CONTENT

1.INTRODUCTION TO ONLINE VOTING SYSTEM :

"ONLINE VOTING SYSTEM" is an online voting technique. In this system people who have citizenship of India and whose age is above 18 years of age and any sex can give his\her vote online without going to any physical polling station. There is a database which is maintained in which all the names of voters with complete information is stored.

In "ONLINE VOTING SYSTEM" a voter can use his\her voting right online without any difficulty. He\She has to be registered first for him/her to vote. Registration is mainly done by the system administrator for security reasons. The system Administrator registers the voters on a special site of the system visited by him only by simply filling a registration form to register voter. Citizens seeking registration are expected to contact the system administrator to submit their details. After the validity of them being citizens of India has been confirmed by the system administrator by comparing their details submitted with those in existing databases such as those as the Registrar of Persons, the citizen is then registered as a voter.

After registration, the voter is assigned a secret Voter ID with which he/she can use to log into the system and enjoy services provided by the system such as voting. If invalid/wrong details are submitted, then the citizen is not registered to vote.

2.BACKGROUND OF STUDY:

The Online voting system (OVS) also known as e-voting is a term encompassing several different types of voting embracing both electronic means of counting votes. Electronic voting technology can include punched cards, optical scan voting systems and specialized voting kiosks (including self contained direct-recording electronic voting systems or DRE). It can also involve transmission of ballots and votes via telephones, private computer networks, or the internet.

Online voting is an electronic way of choosing leaders via a web driven application. The advantage of online voting over the common "queue method" is that the voters have the choice of voting at their own free time and there is reduced congestion. It also minimizes on errors of vote counting. The individual votes are submitted in a database which can be queried to find out who of the aspirants for a given post has the highest number of votes.

This system is geared towards increasing the voting percentage in India since it has been noted that with the old voting method {the Queue System}, the voter turnout has been a wanting case. With system in place also, if high security is applied, cases of false votes shall be reduced.

With the "ONLINE VOTING SYSTEM", a voter can use his\her voting right online without any difficulty. He\She has to register as a voter first before being authorized to vote. The registration should be done prior to the voting date to enable data update in the database.

However, not just anybody can vote. For one to participate in the elections, he/she must have the requirements. For instance, he/she must be a registered citizen i.e. must be 18 and above years old. As already stated, the project 'Online Voting' provides means for fast and convenient voting and access to this system is limited only to registered voters.

Internet voting systems are appealing for several reasons which include; People are getting more used to work with computers to do all sorts of things, namely sensitive operations such as shopping and home banking and they allow people to vote far from where they usually live, helping to reduce absenteeism rate.

3.SIGNIFICANCE OF STUDY :

The main purposes of OVS include:

- Provision of improved voting services to the voters through fast, timely and convenient voting.
- Reduction of the costs incurred by the Indian Electoral Commission during voting time in paying the very many clerks employed for the sake of the success of the manual system.
- Check to ensure that the members who are registered are the only ones to vote. Cases of “Dead People” voting are also minimized.
- Online voting system (OVS) will require being very precise or cost cutting to produce an effective election management system.
- Therefore crucial points that this (OVS) emphasizes on are listed below.

  i. Require less number of staff during the election.

  ii. This system is a lot easier to independently moderate the elections and subsequently reinforce its transparency and fairness.

  iii. Less capital, less effort, and less labor intensive, as the primary cost and effort will focus primarily on creating, managing, and running a secure online portal.

  iv. Increased number of voters as individual will find it easier and more convenient to vote, especially those abroad.

4.Objectives of this project :

The specific objectives of the project include:

- ❖ Reviewing the existing/current voting process or approach in India.
- ❖ Coming up with an automated voting system in India.
- ❖ Implementing an automated/online voting system.
- ❖ Validating the system to ensure that only legible voters are allowed to vote.

5.Project justification :

The ONLINE VOTING SYSTEM-INDIA shall reduce the time spend making long queues at the polling stations during voting. It shall also enable the voters to vote from any part of the globe as explained since this is an online application available on the internet. Cases of vote miscounts shall also be solved since at the backend of this system resides a well developed database using MYSQL that can provide the correct data once it’s correctly queried. Since the voting process shall be open as early as possible, the voters shall have ample time to decide when and whom to vote for.

6.SCOPE OF STUDY :

It is focused on studying the existing system of voting in India and to make sure that the peoples vote is counts, for fairness in the elective positions. This is also will produce:

- Less effort and less labor intensive, as the primary cost and focus primary on creating, managing, and running a secure web voting portal.
- Increasing number of voters as individuals will find it easier and more convenient to vote, especially those abroad.

7. Old Methods Of Voting :

1. **Paper-based voting**: The voter gets a blank ballot and use a pen or a marker to indicate he want to vote for which candidate. Hand-counted ballots is a time and labor consuming process, but it is easy to manufacture paper ballots and the ballots can be retained for verifying, this type is still the most common way to vote.

2. **Lever voting machine**: Lever machine is peculiar equipment, and each lever is assigned for a corresponding candidate. The voter pulls the lever to poll for his favorite candidate. This kind of voting machine can count up the ballots automatically. Because its interface is not user-friendly enough, giving some training to voters is necessary.

3. **Direct recording electronic voting machine**: This type, which is abbreviated to DRE, integrates with keyboard; touch screen, or buttons for the voter press to poll. Some of them lay in voting records and counting the votes is very quickly. But the other DRE without keep voting records are doubted about its accuracy.

4. **Punch card**: The voter uses metallic hole-punch to punch a hole on the blank ballot. It can count votes automatically, but if the voter's perforation is incomplete, the result is probably determined wrongfully.

5. **Optical voting machine**: After each voter fills a circle correspond to their favorite candidate on the blank ballot, this machine selects the darkest mark on each ballot for the vote then computes the total result.

8.SECURITY ISSUES OF ONLINE VOTING :

Foreign experience revealed that they are often confronted by security issues while the online voting system is running. The origin of the security issues was due to not only outsider (such as voters and attackers) but also insider (such as system developers and administrators), even just because the inheritance of some objects in the source code are unsuitable. These errors caused the voting system to crash.

The proposed solutions were correspondingly outlined to hold back these attacks. For example, to avoid hacker making incursion into the voting system via network, we can design our system to transmit data without network. Another example is to limit voter to input particular data, so that we can prevent the command injection from running

9.Requirements:

1) Registration of the voter is done by ELECTION COMMISION OF INDIA.
2) ELECTION COMMISION OF INDIA can change the information any time if required.
3) Registration of the Voter depends upon the information filled by the user.
4) Voter is given a unique ID and PASSWORD.
5) In the DATABASE information of every voter is stored.
6) Database shows the information of every user.

10.Problems with the Existing Voter Registration System

The problems of the existing manual system of voting include among others the following:

1. **Expensive and Time consuming**: The process of collecting data and entering this data into the database takes too much time and is expensive to conduct, for example, time and money is spent in printing data capture forms, in preparing registration stations together with human resources, and there after advertising the days set for registration process including sensitizing voters on the need for registration, as well as time spent on entering this data to the database.

2. **Too much paper work**: The process involves too much paper work and paper storage which is difficult as papers become bulky with the population size.

3. **Errors during data entry:** Errors are part of all human beings; it is very unlikely for humans to be 100 percent efficient in data entry.

4. **Loss of registration forms:** Some times, registration forms get lost after being filled in with voters' details, in most cases these are difficult to follow-up and therefore many remain unregistered even though they are voting age nationals and interested in exercising their right to vote.

5. **Short time provided to view the voter register:** This is a very big problem since not all people have free time during the given short period of time to check and update the voter register.

6. Above all, a number of voters end up being locked out from voting.

11.SOFTWARE USED FOR THIS PROJECT:

i. **MYSQL DBMS-** It allows combination, extraction, manipulation and organization of data in the voters' database. It is platform independent and therefore can be implemented and used across several such as Windows, Linux server and is compatible with various hardware mainframes. It is fast in performance, stable and provides business value at a low cost.

ii. **Android Studio**

iii. **JAVA coding-**This is for advanced user who find PHP codes easy to work with.

iv. **phpWeb browsers**: Mozilla Firefox, Google chrome, Opera and Internet Explorer

v. **Xml Coding.**

12.HARDWARE REQUIREMENTS:

- **Android Phone.**
    - **Processor:** Dual core or equivalent
    - **Memory:** 112 MB
    - **Disk space:** 300 MB of free disk space
    - **Android Version :** Minimum requirement 4.1 Jellybean.

13.User Characteristics :

This system will be used in giving voting online.it having 3 module which are Administrator, Voters and candidates. As all of these have different requirements, the module to meet their need and avoid and any type of confusion. All module are explained bellow ;

Admin :

He/She can add new candidates inside the voting process.

Voters:

Voters can give the votes to the candidates.

Voters ca watch who is winner and how much vote having candidates.

Candidates :

Having Nothing Special work for the candidates. Candidates only watch the result and notice.

14.Security And Reliability Requirements :

Security :

The information hold be secure, there should not be any kind of malfunctioning. All the result will store in securely in the database system. System information will not be change by any person rather than management.

Reliability :

System should not reliable. It should keep secure.

Flexibility :

System is working easily on the Internet with the username and password of the user. The industry has given the right to all the users to use the process by their username and password.

Efficiency :

System should not be efficient enough to all the meet all kind of the requirements as required by the administrator and voters. The system should not hang or lose is efficiency in any kind of worse condition.

15.SYSTEM STUDY

NEEDS OF THE SYSTEM

Social and economic factor : a wave of social and economies changes often follows in the wake of the new technology. New opportunity are arise to improve on voting process or do something that was not previously possible. Change the individuals are organized into groups may then be necessary, and the new groups may complete economic recourses.

Technological factor: people have never before in a time when the scope of scientific inquire as a broad, So when the speed applying the new technology to change the voting process.

16.SYSTEM ANALYSIS

System Analysis is a process by which we attribute process or goals to a human activity, determine how well those purpose are being achieved and specify the requirements of the various tools and techniques that are to be used within the system if the system performances are to be achieved.

1. SYSTEM PLANNING

Planning for information systems has a time horizon and a focus dimension. The time horizon dimension specifies the time range of the plan, where as the focus dimension relates whether the is strategic, managerial, or operational. The system i.e. The Project that we were assigned was required to complete with in 20 weeks. What we had planned is as follows:
Requirement analysis, Preliminary Investigation Information Gathering should be covered within the 1 and 2 week, Since I was not aware of ANDROID STUDIO IDE it require 1 week for me to adjust with that tool. 12 Weeks for the design of the system under development. 1 week for Testing and implementation .And rest 2 week reserve week.

2. PRELIMINARY INVESTIGATION

The initial investigation has the objective of determining the validity of the users request for a candidate system and whether a feasibility study should be conducted.
The objectives of the problem posed by the user must he understood within the framework.

17.FEASIBILITY STUDY

Feasibility is the determination whether or not a project should taken or not. Feasibility study is undertaking tight constraint and culminated in the written and moral feasibility report. The content of this project were used as online voting process.

TECHNICAL FEASIBILITY

The main purpose of feasibility is to test whether with the existing H/w s/w the project can proceed. Science the Spice Board of India, Bhubaneswar has a required H/W Windows NT Based Server, Apache Web Server and Browsers Technical the system feasible.

OPERATIONAL FEASIBILITY

Any project can be beneficial only if it can meet the organizations operating requirements. The feasibility is very much important in order to find out whether the system is work after it is developed installed. Also it include the cations of the users, their remarks and their involvement in the project.

ECONOMICAL FEASIBILITY

A system that can be developed technically and that will be used if installed must still he a good investment for the organization. The organization possesses all the H/w, S/w and of course the trained personnel. So implementing the system will not cause any additional investment for the
Feasibility studies discussed above indicates that the system in feasible.
Following is a list of main requirements of the system

- The system should have a proper login for every Voters and Admin.

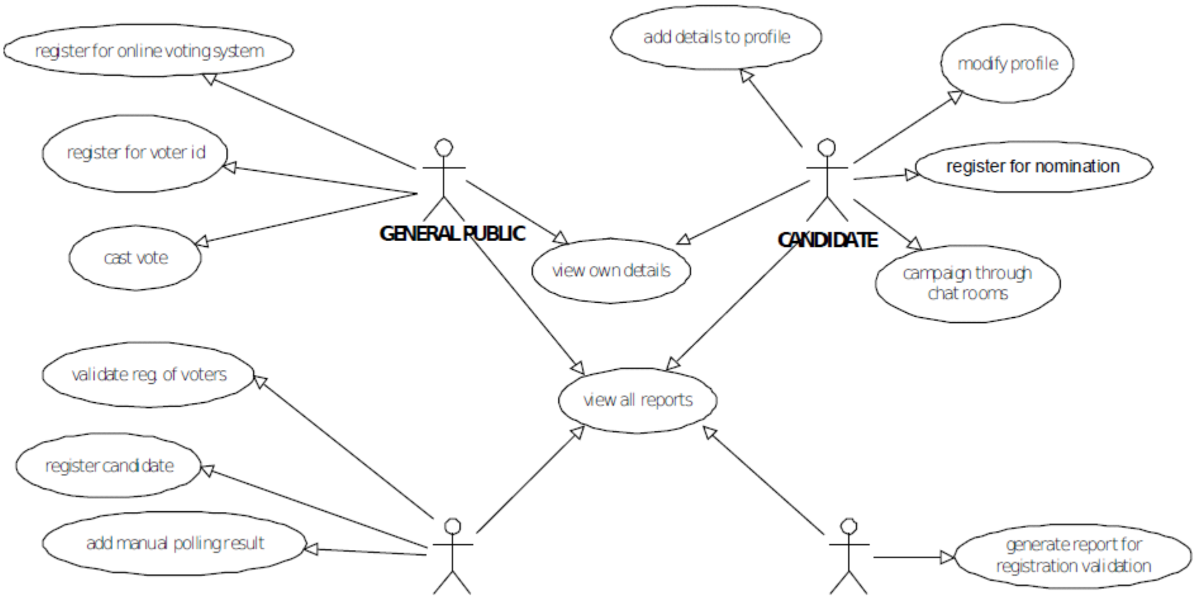
The Login form should appear when the system is invoked. System provides the login id and password.

- The system is cost effective.
- No Hardware Coast Required.
- This project is possible given the source constraint.
- Cost of packaged software development is less.

18.STRUCTURED ANALYSIS

The traditional tools of data gathering have limitations. System flowcharts focus on physical than on logical implementation of the candidate system. Because of these drawbacks, structured tools were introduced for analysis. Structured analysis is a set of techniques and graphical tools (DFD) that allow the analyst to develop a new kind of system specification that are easily understandable to the user

19.BLOCK DIAGRAM :



20.FRONT – END AND BACK-END :

In their most general meanings, the terms front end and back end refer to the initial and the end stages of a process flow. In software design, the **front-end** is the part of a software system that deals with the user, and the **back-end** is the part that processes the input from the front-end. The separation of software systems into "front ends" and "back ends" is a kind of abstraction that helps to keep different parts of the system separated. The general idea is that the front-end is responsible for collecting input from the user, which can be in a variety of forms, and processing it in such a way that it conforms to a specification that the back-end can use. The connection of the front-end to the back-end is a kind of interface.

Front-end and back-end are terms used to characterize program interfaces and services relative to the initial user of these interfaces and services. (The "user" may be a human being or a program.) A ”front-end” application is one that application users interact with directly. A "back-end" application or program serves indirectly in support of the front-end services, usually by being closer to the required resource or having the capability to communicate with the required resource. The back-end application may interact directly with the front-end or, perhaps more typically, is a program called from an intermediate program that mediates front-end and back-end activities. These terms acquire more special meanings in particular areas:-

(1) For software applications, front end is the same as user interface.

(2) In client/server applications, the client part of the program is often called the front end and the server part is called the back end.

(3) Compilers, the programs that translate source code into object code, are often composed of two parts: a front end and a back end. The front end is responsible for checking syntax and detecting errors, whereas the back end performs the actual translation into object code.

21.DATA FLOW DIAGRAM:

Data Flow Diagram is a diagrammatic representation of data movement through a system — manual or automated - from inputs to outputs through processing. These are of two types — Logical Data Flow Diagrams and Physical Data Flow Diagrams. The Data Flow Diagram (DFD) clarifies system requirements and identifies major transformations that will become programs in system design. It is the starting point of system design that decomposes the requirements specifications down to the lowest level of detail.

22.Logical Data Flow Diagrams:

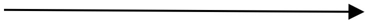
The Logical Data Flow Diagrams represent the transformation of the data from input to output through processing logically and independently of the physical components that may be associated with the system.

