

<ul>

<li><a href="history and scenario" class="item">History and indian scenario</a></li>

<li><a href="impacts of demonetisation" class="item">Impacts of demonetisation</a></li>

<li><a href="demonetisation in india" class="item">Demonetisation in india</a></li>

</ul>

</li>

<li class="dropdown">

<span class="item">About Us</span>

<i></i>

<ul>

<li><a href="/about-us/" class="item">About the application</a></li>

<li><a href="/about-us/" class="item">About the People who developed it</a></li>

</ul>

</li>

<li><a href="help" class="item">HELP</a></li>

</ul>

<ul class="menu" data-menu-name="BACK" data-menu-id="menu-main" id="menu-lang">

<li class="active">

<a href="#" class="item">

English

<span class="bt-check active">

<span class="bt-content"></span>

</span>

</a>

</li>

</ul>

</div>

<div class="box-bottom">

<div class="box-version">

<span>Activate black opinion</span>

<div class="input-check-toggle check-small ">

<input id="version-color" type="checkbox" class="js-dark">

<label for="version-color">

<div class="ball"></div>

</label>

</div>

</div>

</div>

</nav>

<header id="header">

<body bgcolor=''>

<div class="box-left">

<div class="item bt-menu js-nav-main">

<svg class="ico-svg ico-13 ico-rendering" viewbox="0 0 15 13" xmlns="http://www.w3.org/2000/svg">

<use xlink:href="/bundles/tvweb/images/sprite-icons.svg#menu-main" xmlns:xlink="http://www.w3.org/1999/xlink"></use>

</svg>

<span><b><font color="blue">MENU </font></b></span>

</div>

</div>

<div class="box-right" >

<div class="item bt-search js-search">

<svg class="ico-svg" viewbox="0 0 14 14" xmlns="http://www.w3.org/2000/svg">

<use xlink:href="/bundles/tvweb/images/sprite-icons.svg#search" xmlns:xlink="http://www.w3.org/1999/xlink"></use>

</svg>

</div>

<div class="item" id="bt-submit" onClick="location.href='opinion.html'">

<a href='opinion.html' class="button button-large">

<span>GIVE YOUR OPINION HERE!</span>

</a>

<div class="bt-gotop">

<svg class="ico-svg" viewbox="0 0 14 14" xmlns="http://www.w3.org/2000/svg">

<use xlink:href="/bundles/tvweb/images/sprite-icons.svg#gotop" xmlns:xlink="http://www.w3.org/1999/xlink"></use>

</svg>

</div>

</div>

</div>

</header>

<section id="content">

<div class="block pt-0">

<div class="inner">

<div class="box-breadcrumb">

<div class="box-left">

<b>We're blaming 'society', yet we are society.So to make it a better place,we must change ourselves first.</b>

</div>

<div class="box-right" id="post\_collected"></div>

</div>

</div>

<article itemscope="" itemtype="http://schema.org/Article">

<div class="header-post horizontal-center">

<div class="inner width-4">

<div class="row">

<h1 class="heading-large" itemprop="name"> DEMONETIZATION: It is an act of stripping a currency unit of its status as legal tender.</h1>

</div>

<div class="row info">

DEMONETIZATION

<strong><a href="opinion.html">for knowing opinions</a></strong>

about

<strong itemprop="about"><a href="opinion.html">present situation</a></strong>

- <span itemprop="publishedAt">from the indian economy</span>

</div>

</div>

</div>

<div class="entry entry-blog" itemprop="articleBody">

<div class="inner width-4 first-letter">

<p>Demonetization is the<a href="opinion.html">"monetary step"</a></p>

<blockquote>

<p>I want to express my opinions to <del>myself</del> <ins>higher authorities</ins></p>

<p>Appolizing the trend and make them to known our views.</p></blockquote>

<p>Demonetization would not only repair internal economic issues, but also tackles funding to terrorism.Instead of making our own decisions about the people.The move to demonetize Rs 500 and Rs 1,000 currency tenders by the union government of India on last Tuesday, was a laudable and historic effort to clean up the decades long corruption and black money.</p>

<p>it is for, taking advantage of features like: If unaccounted or black money isn’t obliterated, it would make the rich richer and the poor poorer. According to SBI, “ Banks received deposits worth Rs 53,000 crore since the government put out of circulation, high-value banknotes in a bid to drain illegal wealth. Nearly 14 lakh crore are held in Rs 500 and Rs 1000 notes now – nearly 86% of the total value of currencies of all denominations in circulation” [1]. Ultimately, government will possess an adequate amount of money to invest in infrastructure, education, agriculture, and all other sectors to build India economically stronger.Moreover, certain

percentage of unaccounted money stocked in the form of valuable commodities such as gold has been secured in the vaults by certain section of people.</p>

<p><b>Why there is “hue and cry” in the society over demonetization?</b><br><br>As the saying goes “no pain no gain”.In the same way short-term constraints will be associated with long-term benefits in this process, such as all of sudden endeavor to convert

currency, limitation of new tenders throughout the network for a certain period of time.</p>

<p>An Inspiration story of a taxi driver, who sacrificed his share of money for a passenger carrying only Rs 500 notes, has set an example for fellow citizens. Furthermore he said “Even if I earn a little less money, it’ll slightly difficult to manage but that’s the case with everyone. I’ll consider this as my contribution to the country’s growth” . Addressing the nation, Prime Minister of India provided assurance to the citizens, he quoted, “I want to tell the people again and again that the government will do everything to protect the honest” </p>