23.Physical Data Flow Diagrams

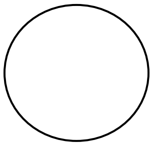
The Physical Dataflow Diagrams show the actual implementation and movement of data between people, departments, and workstations. Each component of a DFD is labeled with a descriptive name. Process names are further numbered that will be used for identification purposes. The number assigned to a specific process does not correspond to the sequence of processes. It is strictly for identification purposes. A data flow diagram allows parallel activities i.e. a number of data-flows coming out from the source and going into the destination. A DFD concentrates on the data moving through the system and not on the devices or equipments. A DFD may consist of a number of levels. The top-level diagram is called the Context Diagram, which consists of a single process and plays a very important role in studying the system.

NOTATIONS:

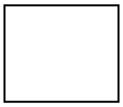
- Data- flows show the movement of data in a specific direction from the source to the destinantion . It represents a packet of data.



- Process show the operations perform on the data,which transform it from input to output



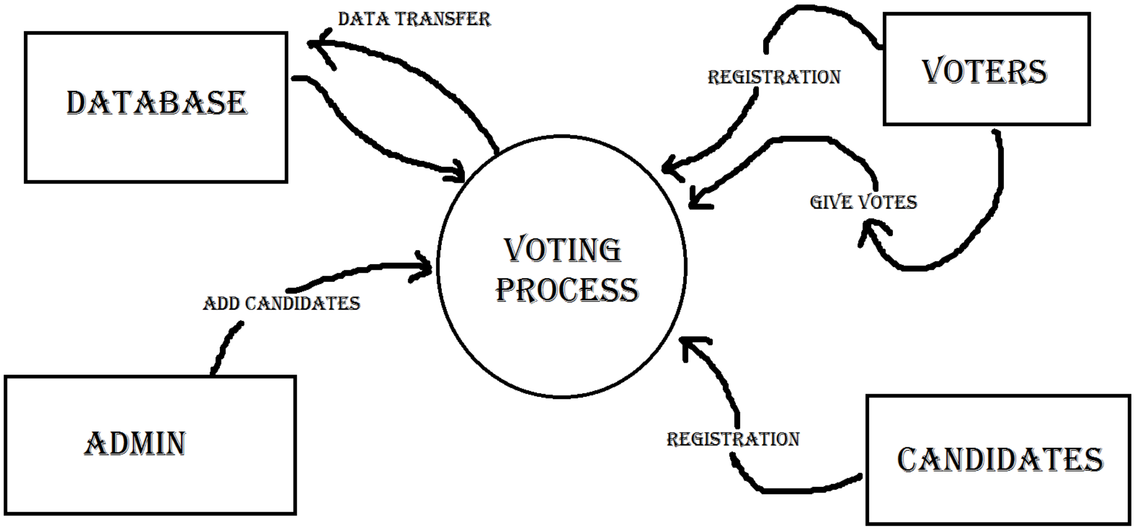
- Source and destiantion of data are external sources and destination of data ,which may be prople ,programs,organisations or other entities interacting with the system ,but are outside its Boundary



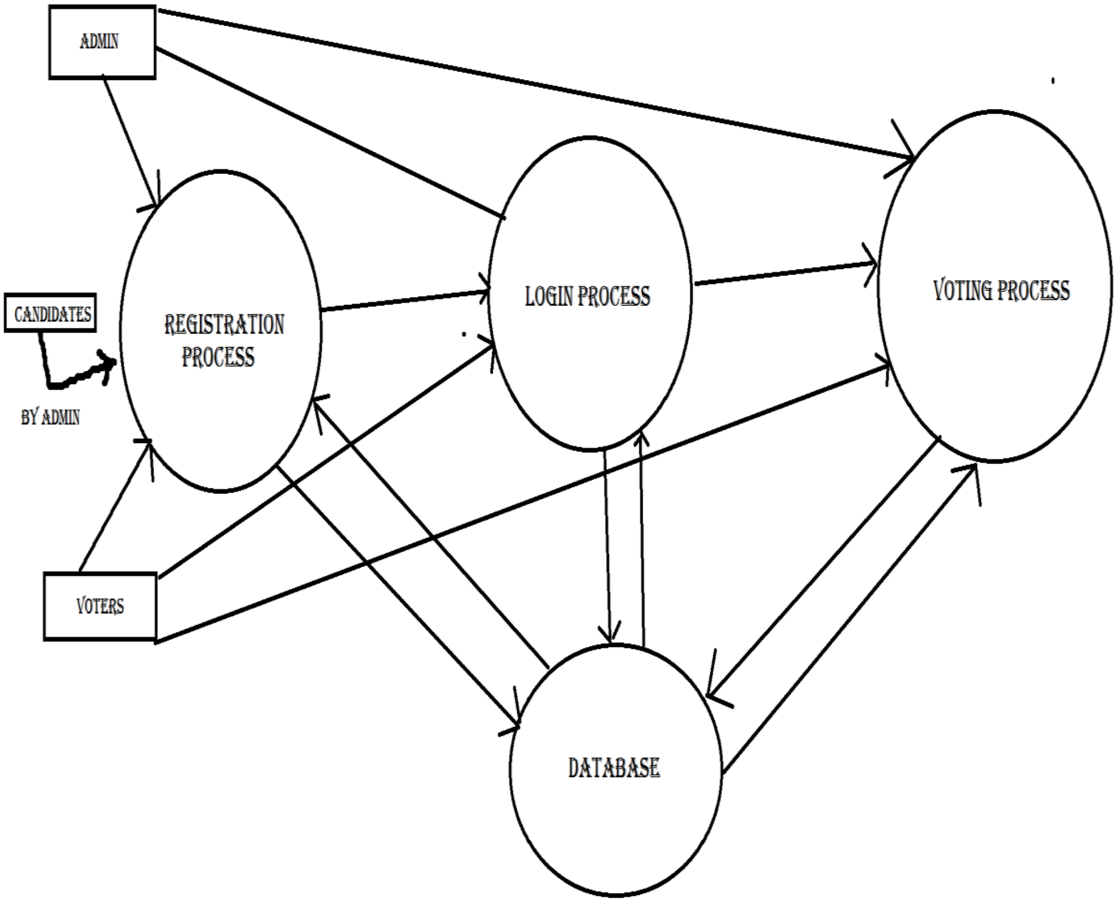
- Data stores are places where data are stored such as files and tables.

Below is the top level(DFD) showing how the user 's request processed by the server with database interaction and sends the response back to the user

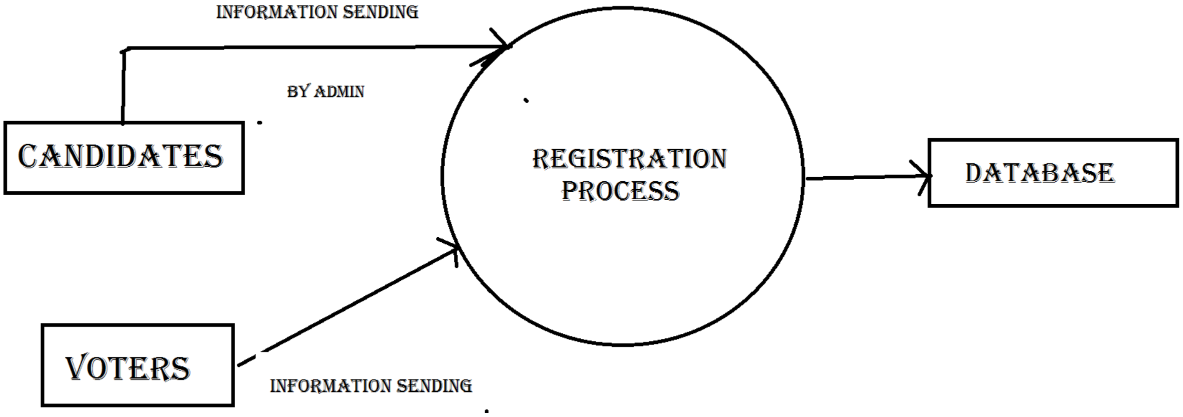
24.Context 0th level diagram:



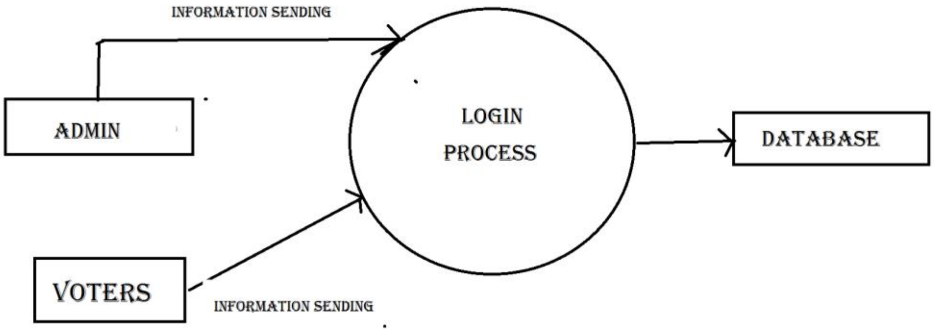
25.1st Level Data Flow Diagram:



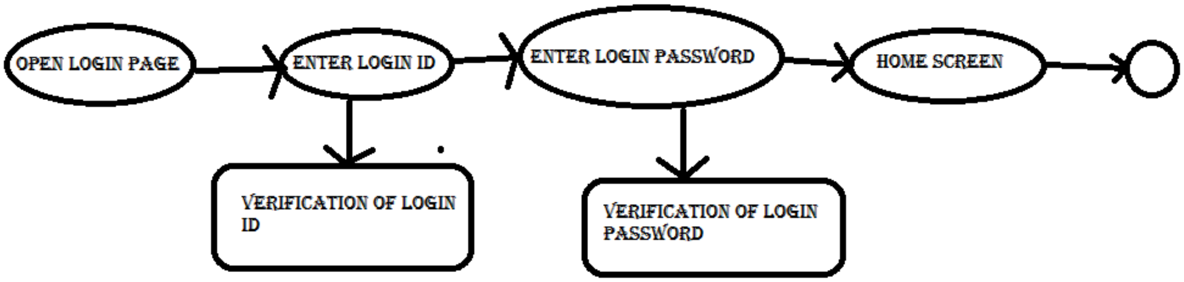
26.2nd Level Data Flow Diagram For Registration Process:



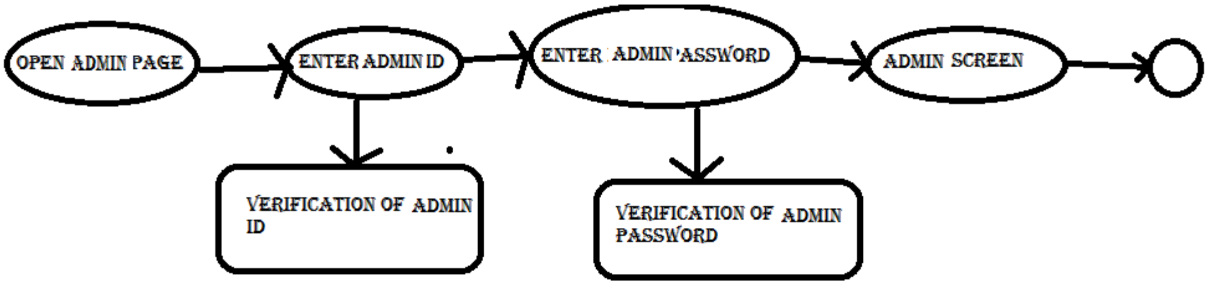
2nd Level Data Flow Diagram For Login Process:



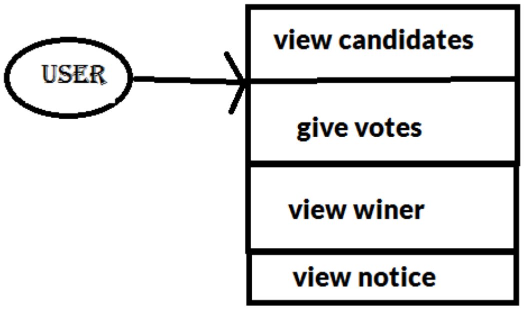
User Login 2nd Data Flow Diagram :



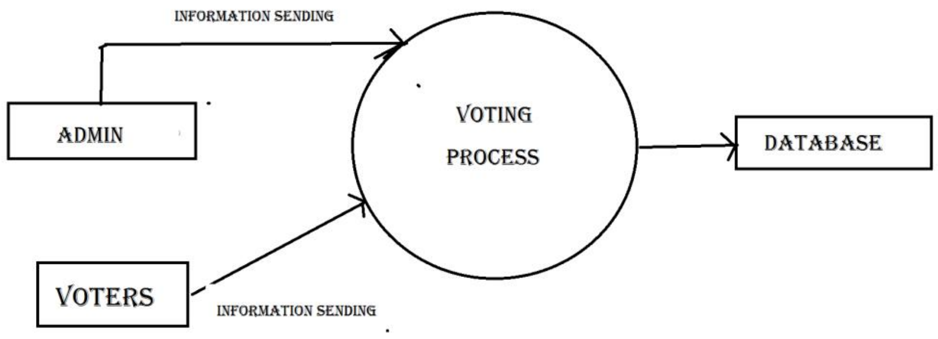
Admin Login 2nd Data Flow Diagram :



User 2nd Level Data Flow Diagram :



2nd Level Data Flow Diagram For Voting Process:



27.*Entity-Relationship Diagram (ERD):*

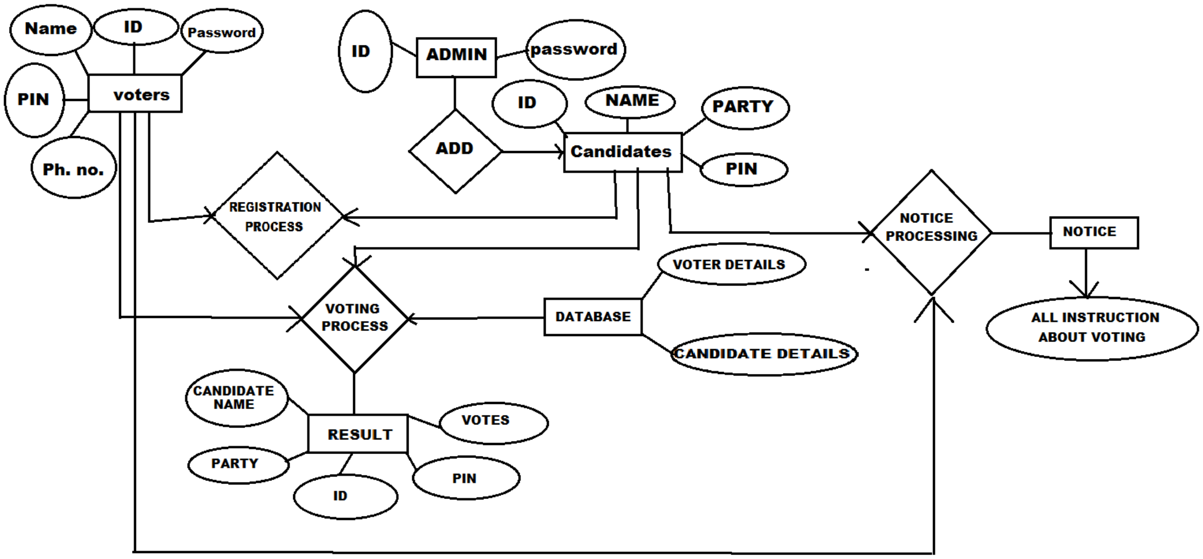
An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities. An ERD is a conceptual and representational model of data used to represent the entity framework infrastructure.

The elements of an ERD are:

- Entities
- Relationships
- Attributes

Steps involved in creating an ERD include:

1. Identifying and defining the entities
2. Determining all interactions between the entities
3. Analyzing the nature of interactions/determining the cardinality of the relationships
4. Creating the ERD



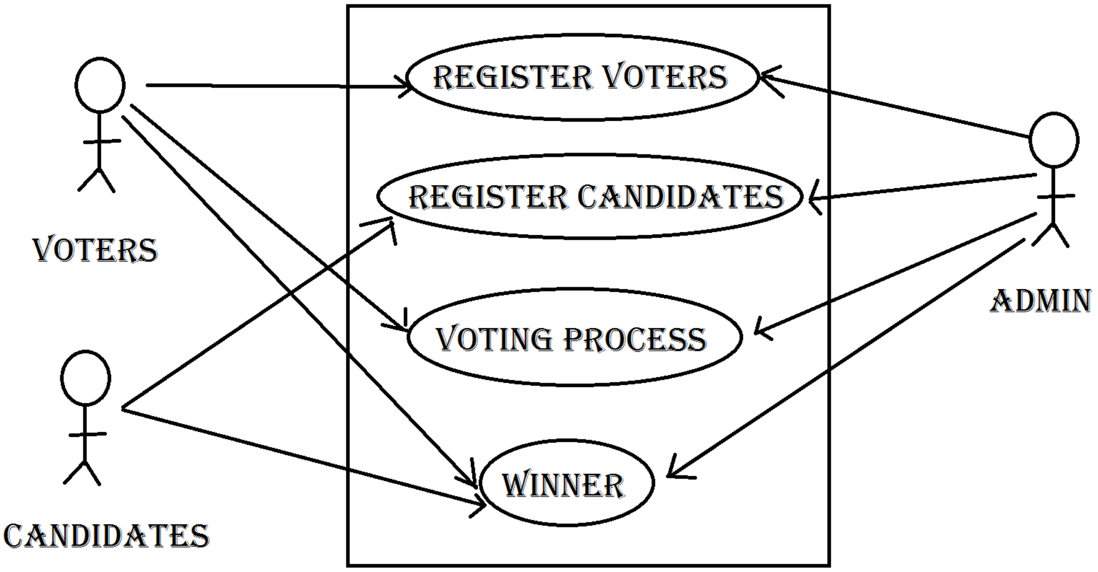
28.USE CASE DIAGRAM :

A use case diagram is a graphic depiction of the interactions among the elements of a system.

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems.

System objectives can include planning overall requirements, validating a hardware design, testing and debugging a software product under development, creating an online help reference, or performing a consumer-service-oriented task. For example, use cases in a product sales environment would include item ordering, catalog updating, payment processing, and customer relations. A use case diagram contains four components.

- The boundary, which defines the system of interest in relation to the world around it.
- The actors, usually individuals involved with the system defined according to their roles.
- The use cases, which are the specific roles played by the actors within and around the system.
- The relationships between and among the actors and the use cases.



29.CLASS DIAGRAM :

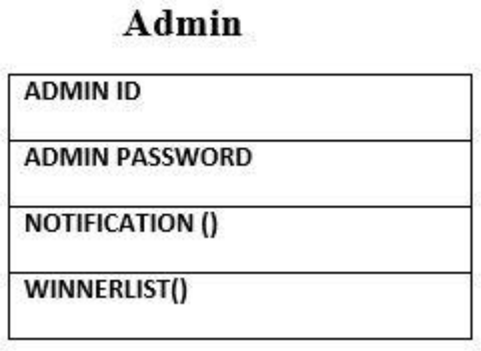
In software engineering, a **class diagram** in the Unified Modeling Language (UML) is a type of static structure **diagram** that describes the structure of a system by showing the system's **classes**, their attributes, operations (or methods), and the relationships among objects.

Voters

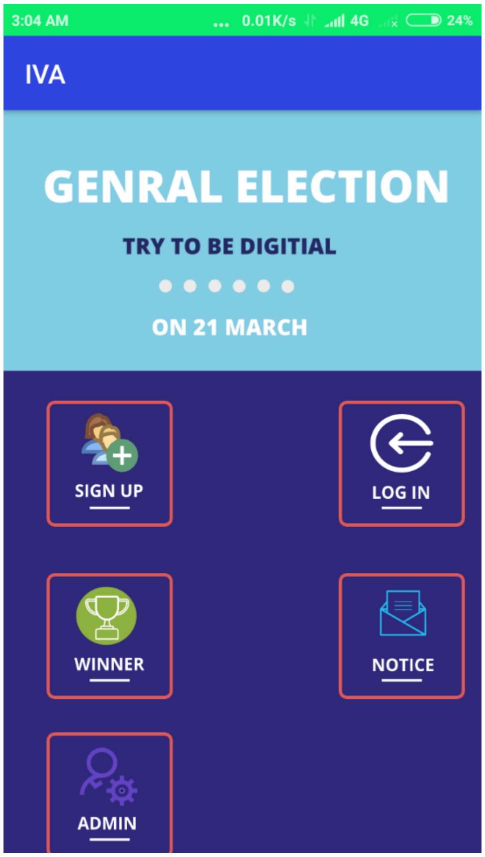
VOTER NAME
VOTER ID
VOTER PH. NO.
VOTER PIN
VOTER PASSWORD
MEMBERSHIP DETAILS ()
ISVOTED()
NOTIFICATION()
WINNERLIST()

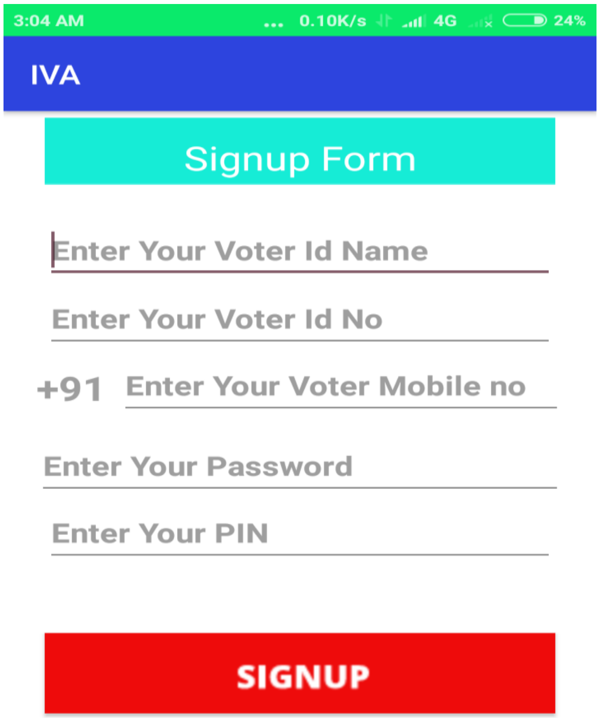
Candidates

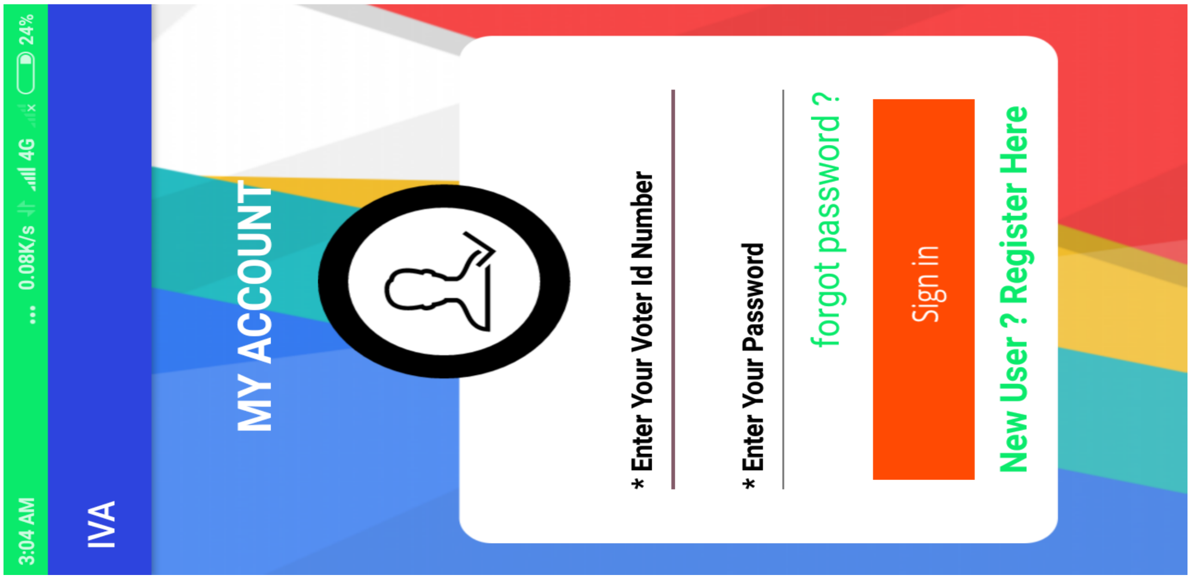
CANDIDATE NAME
CANDIDATE ID
CANDIDATE PARTY
CANDIDATE PIN
CANDIDATE VOTE
NOTIFICATION()

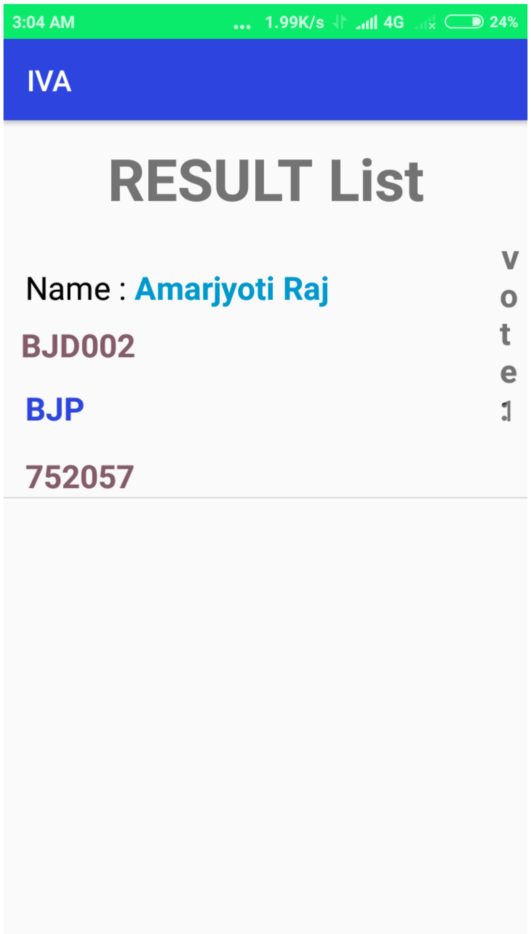

Admin
ADMIN ID
ADMIN PASSWORD
NOTIFICATION ()
WINNERLIST()

30.Screen Shots









IVA

RESULT List

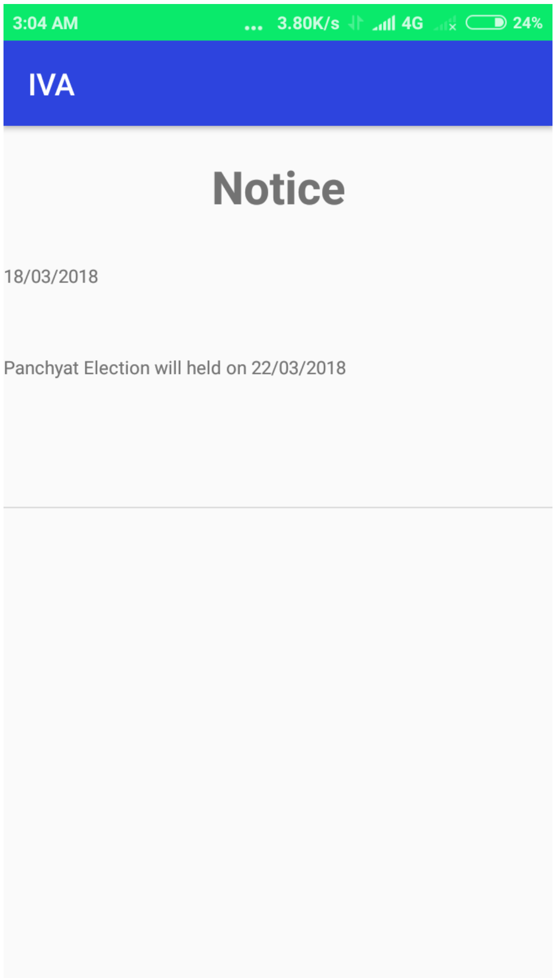
Name : Amarjyoti Raj

BJD002

BJP

752057

vote 1

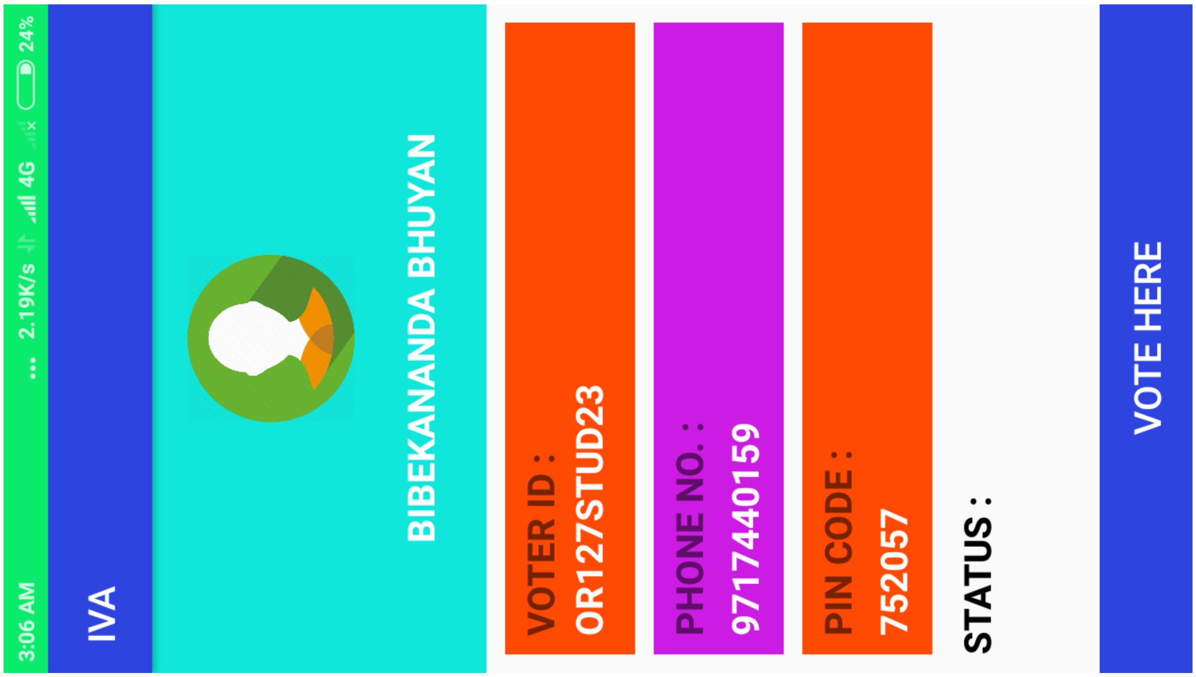


IVA

Notice

18/03/2018

Panchyat Election will held on 22/03/2018



IVA

BIBEKANANDA BHUYAN

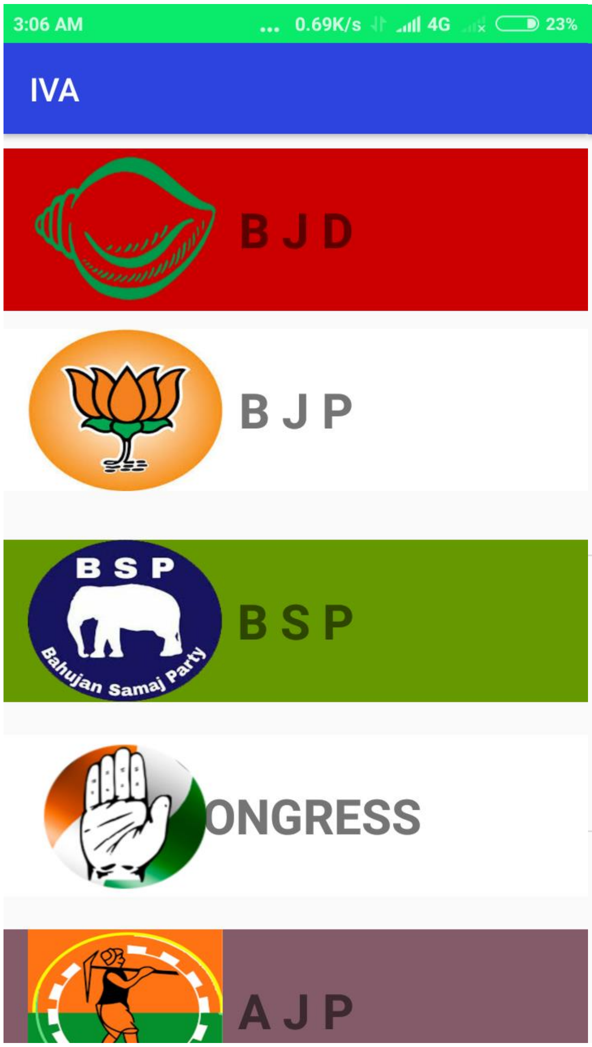
VOTER ID : OR127STUD23

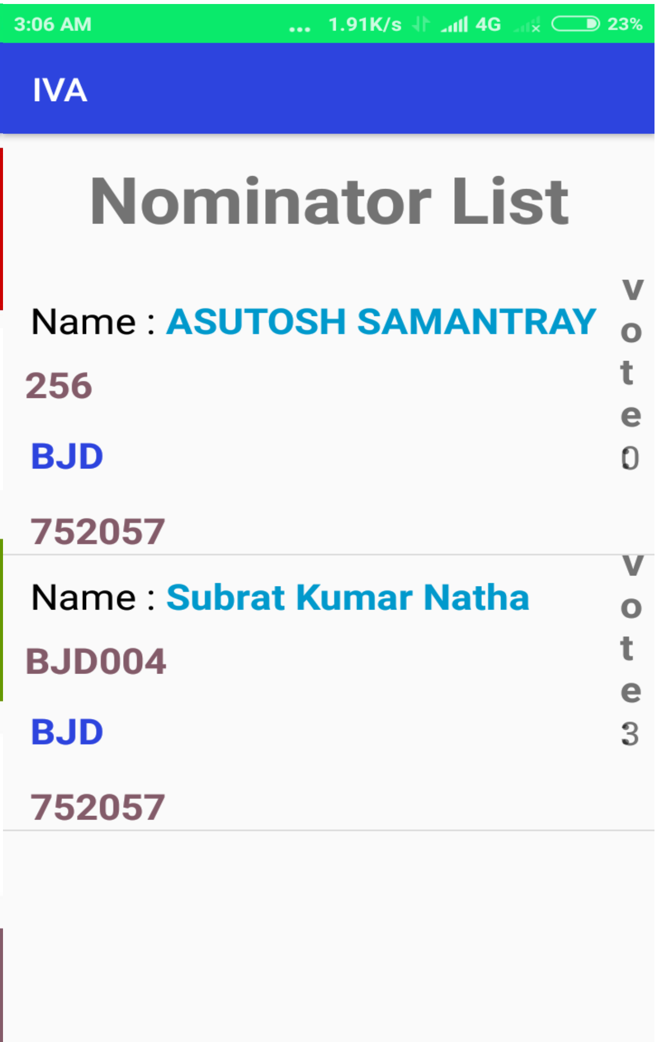
PHONE NO. : 9717440159

PIN CODE : 752057

STATUS :