<p><b>How would we benefit from demonetization?</b><br><br>Above all, every honest taxpayer should hail this decision. In the present economical situation, black money has inflated prices in real estate, gold and a few other sectors, making it a challenge for a common Indian citizen to invest. However the government’s attempt to curb black money will significantly lower the prices in such sectors. According to Finance ministry’s report in 2012, “real-estate accounts for more than 50 per cent of the current black money market”. Demonetization would not only repair internal economic issues, but also tackles funding to terrorism. Counterfeit money is one of the main sources of funding for activities related to terrorism . Also, corrupted government officials and politicians who have earned in illegal ways will have no other option to put that money into usage.</p>

<h2><u>Few examples of countries that undertook demonetisation</u></h2>

<p>1. European monetary union nations demonetised their currencies in 2002 to adopt euro.<br>2. Government of Zimbabwe demonetised its currency in 2015 to protect the economy from hyperinflation<br>

other countries that have demonetised their currencies in the past are:Fiji,Singapore&Philippines<br>Fiji - 1945<br>Singapore - 1969<br>Philippines - 1985</p>

<h2><u>7 Hidden Economic Lessons For Demonetization</u></h2>

<p>1.Trackling black money in economic<br>2. Reduces money circulation<br>3.Boosts bank deposits and savings<br>4. Financial inclusion<br>5. Support government finances<br>6. Boost the GDF(in long run)<br>7.A lesson in behavioural economics

<p><br></p>

</div>

</div>

</article>

</div>

<div class="block pt-0">

<div class="inner width-4">

<div id="disqus\_thread"></div>

<script type="text/javascript">

</script>

</div>

</div>

</section>

<footer id="footer">

<div class="inner">

<div class="box-left">

<h1 class="slogan not-mobile">The secret of the change is to focus all of your energy, not on fighting the old, But on building the new.</h1>

</div>

</div>

<div class="bt-gotop"></div>

</footer>

</div>

<script src="https://code.jquery.com/jquery-2.2.3.min.js" integrity="sha256-a23g1Nt4dtEYOj7bR+vTu7+T8VP13humZFBJNIYoEJo=" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/underscore.js/1.8.3/underscore-min.js" integrity="sha256-obZACiHd7gkOk9iIL/pimWMTJ4W/pBsKu+oZnSeBIek=" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/backbone.js/1.3.3/backbone-min.js" integrity="sha256-0atoj6xVOJUoBM8Vp5PFywwLLE+aNl2svi4Q9UWZ+dQ=" crossorigin="anonymous"></script>

<script src="http://assets.awwwards.com/js/9efd07a.js?v5.38"></script>

<script type="text/javascript">

var app = app || {};

app.baseUrlLocal = '/';

</script>

<!-- Fancybox -->

<script>

$(document).ready(function() {

$(".fancybox").fancybox();

});

</script>

<style>

.blockquoteCenter{

margin:0 auto;

width:auto;

display:table;

}

.entry h2 {

font-family:"MuseoSans";

}

.entry h2{

margin-top:2em;

font-size:2.3em;

}

.entry h4{

font-family:"MuseoSans";

font-size:1em;

}

</style>

</body>

</html>

**opinion.html**

<html>

<head>

<title>DEMONETIZATION</title>

<style>

input[type=text], select {

width: 100%;

padding: 12px 20px;

margin: 8px 0;

display: inline-block;

border: 1px solid #ccc;

border-radius: 4px;

box-sizing: border-box;

}

input[type=submit] {

width: 100%;

background-color:goldenrod;

color: white;

font-weight: bold;

font-size: 21px;

padding: 14px 20px;

margin: 8px 0;

border: none;

border-radius: 4px;

cursor: pointer;

}

input[type=submit]:hover {

background-color: goldenrod;

}

div {

border-radius: 5px;

background-color: #f2f2f2;

padding: 20px;

}

.header img {

float: left;

width: 150px;

height: 120px;

background: #555;

}

.header h1 {

position: relative;

top: 18px;

left: 10px;

}

</style>

<script>

function checknameEmpty()

{

var name=document.f1.name.value;

var status= false;

if(name.length<1){

document.getElementById("nname").innerHTML= "Please enter your name";

name.focus();

status=false;

}

else{

document.getElementById("nname").innerHTML= "";

}

}

function checkageEmpty()

{

var age=document.f1.age1.value;

if (isNaN(age) || age.length>2)

{

document.getElementById("numloc").innerHTML= "Please enter correct age";

return false;

}else{

document.getElementById("numloc").innerHTML= "";

}

}

function checkoccupationEmpty()

{

var occupation=document.f1.getElementById("ocp");

if (occupation.value == "") {

document.getElementById("occ").innerHTML= "Please select your occupation";

return false;

}

return true;

}

function checkopinionEmpty()

{

var op=document.f1.opinion.value;

var status= false;

if(op.length<1){

document.getElementById("opn").innerHTML= "Please give your opinion";

name.focus();

status=false;

}

else{

document.getElementById("opn").innerHTML= "";

}

var name=document.f1.name.value;

var age1=document.f1.age1.value;

var occu=document.f1.occu.value;

var state=document.f1.state.value;

var opinion=document.f1.opinion.value;

if(name.length<1 && age1.length<1 && occu.length<1 && op.length<1

{

echo "Please fill all the fields";

}

}

</script>

</head>

<body>

<div class="header">

<img src="d.jpg">

<h1 align="center"><u><font color="goldenrod" size="36">Analysis On Demonetization</font></u></h1>

</div>

<div>

<form name="f1" method="post" action="3.php">

<label for="Name">Name</label>

<input type="text" id="name" name="name" placeholder="Your name.." onBlur="checknameEmpty()" required>

<span id="nname" style="color:red"></span><br>

<label for="age">Age</label>

<input type="text" id="age1" name="age1" placeholder="Your age.." onBlur="checkageEmpty()" required>

<span id="numloc" style="color:red"></span><br>

<label for="occupation">Occupation</label>

<select id="ocp" name="occu" onBlur="checkoccupationEmpty()" required><span id="occ"></span><br>

<option selected="" value="Default">Please select your occupation...</option>

<option value="Business">Business</option>

<option value="Employee">Employee</option>

<option value="Student">Student</option>

<option value="Housewife">Housewife</option>

<option value="Doctor">Doctor</option>

<option value="Teacher">Teacher</option>

<option value="Journalist">Journalist</option>

<option value="Lawyer">Lawyer</option>

<option value="Clerk">Clerk</option>

<option value="Other">Other</option>

</select>

<label for="state">State</label>

<select id="state" name="state" onBlur="checkstateEmpty()" required><span id="st"></span><br>

<option selected="" value="Default">Please select your State...</option>

<option value="Andhra Pradesh">Andhra Pradesh</option>

<option value="Arunachal Pradesh">Arunachal Pradesh</option>

<option value="Assam">Assam</option>

<option value="Bihar">Bihar</option>

<option value="Chhattisgarh">Chhattisgarh</option>

<option value="Goa">Goa</option>

<option value="Gujarat">Gujarat</option>

<option value="Haryana">Haryana</option>

<option value="Himachal Pradesh">Himachal Pradesh</option>

<option value="Jammu and Kashmir">Jammu and Kashmir</option>

<option value="Jharkhand">Jharkhand</option>

<option value="Karnataka">Karnataka</option>

<option value="Kerala">Kerala</option>

<option value="Madhya Pradesh">Madhya Pradesh</option>

<option value="Maharashtra">Maharashtra</option>

<option value="Manipur">Manipur</option>

<option value="Meghalaya">Meghalaya</option>

<option value="Mizoram">Mizoram</option>

<option value="Nagaland">Nagaland</option>

<option value="Orissa">Orissa</option>

<option value="Punjab">Punjab</option>

<option value="Rajasthan">Rajasthan</option>

<option value="Sikkim">Sikkim</option>

<option value="Tamil Nadu">Tamil Nadu</option>

<option value="Telangana">Telangana</option>

<option value="Tripura">Tripura</option>

<option value="Uttar Pradesh">Uttar Pradesh</option>

<option value="Uttarakhand">Uttarakhand</option>

<option value="West Bengal">West Bengal</option>

</select>

<label for="comment">Opinion</label>

<input type="text" id="opinion" name="opinion" placeholder="Please enter your opinion on Demonetization.." onBlur="checkopinionEmpty()" required><span id="opn" style="color:red"></span><br>

<input type="submit" value="Submit">

</form>

</div>

</body>

</html>

**db.php**

<?php

$connect=mysqli\_connect('localhost','root','','test');

if(mysqli\_connect\_errno($connect))

{

echo 'Failed to connect';

}

else

{

echo '';

}

// create a variable

$my\_name=$\_POST['name'];

$my\_age=$\_POST['age1'];

$my\_occ=$\_POST['occu'];

$my\_state=$\_POST['state'];

$my\_opinion=$\_POST['opinion'];

//Execute the query

$sql = "INSERT INTO fem (name, age1, occu, state, opinion) VALUES ('$my\_name',

'$my\_age', '$my\_occ', '$my\_state', '$my\_opinion')";

if(mysqli\_query($connect, $sql)){

echo "<br>Thank you for sharing your opinion.";

} else{

echo "ERROR: Could not able to execute $sql. " . mysqli\_error($connect);

}

$name=isset($\_POST["name"])?$\_POST["name"]:"";

$age1=isset($\_POST["age1"])?$\_POST["age1"]:"";

$occu=isset($\_POST["occu"])?$\_POST["occu"]:"";

$state=isset($\_POST["state"])?$\_POST["state"]:"";

$opinion=isset($\_POST["opinion"])?$\_POST["opinion"]:"";

mysqli\_close($connect);

?>

<html>

<body>

<input type="Submit" name="s1" value="Click here to go back!!"

onClick="location.href='main.html'">

</body>

</html>

**Sentiment analysis steps for Analyzing Twitter Data set:**

**Step 1: First load the tweet dataset in pig**

data= LOAD 'tweets.csv' USING PigStorage(',') as (sid:int,text:chararray,favorited:chararray,favoriteCount:int,replyToSN:chararray,created:chararray,truncated:chararray,replyToSID:int,id:int,replyToUID:int,statusSource:chararray,screenName:chararray,retweetCount:int,isRetweet:chararray,retweeted:chararray);

**Step 2: Extract id and text from the dataset**

extract\_details = FOREACH load\_tweets GENERATE $0 as id,$1 as text;

**Step 3: Now for the text we need to divide it into words to calculate sentiment analysis**

tokens = foreach extract\_details generate id,text, FLATTEN(TOKENIZE(text)) As word;