VOTE HERE





Nominator List

Name : ASUTOSH SAMANTRAY

256

BJD

752057

vote 0

Name : Subrat Kumar Natha

BJD004

BJD

752057

vote 3

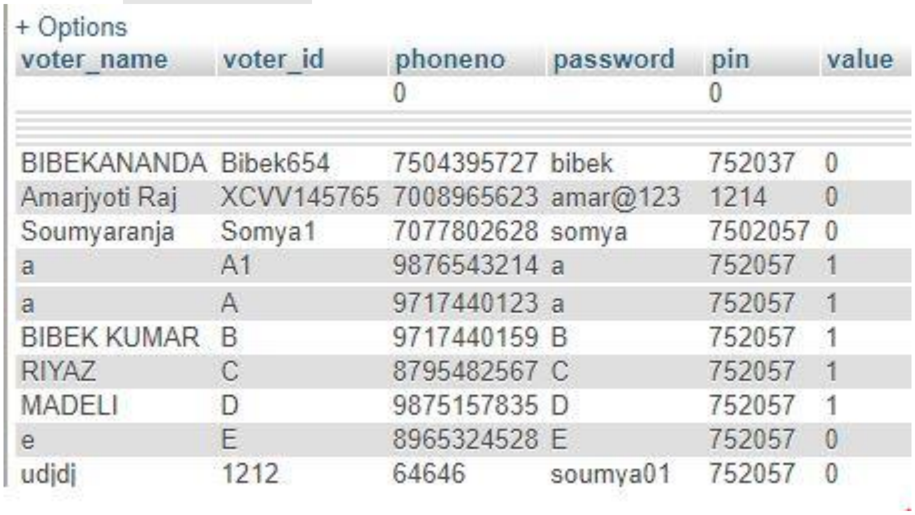
31.Data Base Design

Data Dictionary:

A **data dictionary** is a collection of descriptions of the **data** objects or items in a**data** model for the benefit of programmers and others who need to refer to them. A first step in analyzing a system of objects with which users interact is to identify each object and its relationship to other objects.

Table 1:

Name : `` `voterlist` ``

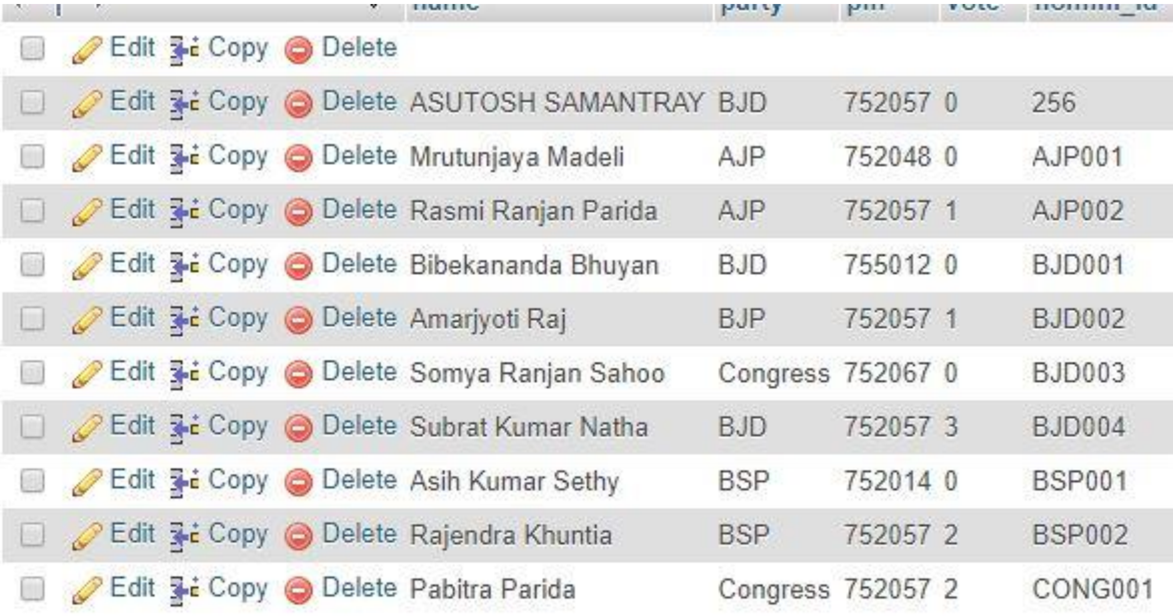


+ Options

voter_name	voter_id	phoneno	password	pin	value
		0		0	
BIBEKANANDA	Bibek654	7504395727	bibek	752037	0
Amarjyoti Raj	XCVV145765	7008965623	amar@123	1214	0
Soumyaranja	Somya1	7077802628	somya	7502057	0
a	A1	9876543214	a	752057	1
a	A	9717440123	a	752057	1
BIBEK KUMAR	B	9717440159	B	752057	1
RIYAZ	C	8795482567	C	752057	1
MADELI	D	9875157835	D	752057	1
e	E	8965324528	E	752057	0
udjdj	1212	64646	soumya01	752057	0

Table 2:

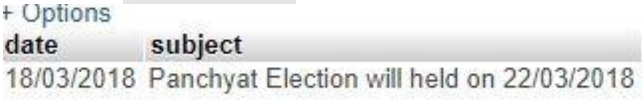
Name : `` ` `nomini_list` ``



	name	party	pin	vote	nomini_id
□ Edit Copy Delete					
□ Edit Copy Delete	ASUTOSH SAMANTRAY	BJD	752057	0	256
□ Edit Copy Delete	Mrutunjaya Madeli	AJP	752048	0	AJP001
□ Edit Copy Delete	Rasmi Ranjan Parida	AJP	752057	1	AJP002
□ Edit Copy Delete	Bibekananda Bhuyan	BJD	755012	0	BJD001
□ Edit Copy Delete	Amarjyoti Raj	BJP	752057	1	BJD002
□ Edit Copy Delete	Somya Ranjan Sahoo	Congress	752067	0	BJD003
□ Edit Copy Delete	Subrat Kumar Natha	BJD	752057	3	BJD004
□ Edit Copy Delete	Asih Kumar Sethy	BSP	752014	0	BSP001
□ Edit Copy Delete	Rajendra Khuntia	BSP	752057	2	BSP002
□ Edit Copy Delete	Pabitra Parida	Congress	752057	2	CONG001

Table 3:

Name : ` `notice`



+ Options

date	subject
18/03/2018	Panchyat Election will held on 22/03/2018

32.Xml Code:

Admin Module:

Activity_adminlogin.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/appb"
    tools:context="com.example.bibeka.iva.adminlogin">
    <ImageView
        android:layout_width="match_parent"
        android:layout_height="400dp"
        android:layout_marginTop="90dp"
        android:layout_marginStart="20dp"
        android:layout_marginEnd="20dp"
        android:background="@drawable/loginform"/>
    <EditText
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:layout_marginTop="240dp"
        android:layout_marginStart="50dp"
        android:layout_marginEnd="50dp"
        android:hint="Enter Admin Id"
        android:id="@+id/iddrop"
        android:textColor="@android:color/background_dark"
        android:textColorHint="@android:color/background_dark"
        android:textStyle="bold" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:layout_marginTop="300dp"
        android:layout_marginStart="50dp"
        android:layout_marginEnd="50dp"
        android:id="@+id/password"
        android:hint="Enter Password"
        android:textColor="@android:color/background_dark"
        android:textColorHint="@android:color/background_dark"
        android:textStyle="bold" />

    <Button
        android:layout_width="match_parent"
        android:onClick="loginadmin"
        android:layout_height="55dp"
        android:layout_marginLeft="50dp"
        android:layout_marginRight="50dp"
        android:layout_marginTop="390dp"
        android:layout_marginEnd="60dp"
        android:layout_marginStart="60dp"
        android:background="@drawable/signin"
        android:gravity="center"
        android:textSize="30dp" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginEnd="55dp"
        android:layout_marginRight="20dp"
        android:layout_marginStart="40dp"
        android:layout_marginTop="30dp"
        android:gravity="center"
        android:text="ADMIN LOGIN"
        android:textColor="@android:color/background_light"
        android:textSize="25dp"
        android:textStyle="bold" />

</RelativeLayout>
```

Activity_newcandidates.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.newcandicates">

    tools:context="com.example.bibeka.iva.Login">
    <RelativeLayout
        android:layout_width="match_parent"
        android:background="@drawable/signuptop"
        android:layout_marginTop="5dp"
        android:layout_marginLeft="25dp"
        android:layout_marginRight="25dp"
        android:layout_height="50dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginTop="10dp"
            android:gravity="center"
            android:text="Register "
            android:textColor="@android:color/background_light"
            android:textSize="25dp" />
    </RelativeLayout>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="60dp">

        <EditText
            android:id="@+id/vname"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="25dp"
            android:layout_marginRight="25dp"
            android:layout_marginTop="20dp"
            android:hint="Enter Candidate Name "
            android:textColor="@android:color/holo_blue_bright"
            android:textSize="20dp"
            android:textStyle="bold" />
        <EditText
            android:id="@+id/vtno"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_marginLeft="25dp"
            android:layout_marginRight="25dp"
            android:layout_marginTop="70dp"
            android:hint="Enter Candidate Party Name "
            android:inputType="textCapWords"
            android:textSize="20dp"
            android:textStyle="bold" />

        <EditText
            android:id="@+id/vphno"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_marginLeft="25dp"
            android:layout_marginRight="25dp"
            android:layout_marginTop="120dp"
            android:hint="Enter Candidate Id "
            android:inputType="number"
            android:textSize="20dp"
            android:textStyle="bold" />


        <EditText
            android:id="@+id/vpin"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_marginLeft="25dp"
            android:layout_marginRight="25dp"
            android:layout_marginTop="180dp"
            android:hint="Enter Your PIN "
            android:inputType="number"
            android:textSize="20dp"
            android:textStyle="bold" />
        <Button
            android:layout_width="match_parent"
            android:layout_marginLeft="25dp"
            android:layout_marginRight="25dp"
            android:layout_height="60dp"
            android:layout_marginTop="250dp"
            android:gravity="center"
            android:text="register"
            android:textStyle="bold"
            android:onClick="signup" />



    </RelativeLayout>

</RelativeLayout>
```

Activity_firstactivity.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/res/android">
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.firstactivity">

    <RelativeLayout
        android:layout_width="match_parent"
        android:background="@drawable/election"
        android:layout_height="210dp"
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true">
    </RelativeLayout>

        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginTop="210dp"
            android:background="@drawable/fback">


            <ImageView
                android:id="@+id/imageView"
                android:layout_width="100dp"
                android:layout_height="100dp"
                android:layout_marginLeft="30dp"
                android:background="@drawable/signup"
                android:onClick="adduser"
               android:layout_marginTop="20dp"/>
            <ImageView
                android:id="@+id/imageView3"
                android:layout_width="100dp"
                android:layout_height="100dp"
                android:layout_marginTop="20dp"
                android:layout_marginLeft="250dp"
                android:background="@drawable/newlogin"
                android:onClick="loginuser"/>

            <ImageView
                android:id="@+id/imageView4"
                android:layout_width="100dp"
                android:layout_height="100dp"
                android:layout_marginTop="150dp"
                android:layout_marginLeft="30dp"
                android:background="@drawable/newwiner"
                android:onClick="Winner"
                />
            <ImageView
                android:id="@+id/imageView5"
                android:layout_width="100dp"
                android:layout_height="100dp"
                android:background="@drawable/newnotice"
                android:layout_marginTop="150dp"
                android:layout_marginLeft="250dp"
                android:onClick="notice" />
            <ImageView
                android:id="@+id/imageView6"
                android:layout_width="100dp"
                android:layout_height="100dp"
                android:background="@drawable/newadmin"
                android:layout_marginTop="270dp"
                android:onClick="admin"
                android:layout_marginLeft="30dp"/>


        </RelativeLayout>


</RelativeLayout>
</ScrollView>
```

User Module:

Activity_login.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/appb"
    tools:context="com.example.bibeka.iva.newuser">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="400dp"
        android:layout_marginTop="90dp"
        android:layout_marginStart="20dp"
        android:layout_marginEnd="20dp"
        android:background="@drawable/loginform"/>
    <EditText
        android:layout_width="match_parent"
        android:id="@+id/vid"
        android:layout_height="50dp"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:layout_marginTop="240dp"
        android:layout_marginStart="50dp"
        android:layout_marginEnd="50dp"
        android:hint="* Enter Your Voter Id Number"
        android:textColor="@android:color/background_dark"
        android:textColorHint="@android:color/background_dark"
        android:textStyle="bold" />
    <EditText
        android:layout_width="match_parent"
        android:id="@+id/vpass"
        android:layout_height="50dp"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:layout_marginTop="300dp"
        android:layout_marginStart="50dp"
        android:layout_marginEnd="50dp"
        android:hint="* Enter Your Password"
        android:textColor="@android:color/background_dark"
        android:textColorHint="@android:color/background_dark"
        android:textStyle="bold" />
    <TextView
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_marginEnd="55dp"
        android:layout_marginRight="20dp"
        android:layout_marginTop="350dp"
        android:gravity="end"
        android:text="forgot password ?"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="20dp" />

    <Button
        android:layout_width="match_parent"
        android:onClick="login"
        android:layout_height="55dp"
        android:layout_marginLeft="50dp"
        android:layout_marginRight="50dp"
        android:layout_marginTop="390dp"
        android:layout_marginEnd="60dp"
        android:layout_marginStart="60dp"
       android:background="@drawable/signin"
        android:gravity="center"
        android:textSize="30dp" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_marginTop="450dp"
        android:layout_marginStart="60dp"
        android:layout_marginEnd="60dp"
        android:gravity="center"
        android:text="New User ? Register Here"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="20dp"
        android:textStyle="bold"
        android:layout_marginRight="20dp"/>


    <TextView
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginEnd="55dp"
        android:layout_marginRight="20dp"
        android:layout_marginStart="40dp"
        android:layout_marginTop="30dp"
        android:gravity="center"
        android:text="MY ACCOUNT "
        android:textColor="@android:color/background_light"
        android:textSize="25dp"
        android:textStyle="bold" />
</RelativeLayout>
```

Activity_afterlogin.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
```

```
tools:context="com.example.bibeka.iva.faterlogin">

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="180dp"
    android:background="@drawable/profileback">
    <ImageView
        android:layout_width="90dp"
        android:layout_height="90dp"
        android:layout_marginTop="19dp"
        android:background="@drawable/userlogo"
        android:foregroundGravity="center"
        android:layout_marginLeft="135dp"
        android:id="@+id/imageView2" />

    <TextView
        android:id="@+id/namedrop"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_alignParentEnd="true"
        android:layout_alignTop="@+id/imageView2"
        android:layout_marginTop="100dp"
        android:gravity="center"
        android:text="Name"
        android:textColor="@android:color/background_light"
        android:textSize="20dp"
        android:textStyle="bold" />
</RelativeLayout>

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_marginStart="10dp"
    android:layout_marginEnd="10dp"
    android:layout_marginTop="190dp"
    android:background="@drawable/design"
    android:layout_height="70dp">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_marginTop="5dp"
        android:layout_marginStart="10dp"
        android:text="VOTER ID :"
        android:textSize="20dp"
        android:textStyle="bold"
        android:layout_alignParentEnd="true"
        android:layout_marginBottom="27dp" />

    <TextView
        android:id="@+id/iddrop"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_alignParentBottom="true"
        android:layout_alignParentStart="true"
        android:layout_marginBottom="9dp"
        android:layout_marginStart="10dp"
        android:text=""
        android:textColor="@android:color/background_light"
        android:textSize="20dp"
        android:textStyle="bold" />
</RelativeLayout>


<RelativeLayout
```

```
        android:layout_width="match_parent"
        android:layout_marginStart="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginTop="270dp"
        android:background="@drawable/designa"
        android:layout_height="70dp">

        <TextView
            android:id="@+id/phonedrop"
            android:layout_width="match_parent"
            android:layout_height="30dp"
            android:layout_marginStart="10dp"
            android:layout_marginTop="35dp"
            android:text=""
            android:textColor="@android:color/background_light"
            android:textSize="20dp"
            android:textStyle="bold" />
        <TextView
            android:layout_width="match_parent"
            android:layout_height="30dp"
            android:text="PHONE NO.  :"
            android:layout_marginStart="10dp"
            android:textSize="20dp"
            android:textStyle="bold"
            android:layout_marginTop="5dp"/>

    </RelativeLayout>



    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_marginStart="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginTop="350dp"
        android:background="@drawable/design"
        android:layout_height="70dp">

        <TextView
            android:id="@+id/pindrop"
            android:layout_width="match_parent"
            android:layout_height="40dp"
            android:layout_marginStart="10dp"
            android:layout_marginTop="35dp"
            android:text=""
            android:textColor="@android:color/background_light"
            android:textSize="20dp"
            android:textStyle="bold" />
        <TextView
            android:layout_width="match_parent"
            android:layout_height="40dp"
            android:layout_marginTop="5dp"
            android:layout_marginStart="10dp"
            android:text="PIN CODE :"
            android:textSize="20dp"
            android:textStyle="bold"/>

    </RelativeLayout>


    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_alignParentBottom="true"
        android:layout_alignParentStart="true">

        <Button
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="?attr/colorPrimary"
            android:onClick="vote"
            android:text="VOTE HERE"
            android:textColor="@android:color/background_light"
            android:textSize="20dp" />
    </RelativeLayout>

    <TextView
        android:layout_width="100dp"
        android:layout_height="40dp"
        android:layout_marginStart="10dp"
        android:layout_marginTop="430dp"
        android:text="STATUS :"
        android:textColor="@android:color/black"
        android:textSize="20dp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/votedrop"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_marginStart="120dp"
        android:layout_marginTop="430dp"
        android:text=""
        android:textColor="?attr/colorPrimary"
        android:textSize="20dp"
        android:textStyle="bold" />

</RelativeLayout>
```

Activity_voterlogolist.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/res/android">
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.voterlogolist">


    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginTop="9dp"
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@android:color/holo_red_dark"
            android:text="B J D"
            android:gravity="center"
            android:textStyle="bold"
            android:onClick="bjd"
            android:textSize="30dp"/>
        <ImageView
            android:layout_width="120dp"
            android:layout_height="match_parent"
            android:background="@drawable/bjd"
            android:layout_marginLeft="15dp"/>
    </RelativeLayout>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginTop="120dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@android:color/background_light"
            android:text="B J P"
            android:gravity="center"
            android:textStyle="bold"
            android:onClick="bjp"
            android:textSize="30dp"/>
        <ImageView
            android:layout_width="120dp"
            android:layout_height="match_parent"
            android:background="@drawable/bjp"
            android:layout_marginLeft="15dp"/>
    </RelativeLayout>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginTop="250dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@android:color/holo_green_dark"
            android:text="B S P"
            android:gravity="center"
            android:textStyle="bold"
            android:onClick="bsp"
            android:textSize="30dp"/>
        <ImageView
            android:layout_width="120dp"
            android:layout_height="match_parent"
            android:background="@drawable/bahujansamaj"
            android:layout_marginLeft="15dp"/>
    </RelativeLayout>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginTop="370dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@android:color/background_light"
            android:text="CONGRESS"
            android:gravity="center"
            android:onClick="congress"
            android:textStyle="bold"
            android:textSize="30dp"/>
        <ImageView
            android:layout_width="120dp"
            android:layout_height="match_parent"
       android:background="@drawable/congresslogo"
            android:layout_marginLeft="15dp"/>
    </RelativeLayout>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginTop="490dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@color/colorAccent"
            android:text="A J P"
            android:gravity="center"
            android:onClick="ajp"
            android:textStyle="bold"
            android:textSize="30dp"
            android:layout_alignParentTop="true"
            android:layout_alignParentStart="true" />
        <ImageView
            android:layout_width="120dp"
            android:layout_height="match_parent"
            android:background="@drawable/amjanataparty"
            android:layout_marginLeft="15dp"/>
    </RelativeLayout>


</RelativeLayout>
</ScrollView>
```

Activity_nominatorlist.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.nominatorlist">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:text="Nominator List"
        android:textSize="40dp"
        android:gravity="center"
        android:textStyle="bold"/>
    <ListView
        android:layout_width="fill_parent"
        android:layout_height="match_parent"
        android:id="@+id/listView1"
        android:layout_marginTop="90dp"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">
    </ListView>

    <ProgressBar
        android:id="@+id/ProgressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:visibility="gone"/>

</RelativeLayout>
```

Activity_congressnominatorlist.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.congressnominatorlist">


    <TextView
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:text="Nominator List"
        android:textSize="40dp"
        android:gravity="center"
        android:textStyle="bold"/>
    <ListView
        android:layout_width="fill_parent"
        android:layout_height="match_parent"
        android:id="@+id/listView1"
        android:layout_marginTop="90dp"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">

    </ListView>

    <ProgressBar
        android:id="@+id/ProgressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:visibility="gone"/>

</RelativeLayout>
```

Activity_bjpnominatorlist.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.bjpnominatorlist">


    <TextView
```

```
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:text="Nominator List"
        android:textSize="40dp"
        android:gravity="center"
        android:textStyle="bold"/>
    <ListView
        android:layout_width="fill_parent"
        android:layout_height="match_parent"
        android:id="@+id/listView1"
        android:layout_marginTop="90dp"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">

    </ListView>

    <ProgressBar
        android:id="@+id/ProgressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:visibility="gone"/>


</RelativeLayout>
```

Activity_bahujantanominatorlist.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.bahujansamajnominatorlist">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:text="Nominator List"
        android:textSize="40dp"
        android:gravity="center"
        android:textStyle="bold"/>
    <ListView
        android:layout_width="fill_parent"
        android:layout_height="match_parent"
        android:id="@+id/listView1"
        android:layout_marginTop="90dp"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">

    </ListView>

    <ProgressBar
        android:id="@+id/ProgressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:visibility="gone"/>


</RelativeLayout>
```

Activity_amjanatapartynominatorlist.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.amjantapartynominatorlist">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:text="Nominator List"
        android:textSize="40dp"
        android:gravity="center"
        android:textStyle="bold"/>
    <ListView
        android:layout_width="fill_parent"
        android:layout_height="match_parent"
        android:id="@+id/listView1"
        android:layout_marginTop="90dp"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">

    </ListView>

    <ProgressBar
        android:id="@+id/ProgressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:visibility="gone"/>


</RelativeLayout>
```

Activity_vote.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.vote">


    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="180dp"
        android:background="@drawable/profileback">
        <ImageView
            android:layout_width="90dp"
            android:layout_height="90dp"
            android:layout_marginTop="19dp"
            android:background="@drawable/userlogo"
            android:foregroundGravity="center"
            android:layout_marginLeft="135dp"
            android:id="@+id/imageView2" />

        <TextView
            android:id="@+id/namedrop"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_alignParentEnd="true"
            android:layout_alignTop="@+id/imageView2"
            android:layout_marginTop="100dp"
            android:gravity="center"
            android:text="Name"
            android:textColor="@android:color/background_light"
            android:textSize="20dp"
            android:textStyle="bold" />
    </RelativeLayout>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_marginStart="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginTop="190dp"
        android:background="@drawable/design"
        android:layout_height="70dp">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="30dp"
            android:layout_marginTop="5dp"
            android:layout_marginStart="10dp"
            android:text="PARTY ID :"
            android:textSize="20dp"
            android:textStyle="bold"
            android:layout_alignParentEnd="true"
            android:layout_marginBottom="27dp" />

        <TextView
            android:id="@+id/iddrop"
            android:layout_width="match_parent"
            android:layout_height="30dp"
            android:layout_alignParentBottom="true"
            android:layout_alignParentStart="true"
            android:layout_marginBottom="9dp"
            android:layout_marginStart="10dp"
            android:text=""
            android:textColor="@android:color/background_light"
            android:textSize="20dp"
            android:textStyle="bold" />
    </RelativeLayout>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_marginStart="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginTop="270dp"
        android:background="@drawable/designa"
        android:layout_height="70dp">

        <TextView
            android:id="@+id/phonedrop"
            android:layout_width="match_parent"
            android:layout_height="30dp"
            android:layout_marginStart="10dp"
            android:layout_marginTop="35dp"
            android:text=""
            android:textColor="@android:color/background_light"
            android:textSize="20dp"
            android:textStyle="bold" />
        <TextView
```

```
            android:layout_width="match_parent"
            android:layout_height="30dp"
            android:text="PARTY :"
            android:layout_marginStart="10dp"
            android:textSize="20dp"
            android:textStyle="bold"
            android:layout_marginTop="5dp"/>

    </RelativeLayout>
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_marginStart="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginTop="350dp"
        android:background="@drawable/design"
        android:layout_height="70dp">

        <TextView
            android:id="@+id/pindrop"
            android:layout_width="match_parent"
            android:layout_height="40dp"
            android:layout_marginStart="10dp"
            android:layout_marginTop="35dp"
            android:text=""
            android:textColor="@android:color/background_light"
            android:textSize="20dp"
            android:textStyle="bold" />
        <TextView
            android:layout_width="match_parent"
            android:layout_height="40dp"
            android:layout_marginTop="5dp"
            android:layout_marginStart="10dp"
            android:text="PIN CODE :"
            android:textSize="20dp"
            android:textStyle="bold"/>

    </RelativeLayout>



    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_alignParentBottom="true"
        android:layout_alignParentStart="true">

        <Button
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@color/colorPrimaryDark"
            android:onClick="vote"
            android:text="VOTE HERE"
            android:textSize="20dp" />
    </RelativeLayout>
    <TextView
        android:layout_width="100dp"
        android:layout_height="40dp"
        android:layout_marginStart="10dp"
        android:layout_marginTop="430dp"
        android:text="VOTE :"
        android:textColor="@android:color/black"
        android:textSize="20dp"
        android:textStyle="bold" />
    <TextView
        android:id="@+id/votedrop"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:layout_marginStart="120dp"
        android:layout_marginTop="430dp"
        android:text=""
        android:textColor="?attr/colorPrimary"
        android:textSize="20dp"
        android:textStyle="bold" />
</RelativeLayout>
```

WINER MODULE :

Activity_winerlist.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.winerlist">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:text="RESULT List"
        android:textSize="40dp"
        android:gravity="center"
        android:textStyle="bold"/>
    <ListView
        android:layout_width="fill_parent"
        android:layout_height="match_parent"
        android:id="@+id/listView1"
        android:layout_marginTop="90dp"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">

    </ListView>

    <ProgressBar
        android:id="@+id/ProgressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:visibility="gone"/>

</RelativeLayout>
```

NOTICE MODULE :

Activity_shownotice.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.bibeka.iva.shownotice">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:text="Notice"
        android:textSize="30dp"
        android:gravity="center"
        android:textStyle="bold"/>
    <ListView
        android:layout_width="fill_parent"
        android:layout_height="match_parent"
        android:id="@+id/listView1"
        android:layout_marginTop="90dp"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">

    </ListView>

    <ProgressBar
        android:id="@+id/ProgressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:visibility="gone"/>


</RelativeLayout>
```

LISTVIEW XML FILE REQURED :

listvalue.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/text1"
        android:text="Name  : "
        android:layout_marginLeft="15dp"
        android:textSize="22dp"
        android:textColor="#000"
        android:layout_marginTop="10dp"
        />

    <TextView
        android:id="@+id/textviewID"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_marginTop="10dp"
        android:layout_toEndOf="@+id/text1"
        android:layout_toRightOf="@+id/text1"
        android:text="Name Show Here"
        android:textColor="@android:color/holo_blue_dark"
        android:textSize="22dp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/textviewSubjectName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textviewID"
        android:layout_marginLeft="12dp"
        android:layout_marginTop="10dp"
        android:text=" Id Show Here"
        android:textColor="@color/colorAccent"
        android:textSize="22dp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_below="@+id/textviewSubjectName"
        android:layout_marginEnd="8dp"
        android:layout_marginStart="15dp"
        android:layout_marginTop="14dp"
        android:text="Party Show Here"
        android:textColor="@color/colorPrimary"
        android:textSize="22dp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Vote Show Here"
        android:textSize="20dp"
        android:gravity="center"
        android:layout_alignBottom="@+id/textView"
        android:layout_alignEnd="@+id/textView" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignEnd="@+id/textView2"
        android:layout_alignParentStart="true"
        android:layout_below="@+id/textView2"
        android:layout_marginStart="15dp"
        android:layout_marginTop="17dp"
        android:text="Pincode Show Here"
        android:textColor="@color/colorAccent"
        android:textSize="22dp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/textView7"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/textView3"
        android:layout_alignStart="@+id/textView2"
        android:text="Vote :"
        android:textStyle="bold"
        android:textSize="22dp"/>

</RelativeLayout>
```

noticeview.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:id="@+id/datedrop"/>
    <TextView
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginTop="60dp"
        android:id="@+id/subject"/>

</RelativeLayout>
```

33.JAVA FILE REQURED :