**Step 4:Now load the dictionary file in pig**

dictionary = load 'AFINN.txt' using PigStorage('\t') AS(word:chararray,rating:int);

**Step 5: Perform a map side join by joining the tokens statement and the dictionary contents using this relation:**

word\_rating = join tokens by word left outer, dictionary by word using 'replicated';

**Step 6:Now we will extract the id,tweet text and word rating(from the dictionary) by using the below relation.**

rating = foreach word\_rating generate tokens::id as id,tokens::text as text, dictionary::rating as rate;

**Step 7:Now, we will group the rating of all the words in a tweet by using the below relation:**

word\_group = group rating by (id,text);

**Step 8:Now, let’s perform the Average operation on the rating of the words per each tweet.**

avg\_rate = foreach word\_group generate group, AVG(rating.rate) as tweet\_rating;

Now we have calculated the Average rating of the tweet using the rating of each word.

From the above relation, we will get all the tweets i.e., both positive and negative.

**Step 9: Now we will filter the positive tweets using the below statement:**

positive\_tweets = filter avg\_rate by tweet\_rating>=0;

**Step 10:negative tweets:**

negative\_tweets = filter avg\_rate by tweet\_rating<0;

**Step 11: To know the count first we need to group the tweets individually.**

grp= group positive\_tweets all;

pos\_count= foreach grp generate COUNT(positive\_tweets);

same with negative tweets

neg= group negative\_tweets all;

neg\_count= foreach neg generate COUNT(negative\_tweets);

**Sentiment Analysis for the data collected from the application:**

**Step 1:** First collect the data from the application. The database used in this application generates the file that contains all the data gathered from the application.

**Step 2:** Copy this file to the hadoop environment and perform the following steps using Pig.

**Step 3:Load the dataset in pig.**

data=load 'fem.csv' using PigStorage(',') as (name:chararray,age1:int,occu:chararray.state:chararray,opinion:chararray);

**Step 4: Extract occupation, state and opinion details from the dataset:**

extract\_details = FOREACH data GENERATE $2 as occupation,$3 as state,$4 as text;

**Step 5: Now for the text we need to divide it into words to calculate sentiment analysis:**

tokens = foreach extract\_details generate occupation,state,text, FLATTEN(TOKENIZE(text)) As word;

**Step 6: Now load the dictionary file in pig:**

dictionary = load 'AFINN.txt' using PigStorage('\t') AS(word:chararray,rating:int);

**Step 7: Perform a map side join by joining the tokens statement and the dictionary contents using this relation:**

word\_rating = join tokens by word left outer, dictionary by word using 'replicated';

**Step 8: Now we will extract the occupation, state, text and word rating(from the dictionary) by using the below relation.**

rating = foreach word\_rating generate tokens::occupation as occupation,tokens::state as state,tokens::text as text, dictionary::rating as rate;

**Step 9: Now, we will group the rating by occupation, state and opinion in the dataset by using the below relation:**

word\_group = group rating by (occupation,state,text);

**Step 10: Now, let’s perform the Average operation on the rating of the words for each opinion.**

avg\_rate = foreach word\_group generate group, AVG(rating.rate) as opinion\_rating;

Now we have calculated the Average rating of the text using the rating of each word.

From the above relation, we will get all the opinions i.e., both positive and negative.

**Step 11:Now we will filter the positive opinions using the below statement:**

positive\_opinion = filter avg\_rate by opinion\_rating>=0;

For negative opinions:

negative\_opinions= filter avg\_rate by opinion\_rating<0;

**STEP 12: To know the count first we need to group the opinions individually.**

grp= group positive\_opinions all;

pos\_count= foreach grp generate COUNT(positive\_opinions);

same with negative opinions

neg= group negative\_opinions all;

neg\_count= foreach neg generate COUNT(negative\_opinions);

TESTING

**5.1 Introduction:**

Testing is the process of finding differences between the expected behavior specified by system models and the observed behavior of the system. Testing is a critical role in quality assurance and ensuring the reliability of development and these errors will be reflected in the code so the application should be thoroughly tested and validated.

Unit testing finds the differences between the object design model and its corresponding components. Structural testing finds differences between the system design model and a subset of integrated subsystems. Functional testing finds differences between the use case model and the system.

Finally performance testing, finds differences between non-functional requirements and actual system performance. From modeling point of view, testing is the attempt of falsification of the system with respect to the system models. The goal of testing is to design tests that exercise defects in the system and to reveal problems.

**Unit Testing:**

Unit testing focuses on the building blocks of the software system, that is, objects and subsystems. It reduces the complexity of overall test activities. Unit testing makes it easier to pinpoint and correct faults. It allows parallelism in the testing activities. In this test each component is tested independent of the other thus allowing parallelism in testing activity.

Different types of testing techniques are available and the most important ones are equivalence testing, boundary testing, path testing and state-based testing, polymorphism testing.

**Ex:** Individual units like scanning the network, collecting the details of the remote system and also the module to view the details are individual units.

**Integration Testing:**

Once faults in each component have been removed and the test cases do not reveal any new faults, components are ready to be integrated into larger subsystems. Integration testing detects faults that have not been detected during unit testing by focusing on small groups of components. Two or more components are integrated and tested, and when no new faults are revealed, additional components are added to the group. Several approaches have been devised to implement an integration testing: big bang testing, bottom-up testing, top-down testing, and sandwich testing.