Adminlogin.java:

```
package com.example.bibeka.iva;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class adminlogin extends AppCompatActivity {
    EditText E1,E2;
    String adminid,password;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_adminlogin);

         E1=(EditText)findViewById(R.id.iddrop);
        E2=(EditText)findViewById(R.id.password);

    }

    public void loginadmin(View view) {
        adminid=E1.getText().toString().trim();
        password=E2.getText().toString().trim();
        if (adminid.equals("govadmin") || password.equals("govadmin"))
        {
            Intent a = new Intent(adminlogin.this,newcandicates.class);
            startActivity(a);
        }
        else
        {
            Toast.makeText(this, "Entered Admin Id Or Password Is Wrong Please
Renter", Toast.LENGTH_LONG).show();
        }

    }
}
```

amjanatapartynominatorlist.java :

```
package com.example.bibeka.iva;

import android.content.Context;
import android.content.Intent;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.List;

public class amjantapartynominatorlist extends AppCompatActivity {
    ListView listView;
    String Rg;
    Button button;


    // Server Http URL
  String HTTP_URL = "https://onlyforyou446.000webhostapp.com/ajp.php?Rg=";

    // String to hold complete JSON response object.
    String FinalJSonObject ;
    String Name1,id1,pin1,party1,vote1;

    String unamedataFromIntent,uiddataFromIntent,upassworddataFromIntent,uphonenodataFromIntent,uvaluedataFromIntent;
    String dataFromIntent;
    ProgressBar progressBar;
    List<Subject> CustomSubjectNamesList;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_amjantapartynominatorlist);
        listView = (ListView) findViewById(R.id.listView1);
        progressBar = (ProgressBar)findViewById(R.id.ProgressBar1);
```

```
         dataFromIntent = getIntent().getStringExtra("pin");
         Rg=dataFromIntent;

        unamedataFromIntent = getIntent().getStringExtra("name");
        uiddataFromIntent = getIntent().getStringExtra("id");
        uvaluedataFromIntent = getIntent().getStringExtra("value");
        upassworddataFromIntent = getIntent().getStringExtra("password");
        uphonenodataFromIntent = getIntent().getStringExtra("phoneno");


        //Showing progress bar just after button click.
        progressBar.setVisibility(View.VISIBLE);

        // Creating StringRequest and set the JSON server URL in here.
        StringRequest stringRequest = new StringRequest(HTTP_URL+Rg,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {

                        // After done Loading store JSON response in FinalJSonObject
string variable.
                        FinalJSonObject = response ;

                        // Calling method to parse JSON object.
                        new
ParseJSonDataClass(amjantapartynominatorlist.this).execute();

                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {

                        // Showing error message if something goes wrong.

Toast.makeText(amjantapartynominatorlist.this,error.getMessage(),Toast.LENGTH_LONG).sh
ow();

                    }
                });

        // Creating String Request Object.
        RequestQueue requestQueue =
Volley.newRequestQueue(amjantapartynominatorlist.this);

        // Passing String request into RequestQueue.
        requestQueue.add(stringRequest);

        listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position,
long id) {
                ViewItem view1;
                view1 = new ViewItem();
                Subject sub;
                sub= new Subject();
                ListViewAdapter list1;
                // Name1=sub.Subject_Name;
                Name1=CustomSubjectNamesList.get(position).Subject_Name;
                id1=CustomSubjectNamesList.get(position).Subject_ID;
                pin1=CustomSubjectNamesList.get(position).Subject_Pin;
                party1=CustomSubjectNamesList.get(position).Subject_Party;
                vote1=CustomSubjectNamesList.get(position).Subject_Vote;
                Intent a = new Intent(amjantapartynominatorlist.this,vote.class);
                a.putExtra("name1",Name1);
                a.putExtra("id1",id1);
                a.putExtra("pin1",pin1);
                a.putExtra("party1",party1);
                a.putExtra("vote1",vote1);


                //User Data Get Through Intent.
                a.putExtra("pin",Rg);
                a.putExtra("name",unamedataFromIntent);
                a.putExtra("id",uiddataFromIntent);
                a.putExtra("phoneno",uphonenodataFromIntent);
                a.putExtra("value",uvaluedataFromIntent);
                a.putExtra("password",upassworddataFromIntent);


                startActivity(a);
                // Toast.makeText(MainActivity.this,""+Name1,
Toast.LENGTH_SHORT).show();
            }
        });
    }


    private class ParseJSonDataClass extends AsyncTask<Void, Void, Void> {

        public Context context;

        // Creating List of Subject class.


        public ParseJSonDataClass(Context context) {

            this.context = context;
        }

        @Override
        protected void onPreExecute() {

            super.onPreExecute();
        }

        @Override
        protected Void doInBackground(Void... arg0) {

            try {

                // Checking whether FinalJSonObject is not equals to null.
                if (FinalJSonObject != null) {

                    // Creating and setting up JSON array as null.
                    JSONArray jsonArray = null;
                    try {

                        // Adding JSON response object into JSON array.
                        jsonArray = new JSONArray(FinalJSonObject);

                        // Creating JSON Object.
                        JSONObject jsonObject;

                        // Creating Subject class object.
                        Subject subject;
                        // Defining CustomSubjectNamesList AS Array List.
                        CustomSubjectNamesList = new ArrayList<Subject>();

                        for (int i = 0; i < jsonArray.length(); i++) {

                            subject = new Subject();

                            jsonObject = jsonArray.getJSONObject(i);

                            //Storing ID into subject list.
                            subject.Subject_ID = jsonObject.getString("nomini_id");

                            //Storing Subject name in subject list.
                            subject.Subject_Name = jsonObject.getString("name");
                            subject.Subject_Pin = jsonObject.getString("pin");
                            subject.Subject_Vote = jsonObject.getString("vote");
                            subject.Subject_Party = jsonObject.getString("party");


                            // Adding subject list object into CustomSubjectNamesList.
                            CustomSubjectNamesList.add(subject);
                        }
                    } catch (JSONException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                    }
                }
            } catch (Exception e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
            return null;
        }

        @Override
        protected void onPostExecute(Void result)

        {
            // After all done loading set complete CustomSubjectNamesList with
application context to ListView adapter.
            ListViewAdapter adapter = new ListViewAdapter(CustomSubjectNamesList,
context);

            // Setting up all data into ListView.
            listView.setAdapter(adapter);

            // Hiding progress bar after all JSON loading done.
            progressBar.setVisibility(View.GONE);

        }
    }

}
```

bahujansamajanominatorlist.java :

```
package com.example.bibeka.iva;

import android.content.Context;
import android.content.Intent;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.List;

public class bahujansamajnominatorlist extends AppCompatActivity {
    ListView listView;
    String Rg;

    Button button;

    // Server Http URL
     String HTTP_URL = "https://onlyforyou446.000webhostapp.com/bsp.php?Rg=";

    // String to hold complete JSON response object.
    String FinalJSonObject ;

    ProgressBar progressBar;
    String Name1,id1,pin1,party1,vote1;

    String
unamedataFromIntent,uiddataFromIntent,upassworddataFromIntent,uphonenodataFromIntent,uvaluedataFromIntent;
    String dataFromIntent;
    List<Subject> CustomSubjectNamesList;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_bahujansamajnominatorlist);
        listView = (ListView) findViewById(R.id.listView1);
        progressBar = (ProgressBar)findViewById(R.id.ProgressBar1);

         dataFromIntent = getIntent().getStringExtra("pin");
         Rg=dataFromIntent;
        unamedataFromIntent = getIntent().getStringExtra("name");
        uiddataFromIntent = getIntent().getStringExtra("id");
        uvaluedataFromIntent = getIntent().getStringExtra("value");
        upassworddataFromIntent = getIntent().getStringExtra("password");
        uphonenodataFromIntent = getIntent().getStringExtra("phoneno");



        //Showing progress bar just after button click.
        progressBar.setVisibility(View.VISIBLE);
        // Creating StringRequest and set the JSON server URL in here.
        StringRequest stringRequest = new StringRequest(HTTP_URL+Rg,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {

                        // After done Loading store JSON response in FinalJSonObject
string variable.
                        FinalJSonObject = response ;

                        // Calling method to parse JSON object.
                        new
ParseJSonDataClass(bahujansamajnominatorlist.this).execute();

                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {

                        // Showing error message if something goes wrong.

Toast.makeText(bahujansamajnominatorlist.this,error.getMessage(),Toast.LENGTH_LONG).sh
ow();

                    }
                });

        // Creating String Request Object.
        RequestQueue requestQueue =
Volley.newRequestQueue(bahujansamajnominatorlist.this);

        // Passing String request into RequestQueue.
        requestQueue.add(stringRequest);


        listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position,
long id) {
                ViewItem view1;
                view1 = new ViewItem();
                Subject sub;
                sub= new Subject();
                ListViewAdapter list1;
                // Name1=sub.Subject_Name;
                Name1=CustomSubjectNamesList.get(position).Subject_Name;
                id1=CustomSubjectNamesList.get(position).Subject_ID;
                pin1=CustomSubjectNamesList.get(position).Subject_Pin;
                party1=CustomSubjectNamesList.get(position).Subject_Party;
                vote1=CustomSubjectNamesList.get(position).Subject_Vote;
                Intent a = new Intent(bahujansamajnominatorlist.this,vote.class);
                a.putExtra("name1",Name1);
                a.putExtra("id1",id1);
                a.putExtra("pin1",pin1);
                a.putExtra("party1",party1);
                a.putExtra("vote1",vote1);


                //User Data Get Through Intent.
                a.putExtra("pin",Rg);
                a.putExtra("name",unamedataFromIntent);
                a.putExtra("id",uiddataFromIntent);
                a.putExtra("phoneno",uphonenodataFromIntent);
                a.putExtra("value",uvaluedataFromIntent);
                a.putExtra("password",upassworddataFromIntent);


                startActivity(a);
                // Toast.makeText(MainActivity.this,""+Name1,
Toast.LENGTH_SHORT).show();
            }
        });

    }


    private class ParseJSonDataClass extends AsyncTask<Void, Void, Void> {

        public Context context;

        // Creating List of Subject class.


        public ParseJSonDataClass(Context context) {

            this.context = context;
        }

        @Override
        protected void onPreExecute() {

            super.onPreExecute();
        }

        @Override
        protected Void doInBackground(Void... arg0) {

            try {

                // Checking whether FinalJSonObject is not equals to null.
                if (FinalJSonObject != null) {

                    // Creating and setting up JSON array as null.
                    JSONArray jsonArray = null;
                    try {

                        // Adding JSON response object into JSON array.
                        jsonArray = new JSONArray(FinalJSonObject);

                        // Creating JSON Object.
                        JSONObject jsonObject;

                        // Creating Subject class object.
                        Subject subject;

                        // Defining CustomSubjectNamesList AS Array List.
                        CustomSubjectNamesList = new ArrayList<Subject>();

                        for (int i = 0; i < jsonArray.length(); i++) {

                            subject = new Subject();

                            jsonObject = jsonArray.getJSONObject(i);

                            //Storing ID into subject list.
                            subject.Subject_ID = jsonObject.getString("nomini_id");

                            //Storing Subject name in subject list.
                            subject.Subject_Name = jsonObject.getString("name");
                            subject.Subject_Pin = jsonObject.getString("pin");
                            subject.Subject_Vote = jsonObject.getString("vote");
                            subject.Subject_Party = jsonObject.getString("party");


                            // Adding subject list object into CustomSubjectNamesList.
                            CustomSubjectNamesList.add(subject);
                        }
                    } catch (JSONException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                    }
                }
            } catch (Exception e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
            return null;
        }

        @Override
        protected void onPostExecute(Void result)

        {
            // After all done loading set complete CustomSubjectNamesList with
application context to ListView adapter.
            ListViewAdapter adapter = new ListViewAdapter(CustomSubjectNamesList,
context);

            // Setting up all data into ListView.
            listView.setAdapter(adapter);

            // Hiding progress bar after all JSON loading done.
            progressBar.setVisibility(View.GONE);
```

winerlist.java

```
package com.example.bibeka.iva;

import android.content.Context;
import android.os.AsyncTask;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;
import android.widget.ListView;
import android.widget.ProgressBar;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.List;

public class winerlist extends AppCompatActivity {
    ListView listView;
    String Rg;

    Button button;

    // Server Http URL
    String HTTP_URL =
"https://onlyforyou446.000webhostapp.com/bjpnominationlist.php?Rg=752057";
    String h= HTTP_URL+Rg;


    // String to hold complete JSON response object.
    String FinalJSonObject ;
    String
unamedataFromIntent,uiddataFromIntent,upassworddataFromIntent,uphonenodataFromIntent,u
valuedataFromIntent;
    String dataFromIntent;
    String Name1,id1,pin1,party1,vote1;
    List<Subject> CustomSubjectNamesList;
    ProgressBar progressBar;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_winerlist);
        listView = (ListView) findViewById(R.id.listView1);
        progressBar = (ProgressBar)findViewById(R.id.ProgressBar1);

       // dataFromIntent = getIntent().getStringExtra("pin");
       // Rg="752057";

       /* unamedataFromIntent = getIntent().getStringExtra("name");
        uiddataFromIntent = getIntent().getStringExtra("id");
        uvaluedataFromIntent = getIntent().getStringExtra("value");
        upassworddataFromIntent = getIntent().getStringExtra("password");
        uphonenodataFromIntent = getIntent().getStringExtra("phoneno");*/



        //Showing progress bar just after button click.
        progressBar.setVisibility(View.VISIBLE);

        // Creating StringRequest and set the JSON server URL in here.
        StringRequest stringRequest = new StringRequest(HTTP_URL,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {

                        // After done Loading store JSON response in FinalJSonObject
string variable.
                        FinalJSonObject = response ;

                        // Calling method to parse JSON object.
                        new winerlist.ParseJSonDataClass(winerlist.this).execute();

                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {

                        // Showing error message if something goes wrong.
                       //
Toast.makeText(bjpnominatorlist.this,error.getMessage(),Toast.LENGTH_LONG).show();

                    }
                });

        // Creating String Request Object.
        RequestQueue requestQueue = Volley.newRequestQueue(winerlist.this);

        // Passing String request into RequestQueue.
        requestQueue.add(stringRequest);


    }


    private class ParseJSonDataClass extends AsyncTask<Void, Void, Void> {

        public Context context;

        // Creating List of Subject class.


        public ParseJSonDataClass(Context context) {

            this.context = context;
        }

        @Override
        protected void onPreExecute() {

            super.onPreExecute();
        }

        @Override
        protected Void doInBackground(Void... arg0) {

            try {

                // Checking whether FinalJSonObject is not equals to null.
                if (FinalJSonObject != null) {

                    // Creating and setting up JSON array as null.
                    JSONArray jsonArray = null;
                    try {

                        // Adding JSON response object into JSON array.
                        jsonArray = new JSONArray(FinalJSonObject);

                        // Creating JSON Object.
                        JSONObject jsonObject;

                        // Creating Subject class object.
                        Subject subject;

                        // Defining CustomSubjectNamesList AS Array List.
                        CustomSubjectNamesList = new ArrayList<Subject>();
                        for (int i = 0; i < jsonArray.length(); i++) {

                            subject = new Subject();

                            jsonObject = jsonArray.getJSONObject(i);

                            //Storing ID into subject list.
                            subject.Subject_ID = jsonObject.getString("nomini_id");

                            //Storing Subject name in subject list.
                            subject.Subject_Name = jsonObject.getString("name");
                            subject.Subject_Pin = jsonObject.getString("pin");
                            subject.Subject_Vote = jsonObject.getString("vote");
                            subject.Subject_Party = jsonObject.getString("party");


                            // Adding subject list object into CustomSubjectNamesList.
                            CustomSubjectNamesList.add(subject);
                        }
                    } catch (JSONException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                    }
                }
            } catch (Exception e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
            return null;
        }

        @Override
        protected void onPostExecute(Void result)

        {
            // After all done loading set complete CustomSubjectNamesList with
application context to ListView adapter.
            ListViewAdapter adapter = new ListViewAdapter(CustomSubjectNamesList,
context);

            // Setting up all data into ListView.
            listView.setAdapter(adapter);

            // Hiding progress bar after all JSON loading done.
            progressBar.setVisibility(View.GONE);

        }
    }
}
```

voterlogolist.java

```
package com.example.bibeka.iva;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
```

```
public class voterlogolist extends AppCompatActivity {
    String Rg;
    String
unamedataFromIntent,uiddataFromIntent,upassworddataFromIntent,uphonenodataFromIntent,u
valuedataFromIntent;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_voterlogolist);
        String pindataFromIntent = getIntent().getStringExtra("pin");
        Rg=pindataFromIntent;
        unamedataFromIntent = getIntent().getStringExtra("name");
         uiddataFromIntent = getIntent().getStringExtra("id");
        uvaluedataFromIntent = getIntent().getStringExtra("value");
         upassworddataFromIntent = getIntent().getStringExtra("password");
         uphonenodataFromIntent = getIntent().getStringExtra("phoneno");

    }

    public void bjd(View view) {
        Intent i = new Intent(voterlogolist.this,nominatorlist.class);
        i.putExtra("pin",Rg);
        i.putExtra("name",unamedataFromIntent);
        i.putExtra("id",uiddataFromIntent);
        i.putExtra("phoneno",uphonenodataFromIntent);
        i.putExtra("value",uvaluedataFromIntent);
        i.putExtra("password",upassworddataFromIntent);
        startActivity(i);

    }

    public void bjp(View view) {
        Intent i = new Intent(voterlogolist.this,bjpnominatorlist.class);
        i.putExtra("pin",Rg);
        i.putExtra("name",unamedataFromIntent);
        i.putExtra("id",uiddataFromIntent);
        i.putExtra("phoneno",uphonenodataFromIntent);
        i.putExtra("value",uvaluedataFromIntent);
        i.putExtra("password",upassworddataFromIntent);
        startActivity(i);

    }

    public void bsp(View view) {
        Intent i = new Intent(voterlogolist.this,bahujansamajnominatorlist.class);
        i.putExtra("pin",Rg);
        i.putExtra("name",unamedataFromIntent);
        i.putExtra("id",uiddataFromIntent);
        i.putExtra("phoneno",uphonenodataFromIntent);
        i.putExtra("value",uvaluedataFromIntent);
        i.putExtra("password",upassworddataFromIntent);
        startActivity(i);


    }

    public void congress(View view) {
        Intent i = new Intent(voterlogolist.this,congressnominatorlist.class);
        i.putExtra("pin",Rg);
        i.putExtra("name",unamedataFromIntent);
        i.putExtra("id",uiddataFromIntent);
        i.putExtra("phoneno",uphonenodataFromIntent);
        i.putExtra("value",uvaluedataFromIntent);
        i.putExtra("password",upassworddataFromIntent);
        startActivity(i);

    }

    public void ajp(View view) {
        Intent i = new Intent(voterlogolist.this,amjantapartynominatorlist.class);
        i.putExtra("pin",Rg);
        i.putExtra("name",unamedataFromIntent);
        i.putExtra("id",uiddataFromIntent);
        i.putExtra("phoneno",uphonenodataFromIntent);
        i.putExtra("value",uvaluedataFromIntent);
        i.putExtra("password",upassworddataFromIntent);
        startActivity(i);

    }
}
```