Many tested modules are combined into subsystem. Which are then tested? The emphasis is on testing interface between modules. This activity can be considered to test the design.

**Ex:** Suppose currently we are working with administrator module and we had a requirement to add a new node to the system. For this we must be able to scan the network and also collect the details of the system and the collected information must be available to generate reports to the administrator. So for that we must test the system whether it is possible to do like that. That can be achieved through Integration testing.

**System Testing:**

Unit testing and Integration testing focuses on finding faults in individual components and the interfaces between the components. Once components have been integrated, system testing ensures that the complete system complies with the functional and nonfunctional requirements. Several system testing activities are performed: Functional testing, Performance testing, Pilot testing, Acceptance testing, Installation testing.

This is a series of testing whose purpose is to fully exercise the entire software system. System testing verifies that all elements mesh properly and the overall system function/performance is achieved. System testing is done using real data. The system performance was found to be working as per requirements specified.

**Ex:** After integration system is done now we need to test the overall system functionality after integrating all the modules.

**5.2 Testing Approaches:**

There are three basic approaches to testing:

* Functional (Black box testing)
* Conditional
* Structural (White box testing)

**5.2.1 White box Testing:**

Test cases are classified depending on which aspect of the system model is tested. Whitebox tests focuses on the internal structure of the component. A Whitebox tests make sure that, independently from the particular input/output behavior, every state in the dynamic model of the object and every interaction among the objects is tested.

This test focuses on the program control structure. Test cases are derived to ensure that all the statements in the program have been executed at least once during testing and that all logical conditions have been exercised.

**5.2.2 Condition testing:**

Condition testing is a test case design method that exercises the logical conditions contained in a module the condition testing focuses on testing each condition in the operation.

**5.2.3 Black box Testing:**

Blackbox tests focuses on the input/output behavior of the component. Blackbox tests do not deal with the internal aspects of the component, nor with the behavior or the structure of the components.

These tests are designed to validate functional requirements without regard to working of the program. Black box testing technique focuses on the information domain of the software deriving test cases by partitioning input and output in a manner that provides thorough test coverage. This approach also deals with the identification of class libraries or software packages or an other application systems, which are already present so that developer can implement the object models with a decreased burden of testing.

Unit testing combines both testing techniques: blackbox testing to test the functionality of the component, and whitebox testing to test structural and dynamic aspects of the component.

**5.3 Test case execution and analysis**

**User filling form details**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Test Case*** | ***Condition Being Checked*** | **Expected Output** | **Actual Output** |
| Any compulsory field not selected | Checking required fields | Please fill required fields | Please fill required fields |
| Incorrect name | Read name | Name should be in characters | Name should be in characters |
| Invalid Age | Read Age | Age must be in numerical values | Age must be in numerical values |

**Validations on file**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Test Case*** | ***Condition Being Checked*** | **Expected Output** | **Actual Output** |
| When a dataset is selected for analysis | Checking the format of the file | File Format is invalid | File Format is invalid |
| When the dataset is copied to hadoop environment. | Checking whether the data is present in the file | Data is present in the file | There is no data in the file |

**Result**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Test Case*** | ***Condition Being Checked*** | **Expected Output** | **Actual Output** |
| Performing visualization | Selecting the fields for visualization | Please fill select the correct field | Please fill select the correct field |
| Invalid data for visualization | Checking data available for visualization | Visualization is done. | Visualization cannot be done |
| Submitting valid data for visualization | Generate report | Redirect to visualization page | Redirect to visualization page |

# RESULTS AND REPORTS

**Main Screen**

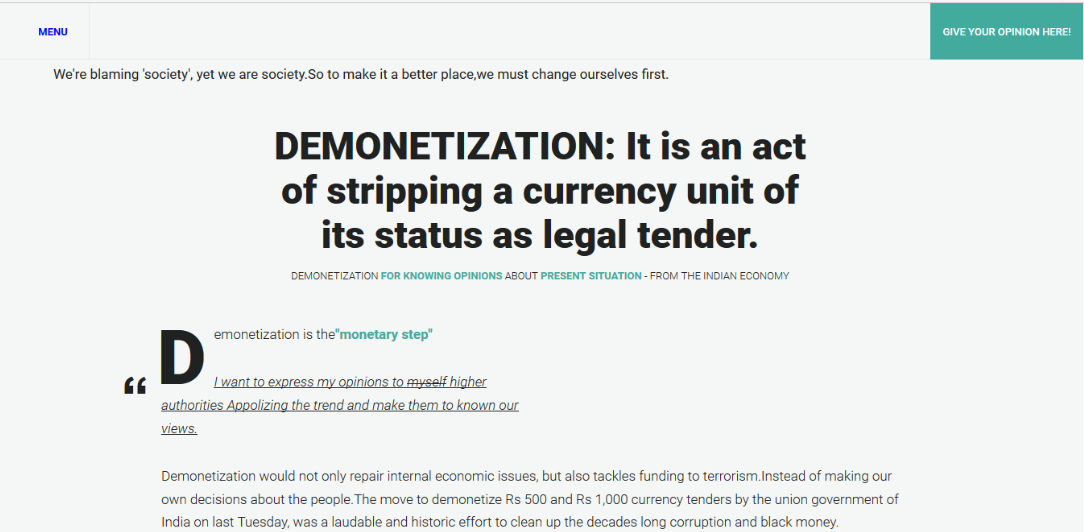
****

Figure 6.1 Home Page

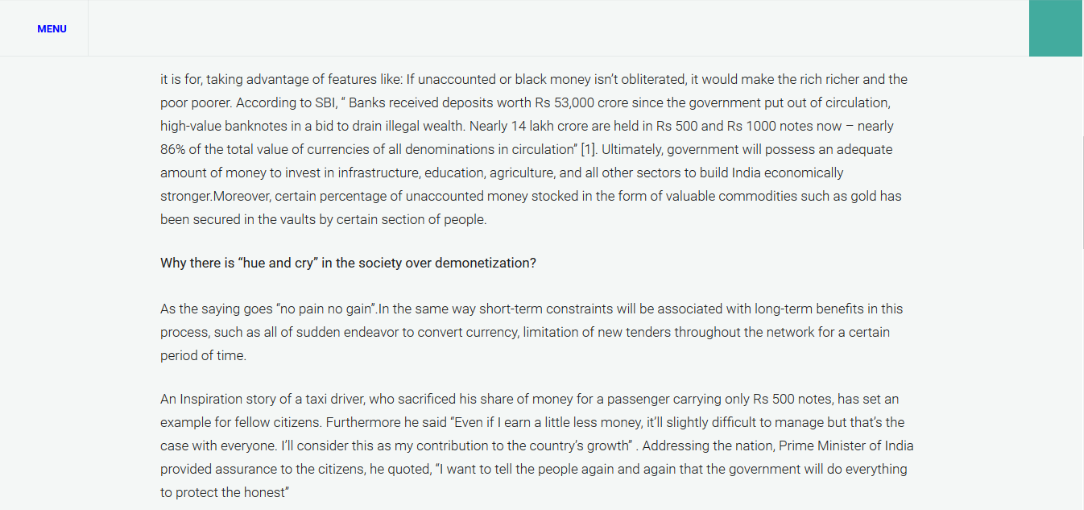


Figure 6.1.2

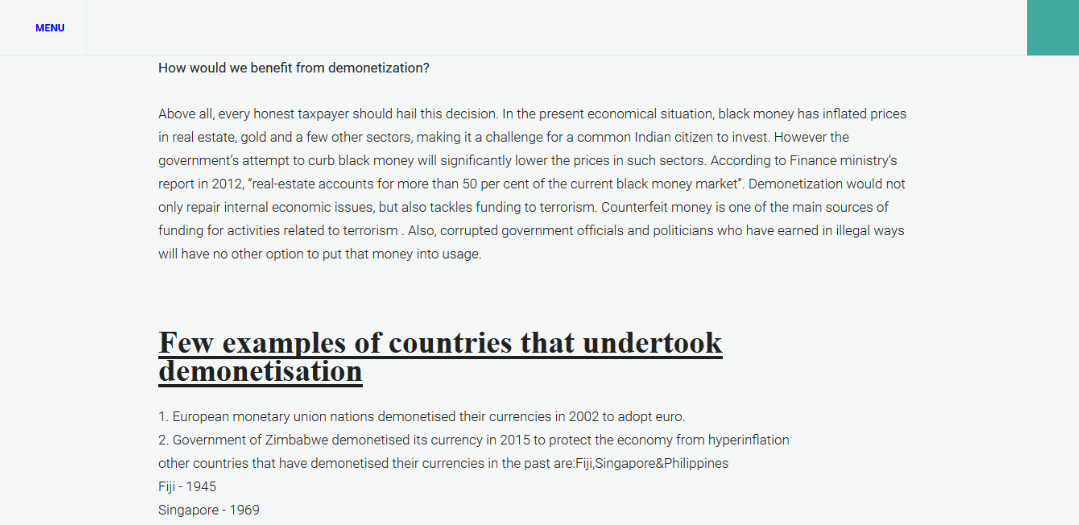
****

Figure 6.1.3

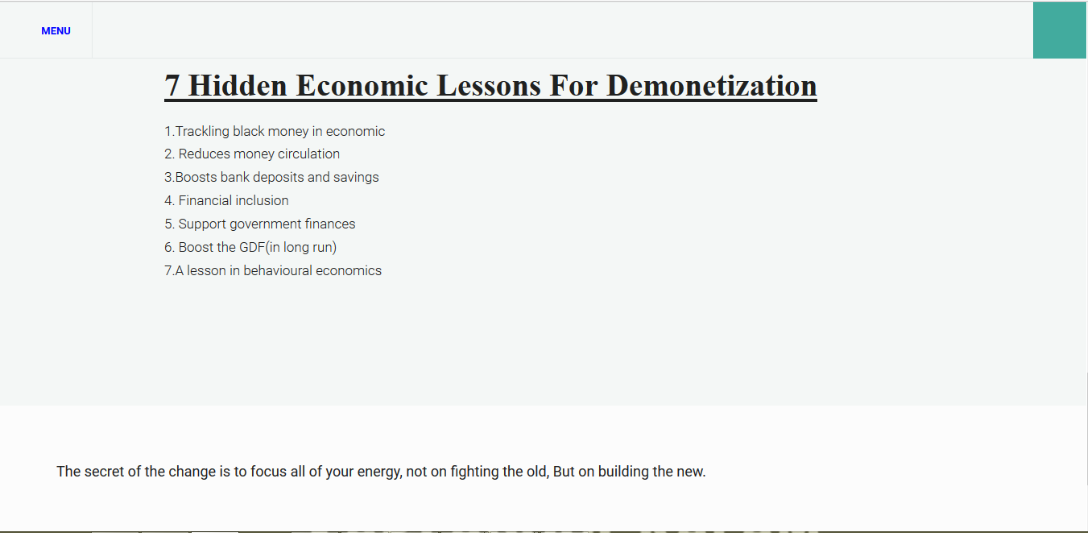


Figure 6.1.4