vote.java

```
package com.example.bibeka.iva;

import android.app.ProgressDialog;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;

import com.android.volley.AuthFailureError;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import java.util.HashMap;
import java.util.Map;

public class vote extends AppCompatActivity {
    TextView name2,id2,pin2,party2,vote2;
    String
unamedataFromIntent,uiddataFromIntent,upassworddataFromIntent,uphonenodataFromIntent,u
valuedataFromIntent;
    String dataFromIntent;
    Integer voteupdate,valueupdate;
    String iddataFromIntent;
    String namedataFromIntent;
    String pindataFromIntent;
    String partydataFromIntent;
    String votedataFromIntent;

    private static final String URl
="https://onlyforyou446.000webhostapp.com/voted.php?Rg=";
    private static final String URl1
="https://onlyforyou446.000webhostapp.com/voted1.php?Rg1=";

    private static final String KEY_NAME="vname";
    private static final String KEY_ID="vid";
    private static final String KEY_PASSWORD ="vpassword";
    private static final String KEY_PHONENO="vphoneno";
    private static final String KEY_PIN="vpin";
    private static final String KEY_VALUE="vvalue";
    public static final String KEY_vote = "vote2";
    private ProgressDialog load;
    private ProgressDialog load1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_vote);
        name2=(TextView)findViewById(R.id.namedrop);
        id2=(TextView)findViewById(R.id.iddrop);
        party2=(TextView)findViewById(R.id.phonedrop);
        pin2=(TextView)findViewById(R.id.pindrop);
        vote2=(TextView)findViewById(R.id.votedrop);


         iddataFromIntent = getIntent().getStringExtra("id1");
         namedataFromIntent = getIntent().getStringExtra("name1");
         pindataFromIntent = getIntent().getStringExtra("pin1");
         partydataFromIntent = getIntent().getStringExtra("party1");
         votedataFromIntent = getIntent().getStringExtra("vote1");
        voteupdate=Integer.parseInt(votedataFromIntent);



        //User Data Get Though Intent.


        dataFromIntent = getIntent().getStringExtra("pin");
        unamedataFromIntent = getIntent().getStringExtra("name");
        uiddataFromIntent = getIntent().getStringExtra("id");
        uvaluedataFromIntent = getIntent().getStringExtra("value");
        upassworddataFromIntent = getIntent().getStringExtra("password");
        uphonenodataFromIntent = getIntent().getStringExtra("phoneno");
        valueupdate=Integer.parseInt(uvaluedataFromIntent);




        name2.setText(namedataFromIntent);
        id2.setText(iddataFromIntent);
        party2.setText(partydataFromIntent);
        pin2.setText(pindataFromIntent);
        vote2.setText(votedataFromIntent);

        //Data Showed By Toast
       /* Toast.makeText(this, ""+dataFromIntent, Toast.LENGTH_SHORT).show();

        Toast.makeText(this, ""+unamedataFromIntent, Toast.LENGTH_SHORT).show();

        Toast.makeText(this, ""+uiddataFromIntent, Toast.LENGTH_SHORT).show();

        Toast.makeText(this, ""+uvaluedataFromIntent, Toast.LENGTH_SHORT).show();

        Toast.makeText(this, ""+upassworddataFromIntent, Toast.LENGTH_SHORT).show();

        Toast.makeText(this, ""+uphonenodataFromIntent, Toast.LENGTH_SHORT).show();*/

    }

    public void vote(View view) {
        if (valueupdate==0)
        {
            valueupdate= valueupdate+1;
            voteupdate=voteupdate+1;
            uvaluedataFromIntent=valueupdate.toString().trim();
            votedataFromIntent=voteupdate.toString().trim();
            String Rg=uiddataFromIntent;


            load = ProgressDialog.show(this,"Please wait...","You Are On The
Way...",false,false);

            StringRequest request= new StringRequest(Request.Method.POST, URl+Rg, new
Response.Listener<String>() {
                @Override
                public void onResponse(String response) {
                    Toast.makeText(getApplicationContext(), "Please Wait ",
Toast.LENGTH_SHORT).show();
                    load.dismiss();
                }
            },
                    new Response.ErrorListener() {
                        @Override
                        public void onErrorResponse(VolleyError error) {
                            Toast.makeText(getApplicationContext(), "Check Internet
Connection Or Alrady Registed", Toast.LENGTH_LONG).show();
                        }

                    })
            {
                protected Map<String, String> getParams() throws AuthFailureError {

                    Map<String,String> param = new HashMap<String, String>();
                   /* param.put(KEY_NAME,unamedataFromIntent);
                    param.put(KEY_ID,uiddataFromIntent);
                    param.put(KEY_PASSWORD,upassworddataFromIntent);
                    param.put(KEY_PHONENO,uphonenodataFromIntent);
                    param.put(KEY_PIN,dataFromIntent);*/
                    param.put(KEY_VALUE,uvaluedataFromIntent);
                    return param;

                }
            };
            RequestQueue requestqueue = Volley.newRequestQueue(this);
            requestqueue.add(request);
               votedataupdate();

        }
               else
                  {

                       //COMMAND RUN IF VALUE IS MORE THAN 1
                      Toast.makeText(this, "You are alrady Voated",
Toast.LENGTH_SHORT).show();
                  }
    }

    private void votedataupdate() {
        String Rg1=iddataFromIntent;
        load1 = ProgressDialog.show(this,"Please wait...","You Are On The
Way...",false,false);

        StringRequest request= new StringRequest(Request.Method.POST, URl1+Rg1, new
Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                Toast.makeText(getApplicationContext(), "Vote Succesful",
Toast.LENGTH_SHORT).show();
                load1.dismiss();
            }
        },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {
                        Toast.makeText(getApplicationContext(), "Check Internet
Connection Or Alrady Registed", Toast.LENGTH_LONG).show();
                    }

                })
        {
            protected Map<String, String> getParams() throws AuthFailureError {

                Map<String,String> param = new HashMap<String, String>();
                   /* param.put(KEY_NAME,unamedataFromIntent);
                    param.put(KEY_ID,uiddataFromIntent);
                    param.put(KEY_PASSWORD,upassworddataFromIntent);
                    param.put(KEY_PHONENO,uphonenodataFromIntent);
                    param.put(KEY_PIN,dataFromIntent);*/
                param.put(KEY_vote,votedataFromIntent);
                return param;

            }
        };
        RequestQueue requestqueue = Volley.newRequestQueue(this);
        requestqueue.add(request);

    }
}
```

ViewItem.java

```
package com.example.bibeka.iva;

/**
 * Created by Juned on 6/17/2017.
 */

import android.widget.TextView;

class ViewItem
{
    TextView IdTextView;
    TextView NameTextView;
    TextView PinTextView;
    TextView VoteTextView;
    TextView PartyTextView;

}
```

Subject.java

```
package com.example.bibeka.iva;

/**
 * Created by Juned on 6/17/2017.
 */

public class Subject {

    public String Subject_ID;
    public String Subject_Name;
    public String Subject_Pin;
    public String Subject_Vote;
    public String Subject_Party;


}
```

shownotice.java

```
package com.example.bibeka.iva;

import android.content.Context;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ListView;
import android.widget.ProgressBar;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.List;

public class shownotice extends AppCompatActivity {
    ListView listView;
    String Rg;

    Button button;

    // Server Http URL
    String HTTP_URL = "https://onlyforyou446.000webhostapp.com/notice.php";


    // String to hold complete JSON response object.
    String FinalJSonObject;
    String unamedataFromIntent, uiddataFromIntent, upassworddataFromIntent,
uphonenodataFromIntent, uvaluedataFromIntent;
    String dataFromIntent;
    String Name1, id1, pin1, party1, vote1;
    List<noticesub> CustomSubjectNamesList;
    ProgressBar progressBar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_shownotice);
        listView = (ListView) findViewById(R.id.listView1);
        progressBar = (ProgressBar) findViewById(R.id.ProgressBar1);

        // dataFromIntent = getIntent().getStringExtra("pin");
        // Rg="752057";

       /* unamedataFromIntent = getIntent().getStringExtra("name");
        uiddataFromIntent = getIntent().getStringExtra("id");
        uvaluedataFromIntent = getIntent().getStringExtra("value");
        upassworddataFromIntent = getIntent().getStringExtra("password");
        uphonenodataFromIntent = getIntent().getStringExtra("phoneno");*/


        //Showing progress bar just after button click.
        progressBar.setVisibility(View.VISIBLE);

        // Creating StringRequest and set the JSON server URL in here.
        StringRequest stringRequest = new StringRequest(HTTP_URL,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {

                        // After done Loading store JSON response in FinalJSonObject
string variable.
                        FinalJSonObject = response;

                        // Calling method to parse JSON object.
                        new shownotice.ParseJSonDataClass(shownotice.this).execute();

                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {

                        // Showing error message if something goes wrong.
                        //
Toast.makeText(bjpnominatorlist.this,error.getMessage(),Toast.LENGTH_LONG).show();

                    }
                });

        // Creating String Request Object.
        RequestQueue requestQueue = Volley.newRequestQueue(shownotice.this);

        // Passing String request into RequestQueue.
        requestQueue.add(stringRequest);
    }

    private class ParseJSonDataClass extends AsyncTask<Void, Void, Void> {

        public Context context;

        // Creating List of Subject class.
        public ParseJSonDataClass(Context context) {

            this.context = context;
        }

        @Override
        protected void onPreExecute() {

            super.onPreExecute();
        }

        @Override
        protected Void doInBackground(Void... arg0) {

            try {

                // Checking whether FinalJSonObject is not equals to null.
                if (FinalJSonObject != null) {

                    // Creating and setting up JSON array as null.
                    JSONArray jsonArray = null;
                    try {

                        // Adding JSON response object into JSON array.
                        jsonArray = new JSONArray(FinalJSonObject);

                        // Creating JSON Object.
                        JSONObject jsonObject;

                        // Creating Subject class object.
                        noticesub subject;

                        // Defining CustomSubjectNamesList AS Array List.
                        CustomSubjectNamesList = new ArrayList<noticesub>();

                        for (int i = 0; i < jsonArray.length(); i++) {

                            subject = new noticesub();

                            jsonObject = jsonArray.getJSONObject(i);

                            //Storing ID into subject list.
                            subject.Subject_DATE = jsonObject.getString("date");

                            //Storing Subject name in subject list.
                            subject.Subject_SUBJECT = jsonObject.getString("subject");



                            // Adding subject list object into CustomSubjectNamesList.
                            CustomSubjectNamesList.add(subject);
                        }
                    } catch (JSONException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                    }
                }
            } catch (Exception e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
            return null;
        }

        @Override
        protected void onPostExecute(Void result)

        {
            // After all done loading set complete CustomSubjectNamesList with 
application context to ListView adapter.
            noticelistviewadapter adapter = new 
noticelistviewadapter(CustomSubjectNamesList, context);

            // Setting up all data into ListView.
            listView.setAdapter(adapter);

            // Hiding progress bar after all JSON loading done.
            progressBar.setVisibility(View.GONE);

        }

    }
}
```

noticeviewitem.java

```
package com.example.bibeka.iva;

import android.widget.TextView;

/**
 * Created by Bibeka on 3/19/2018.
 */

public class noticeviewitem
{
    TextView DateTextView;
    TextView SubjectTextView;
}
```

noticesub.java

```
package com.example.bibeka.iva;

/**
 * Created by Bibeka on 3/19/2018.
 */

public class noticesub
{
    public String Subject_DATE;
    public String Subject_SUBJECT;
}
```

noticelistviewadapter.java

```
package com.example.bibeka.iva;

import android.app.Activity;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.TextView;

import java.util.List;

/**
 * Created by Bibeka on 3/19/2018.
 */

public class noticelistviewadapter extends BaseAdapter
{
    Context context;

    List<noticesub> TempSubjectList;

    public noticelistviewadapter(List<noticesub> listValue, Context context)
    {
        this.context = context;
        this.TempSubjectList = listValue;
    }

    @Override
    public int getCount()
    {
        return this.TempSubjectList.size();
    }

    @Override
    public Object getItem(int position)
    {
        return this.TempSubjectList.get(position);
    }

    @Override
    public long getItemId(int position)
    {
        return position;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent)
    {
        noticeviewitem viewItem = null;

        if(convertView == null)
        {
            viewItem = new noticeviewitem();

            LayoutInflater layoutInfiater =
(LayoutInflater)this.context.getSystemService(Activity.LAYOUT_INFLATER_SERVICE);

            convertView = layoutInfiater.inflate(R.layout.noticeview, null);
            viewItem.DateTextView = (TextView)convertView.findViewById(R.id.datedrop);
            viewItem.SubjectTextView =
(TextView)convertView.findViewById(R.id.subject);
            convertView.setTag(viewItem);
        }
        else
        {
            viewItem = (noticeviewitem) convertView.getTag();
        }



        viewItem.DateTextView.setText(TempSubjectList.get(position).Subject_DATE);

viewItem.SubjectTextView.setText(TempSubjectList.get(position).Subject_SUBJECT);


        return convertView;
    }
}
```

newuser.java

```
package com.example.bibeka.iva;

import android.app.ProgressDialog;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import com.android.volley.AuthFailureError;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import java.util.HashMap;
import java.util.Map;

public class newuser extends AppCompatActivity {
EditText e1,e2,e3,e4,e5;
    Integer value;
    String name,id,pass,valuepass;
    String phoneno,pin;
    private static final String URl
="https://onlyforyou446.000webhostapp.com/voterlistadd.php";
    private static final String KEY_NAME="vname";
    private static final String KEY_ID="vid";
    private static final String KEY_PASSWORD ="vpassword";
    private static final String KEY_PHONENO="vphoneno";
    private static final String KEY_PIN="vpin";
    private static final String KEY_VALUE="vvalue";
    private ProgressDialog load;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_newuser);
        reffrence();
    }


    private void reffrence() {
        e1=(EditText)findViewById(R.id.vname);
        e2=(EditText)findViewById(R.id.vtno);
        e3=(EditText)findViewById(R.id.vphno);
        e4=(EditText)findViewById(R.id.vpass);
        e5=(EditText)findViewById(R.id.vpin);
    }




    public void signup(View view) {
        name=e1.getText().toString();
        id=e2.getText().toString();
        phoneno=e3.getText().toString();
        pass=e4.getText().toString();
        pin=e5.getText().toString();
        value=0;
        valuepass=value.toString();
        Toast.makeText(this, ""+name, Toast.LENGTH_SHORT).show();
        load = ProgressDialog.show(this,"Please wait...","You Are On The
Way...",false,false);

        StringRequest request= new StringRequest(Request.Method.POST, URl, new
Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                Toast.makeText(getApplicationContext(), "Resgistation Succesful",
Toast.LENGTH_SHORT).show();
                load.dismiss();
            }
        },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {
                        Toast.makeText(getApplicationContext(), "Check Internet
Connection Or Alrady Registed", Toast.LENGTH_LONG).show();
                    }

                })
        {
            protected Map<String, String> getParams() throws AuthFailureError {

                Map<String,String> param = new HashMap<String, String>();
                param.put(KEY_NAME,name);
                param.put(KEY_ID,id);
                param.put(KEY_PASSWORD,pass);
                param.put(KEY_PHONENO,phoneno);
                param.put(KEY_PIN,pin);
                param.put(KEY_VALUE,valuepass);
                return param;

            }
        };
        RequestQueue requestqueue = Volley.newRequestQueue(this);
        requestqueue.add(request);
    }
}
```

newcandidates.java

```
package com.example.bibeka.iva;

import android.app.ProgressDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import com.android.volley.AuthFailureError;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import java.util.HashMap;
import java.util.Map;

public class newcandicates extends AppCompatActivity {
    EditText e1,e2,e3,e4,e5;
    Integer value;
    String name,id,pass,vote;
    String phoneno,pin;
    private static final String URl
="https://onlyforyou446.000webhostapp.com/nominilistadd.php";
    private static final String KEY_NAME="cname";
    private static final String KEY_ID="cid";
    private static final String KEY_PARTY ="cparty";
    private static final String KEY_PIN="cpin";
    private static final String KEY_VOTE="cvote";
    private ProgressDialog load;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_newcandicates);
        e1=(EditText)findViewById(R.id.vname);
        e2=(EditText)findViewById(R.id.vtno);//patryn
        e3=(EditText)findViewById(R.id.vphno);//id
        e5=(EditText)findViewById(R.id.vpin);
    }

    public void signup(View view) {
        name=e1.getText().toString();
        phoneno=e2.getText().toString();
        id=e3.getText().toString();
        pin=e5.getText().toString();
        value=0;
        vote=value.toString();
        Toast.makeText(this, ""+name, Toast.LENGTH_SHORT).show();
        load = ProgressDialog.show(this,"Please wait...","You Are On The
Way...",false,false);
        StringRequest request= new StringRequest(Request.Method.POST, URl, new
Response.Listener<String>() {
```

```
            @Override
            public void onResponse(String response) {
                Toast.makeText(getApplicationContext(), "Resgistation Succesful",
Toast.LENGTH_SHORT).show();
                load.dismiss();
            }
        },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {
                        Toast.makeText(getApplicationContext(), "Check Internet
Connection Or Alrady Registed", Toast.LENGTH_LONG).show();
                    }

                })
        {
            protected Map<String, String> getParams() throws AuthFailureError {

                Map<String,String> param = new HashMap<String, String>();
                param.put(KEY_NAME,name);
                param.put(KEY_ID,id);
                param.put(KEY_PARTY,phoneno);
                param.put(KEY_PIN,pin);
                param.put(KEY_VOTE,vote);
                return param;

            }
        };
        RequestQueue requestqueue = Volley.newRequestQueue(this);
        requestqueue.add(request);
    }
}
```