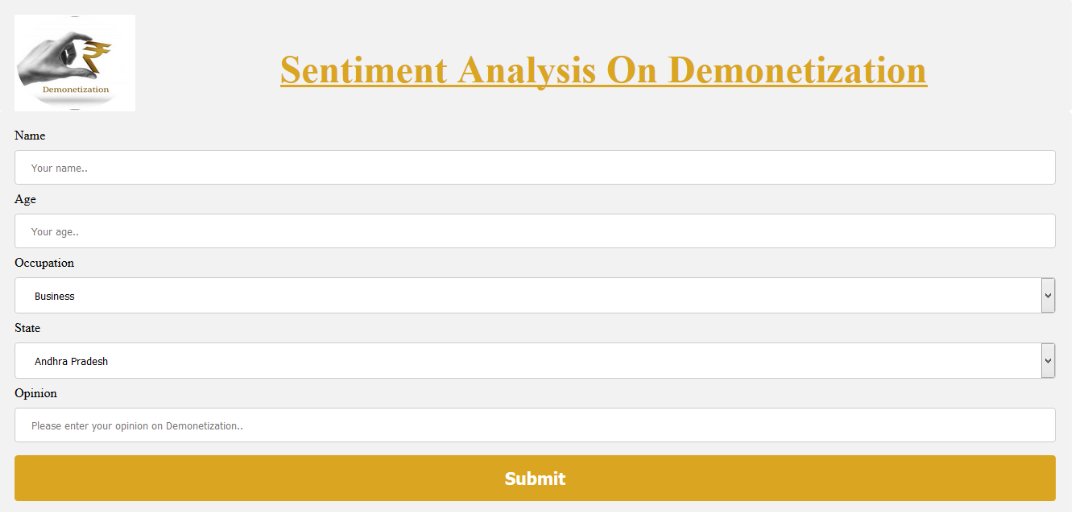
****

Figure 6.2 Opinion Screen

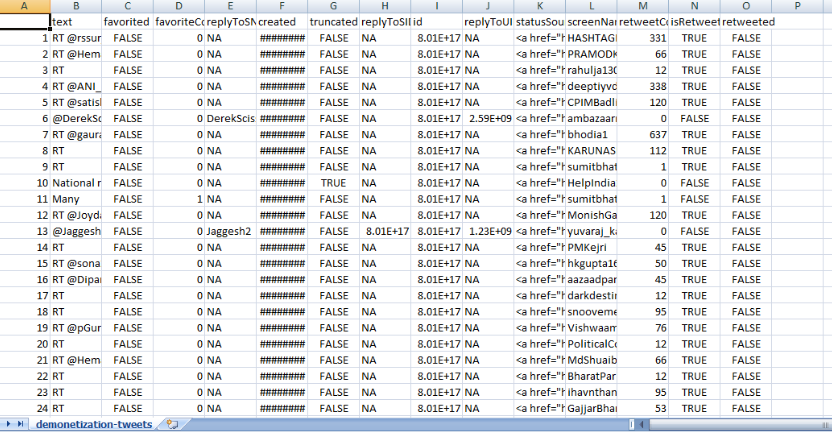
****

Figure 6.3 Twitter DataSet(Total 8000 records)

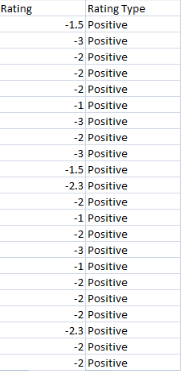
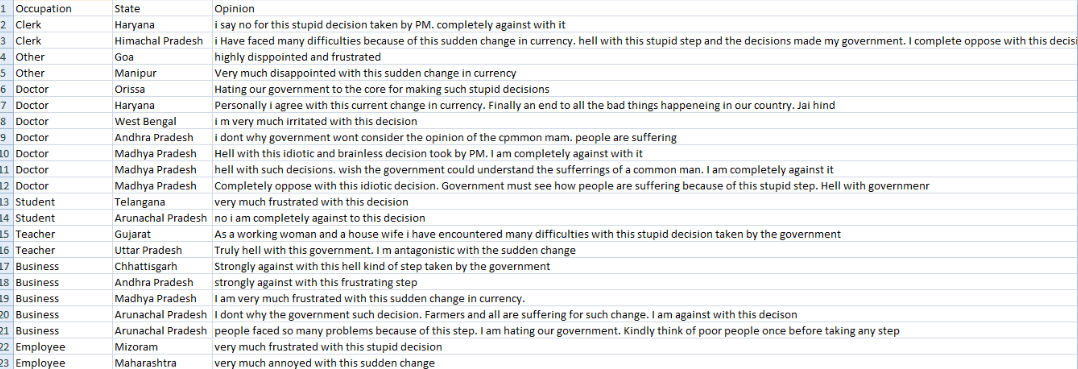
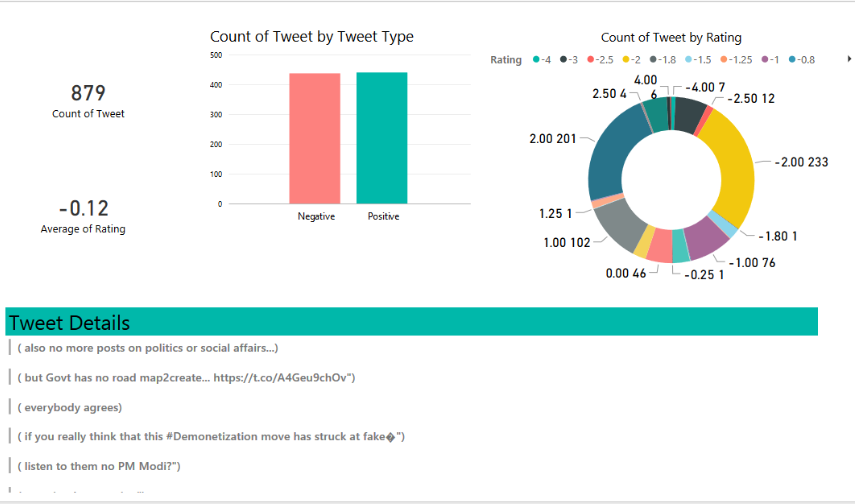


Figure 6.4 DataSet from the application

**:**

****

**Figure 6.5 Visualization of Twitter DataSet after performing sentiment analysis**

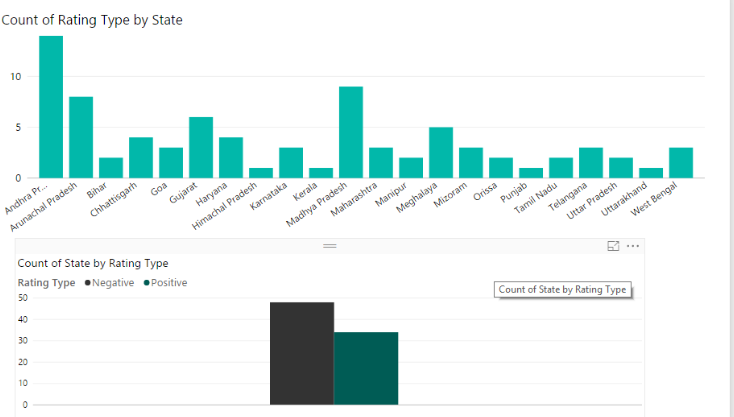
****

Figure 6.6 Visualization based on States

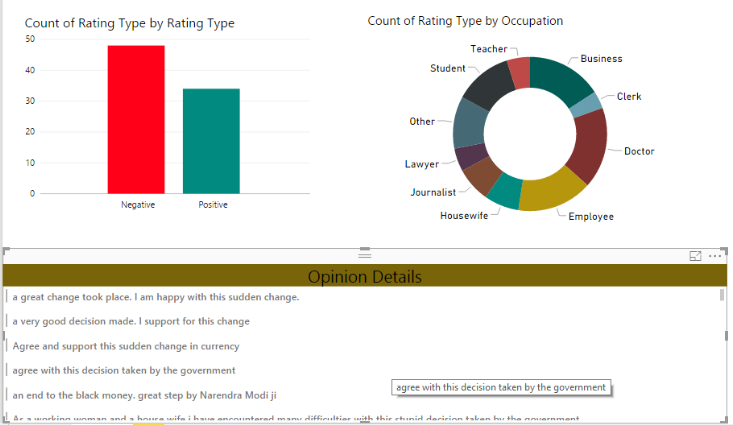
****

Figure 6.7 Visualization Based on Occupation

# CONCLUSION

**Conclusion:**

This application presents the part of an organization work as per the requirements, specifications and conditions mentioned in the user manual. This application has been developed and completed successfully and also tests successfully by taking “Test Cases”. It is user friendly and has all the needed menu options, which can be utilized by the user to perform the desired operations. Moreover help messages are provided which will help any authorized user from using the system without trouble.

The application overcomes all traditional limitations presented in real time applications. As this application is secure, reliable, fast, simple and advanced so this can be highly adoptable to deploy and use.

# FUTURE SCOPE

**Future Scope:**

Dynamic analysis and visualization can be done. A bot can be created for collecting feedback. Different kind of languages can be implemented for accessing this application.

# BIBLIOGRAPHY

**Bibliography:**

We had a lot of help online and these are some mentions of all the references, which helped us in our development process and ended up with us finishing the application with flying colors.

**Book References:**

1. Unified Modeling Language by Grady Booch

2. An Introduction to data structures by [Jean-Paul Tremblay](http://www.amazon.in/s/ref=dp_byline_sr_book_1?ie=UTF8&field-author=Jean-Paul+Tremblay&search-alias=stripbooks), [Paul Sorenson](http://www.amazon.in/s/ref=dp_byline_sr_book_2?ie=UTF8&field-author=Paul+Sorenson&search-alias=stripbooks)

3. Easy Java Web by Siva Ramaya Madhupada

**Online Reference**

1. https://www.brandwatch.com/blog/understanding-sentiment-analysis/

2. <http://www.stackoverflow.com>

3. <https://www.tutorialspoint.com>

4. https://pig.apache.org/

5. https://hortonworks.com/apache/pig/

6. http://blog.enablecloud.com/2014/06/sentiment-analysis-using-pig.html