Login.java

```
package com.example.bibeka.iva;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import com.android.volley.AuthFailureError;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import java.util.HashMap;
import java.util.Map;

public class Login extends AppCompatActivity {
    EditText ed1,ed2;
    String v1id,v1pass;
    public String lurl = "https://onlyforyou446.000webhostapp.com/voteruserlogin.php";
    private static final String KEY_ID = "v_id";
    private static final String KEY_PASSWORD = "v_Password";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        reffrence();
    }

    private void reffrence() {

        ed1=(EditText)findViewById(R.id.vid);
        ed2=(EditText)findViewById(R.id.vpass);


    }

    public void login(View view) {
       v1id=ed1.getText().toString();
       v1pass=ed2.getText().toString();


        StringRequest request1 = new StringRequest(Request.Method.POST, lurl, new
Response.Listener<String>() {
            @Override
            public void onResponse(String response) {

                if (response.trim().equals("success")) {
                    Toast.makeText(getApplicationContext(), "YOU ARE LOGED IN",
Toast.LENGTH_SHORT).show();
                    Intent a = new Intent(Login.this,faterlogin.class);
                    a.putExtra("r",v1id);
                    startActivity(a);



                } else {
                    Toast.makeText(getApplicationContext(), "CHECK YOUR ID AND
PASSWORD OR INTENET CONNECTION", Toast.LENGTH_LONG).show();
//   /
                }

            }
        },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {
                        Toast.makeText(getApplicationContext(), "Error OCCURED OR
CHECK YOUR INTENET CONNECTION", Toast.LENGTH_SHORT).show();
                    }

                })

        {
            @Override
            protected Map<String, String> getParams() throws AuthFailureError {
                Map<String,String> map = new HashMap<String,String>();
                map.put(KEY_ID,v1id);
                map.put(KEY_PASSWORD,v1pass);
                return map;
            }
        };
        RequestQueue requestQueue = Volley.newRequestQueue(this);
        requestQueue.add(request1);


    }
}
```

ListViewAdapter.java

```
package com.example.bibeka.iva;

/**
 * Created by Juned on 6/17/2017.
 */

import android.app.Activity;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.TextView;

import java.util.List;

public class ListViewAdapter extends BaseAdapter
{
    Context context;

    List<Subject> TempSubjectList;

    public ListViewAdapter(List<Subject> listValue, Context context)
    {
        this.context = context;
        this.TempSubjectList = listValue;
    }

    @Override
    public int getCount()
    {
        return this.TempSubjectList.size();
    }

    @Override
    public Object getItem(int position)
    {
        return this.TempSubjectList.get(position);
    }

    @Override
    public long getItemId(int position)
    {
        return position;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent)
    {
        ViewItem viewItem = null;

        if(convertView == null)
        {
            viewItem = new ViewItem();

            LayoutInflater layoutInfiater = 
(LayoutInflater)this.context.getSystemService(Activity.LAYOUT_INFLATER_SERVICE);

            convertView = layoutInfiater.inflate(R.layout.listvalue, null);

            viewItem.IdTextView = 
(TextView)convertView.findViewById(R.id.textviewSubjectName);

            viewItem.NameTextView = 
(TextView)convertView.findViewById(R.id.textviewID);
            viewItem.PinTextView = (TextView)convertView.findViewById(R.id.textView3);
            viewItem.VoteTextView = 
(TextView)convertView.findViewById(R.id.textView2);
            viewItem.PartyTextView = 
(TextView)convertView.findViewById(R.id.textView);

            convertView.setTag(viewItem);
        }
        else
        {
            viewItem = (ViewItem) convertView.getTag();
        }



        viewItem.IdTextView.setText(TempSubjectList.get(position).Subject_ID);
        viewItem.NameTextView.setText(TempSubjectList.get(position).Subject_Name);
        viewItem.PinTextView.setText(TempSubjectList.get(position).Subject_Pin);
        viewItem.VoteTextView.setText(TempSubjectList.get(position).Subject_Vote);
        viewItem.PartyTextView.setText(TempSubjectList.get(position).Subject_Party);


        return convertView;
    }
}
```

afterlogin.java

```
package com.example.bibeka.iva;

import android.app.ProgressDialog;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

public class faterlogin extends AppCompatActivity {
    private ProgressDialog load;
    TextView namedrop1,iddrop1,pindrop1,phonedrop1;
    String pin = "";
    String name = "";
    String id = "";
    String value = "";
    String phoneno1 = "";
    String password = "";


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_faterlogin);
        reference();

        String dataFromIntent = getIntent().getStringExtra("r");
        String Rg=dataFromIntent;



        load = ProgressDialog.show(this,"Please wait...","Fetching
Details...",false,false);
        String url2 = Config.DATA_URL+Rg;

        StringRequest stringRequest1 = new StringRequest(url2, new
Response.Listener<String>() {
            @Override
            public void onResponse(String response11) {
                showJSON11(response11);
            }
        },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {

Toast.makeText(faterlogin.this,error.getMessage().toString(),Toast.LENGTH_LONG).show()
;
                    }
                });

        RequestQueue requestQueue2 = Volley.newRequestQueue(this);
        requestQueue2.add(stringRequest1);
    }

    private void reference() {
        namedrop1=(TextView)findViewById(R.id.namedrop);
        iddrop1=(TextView)findViewById(R.id.iddrop);
        pindrop1=(TextView)findViewById(R.id.pindrop);
        phonedrop1=(TextView)findViewById(R.id.phonedrop);
    }

    private void showJSON11(String response11) {


        try {
            JSONObject jsonObject = new JSONObject(response11);
            JSONArray result1 = jsonObject.getJSONArray(Config.JSON_ARRAY);
            JSONObject collegeData1 = result1.getJSONObject(0);
            name = collegeData1.getString(Config.KEY_NAME);
            id = collegeData1.getString(Config.KEY_ID);
            pin = collegeData1.getString(Config.KEY_PIN);
            value = collegeData1.getString(Config.VALUE);
            phoneno1 = collegeData1.getString(Config.KEY_phoneno);
            password = collegeData1.getString(Config.KEY_password);


        }
        catch (JSONException e) {
            e.printStackTrace();
        }


        namedrop1.setText(name);
        iddrop1.setText(id);
        pindrop1.setText(pin);
        phonedrop1.setText(phoneno1);

       /* Toast.makeText(this, "Name :"+name, Toast.LENGTH_LONG).show();
        Toast.makeText(this, "Id :"+id, Toast.LENGTH_LONG).show();
        Toast.makeText(this, "Pin :"+pin, Toast.LENGTH_LONG).show();
        Toast.makeText(this, "Value :"+value, Toast.LENGTH_LONG).show();
        Toast.makeText(this, "password"+password, Toast.LENGTH_SHORT).show();
        Toast.makeText(this, "phoneo"+phoneno1, Toast.LENGTH_SHORT).show();*/
        load.dismiss();


    }

    public void vote(View view) {
        Intent i = new Intent(faterlogin.this,voterlogolist.class);
        startActivity(i);
        i.putExtra("pin",pin);
        i.putExtra("name",name);
        i.putExtra("id",id);
        i.putExtra("phoneno",phoneno1);
        i.putExtra("value",value);
        i.putExtra("password",password);
        startActivity(i);

    }
}
```

congressnominatorlist.java

```
package com.example.bibeka.iva;

import android.content.Context;
import android.content.Intent;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.List;

public class congressnominatorlist extends AppCompatActivity {
    ListView listView;

    Button button;

    // Server Http URL
String HTTP_URL = "https://onlyforyou446.000webhostapp.com/congress.php?Rg=";

    // String to hold complete JSON response object.
    String FinalJSonObject ;
    String Rg;


    ProgressBar progressBar;
    String Name1,id1,pin1,party1,vote1;
    List<Subject> CustomSubjectNamesList;
    String
unamedataFromIntent,uiddataFromIntent,upassworddataFromIntent,uphonenodataFromIntent,u
valuedataFromIntent;
    String dataFromIntent;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_congressnominatorlist);
        listView = (ListView) findViewById(R.id.listView1);
        progressBar = (ProgressBar)findViewById(R.id.ProgressBar1);

         dataFromIntent = getIntent().getStringExtra("pin");
         Rg=dataFromIntent;


        unamedataFromIntent = getIntent().getStringExtra("name");
        uiddataFromIntent = getIntent().getStringExtra("id");
        uvaluedataFromIntent = getIntent().getStringExtra("value");
        upassworddataFromIntent = getIntent().getStringExtra("password");
        uphonenodataFromIntent = getIntent().getStringExtra("phoneno");


        //Showing progress bar just after button click.
        progressBar.setVisibility(View.VISIBLE);

        // Creating StringRequest and set the JSON server URL in here.
        StringRequest stringRequest = new StringRequest(HTTP_URL+Rg,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {

                        // After done Loading store JSON response in FinalJSonObject
string variable.
```

```
                        FinalJSonObject = response ;

                        // Calling method to parse JSON object.
                        new ParseJSonDataClass(congressnominatorlist.this).execute();

                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {

                        // Showing error message if something goes wrong.

Toast.makeText(congressnominatorlist.this,error.getMessage(),Toast.LENGTH_LONG).show()
;

                    }
                });

        // Creating String Request Object.
        RequestQueue requestQueue =
Volley.newRequestQueue(congressnominatorlist.this);

        // Passing String request into RequestQueue.
        requestQueue.add(stringRequest);

        listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position,
long id) {
                ViewItem view1;
                view1 = new ViewItem();
                Subject sub;
                sub= new Subject();
                ListViewAdapter list1;
                // Name1=sub.Subject_Name;
                Name1=CustomSubjectNamesList.get(position).Subject_Name;
                id1=CustomSubjectNamesList.get(position).Subject_ID;
                pin1=CustomSubjectNamesList.get(position).Subject_Pin;
                party1=CustomSubjectNamesList.get(position).Subject_Party;
                vote1=CustomSubjectNamesList.get(position).Subject_Vote;
                Intent a = new Intent(congressnominatorlist.this,vote.class);
                a.putExtra("name1",Name1);
                a.putExtra("id1",id1);
                a.putExtra("pin1",pin1);
                a.putExtra("party1",party1);
                a.putExtra("vote1",vote1);


                //User Data Get Through Intent.
                a.putExtra("pin",Rg);
                a.putExtra("name",unamedataFromIntent);
                a.putExtra("id",uiddataFromIntent);
                a.putExtra("phoneno",uphonenodataFromIntent);
                a.putExtra("value",uvaluedataFromIntent);
                a.putExtra("password",upassworddataFromIntent);


                startActivity(a);
                // Toast.makeText(MainActivity.this,""+Name1,
Toast.LENGTH_SHORT).show();
            }
        });
```

```
}

private class ParseJSonDataClass extends AsyncTask<Void, Void, Void> {

    public Context context;

    // Creating List of Subject class.


    public ParseJSonDataClass(Context context) {

        this.context = context;
    }

    @Override
    protected void onPreExecute() {

        super.onPreExecute();
    }

    @Override
    protected Void doInBackground(Void... arg0) {

        try {

            // Checking whether FinalJSonObject is not equals to null.
            if (FinalJSonObject != null) {

                // Creating and setting up JSON array as null.
                JSONArray jsonArray = null;
                try {

                    // Adding JSON response object into JSON array.
                    jsonArray = new JSONArray(FinalJSonObject);

                    // Creating JSON Object.
                    JSONObject jsonObject;

                    // Creating Subject class object.
                    Subject subject;

                    // Defining CustomSubjectNamesList AS Array List.
                    CustomSubjectNamesList = new ArrayList<Subject>();

                    for (int i = 0; i < jsonArray.length(); i++) {

                        subject = new Subject();

                        jsonObject = jsonArray.getJSONObject(i);

                        //Storing ID into subject list.
                        subject.Subject_ID = jsonObject.getString("nomini_id");

                        //Storing Subject name in subject list.
                        subject.Subject_Name = jsonObject.getString("name");
                        subject.Subject_Pin = jsonObject.getString("pin");
                        subject.Subject_Vote = jsonObject.getString("vote");
                        subject.Subject_Party = jsonObject.getString("party");


                        // Adding subject list object into CustomSubjectNamesList.
                        CustomSubjectNamesList.add(subject);
                        }
                    } catch (JSONException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                    }
                }
            } catch (Exception e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
            return null;
        }

        @Override
        protected void onPostExecute(Void result)

        {
            // After all done loading set complete CustomSubjectNamesList with
application context to ListView adapter.
            ListViewAdapter adapter = new ListViewAdapter(CustomSubjectNamesList,
context);

            // Setting up all data into ListView.
            listView.setAdapter(adapter);

            // Hiding progress bar after all JSON loading done.
            progressBar.setVisibility(View.GONE);

        }
    }

}
```

Config.java

```
package com.example.bibeka.iva;

/**
 * Created by Bibeka on 2/19/2018.
 */

public class Config {
    public static final String DATA_URL =
"https://onlyforyou446.000webhostapp.com/fetchvoterlist.php?Rg=";
    public static final String DATA_URL1 =
"https://onlyforyou446.000webhostapp.com/nominilistfetch.php?Rg=";


    public static final String KEY_NAME = "name";
    public static final String KEY_ID = "id";
    public static final String KEY_PIN = "pin2";
    public static final String JSON_ARRAY = "result";
    public static final String VALUE = "value2";
    public static final String KEY_password = "password2";
    public static final String KEY_phoneno = "phoneno2";
```

bjpnominatorlist.java:

```
package com.example.bibeka.iva;

import android.content.Context;
import android.content.Intent;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.List;

public class bjpnominatorlist extends AppCompatActivity {
    ListView listView;
    String Rg;

    Button button;

    // Server Http URL
    String HTTP_URL =
"https://onlyforyou446.000webhostapp.com/bjpnominationlist.php?Rg=";
    String h= HTTP_URL+Rg;


    // String to hold complete JSON response object.
    String FinalJSonObject ;
    String
unamedataFromIntent,uiddataFromIntent,upassworddataFromIntent,uphonenodataFromIntent,u
valuedataFromIntent;
    String dataFromIntent;
    String Name1,id1,pin1,party1,vote1;
    List<Subject> CustomSubjectNamesList;


    ProgressBar progressBar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_bjpnominatorlist);

        listView = (ListView) findViewById(R.id.listView1);
        progressBar = (ProgressBar)findViewById(R.id.ProgressBar1);

        dataFromIntent = getIntent().getStringExtra("pin");
```

```
         Rg=dataFromIntent;

        unamedataFromIntent = getIntent().getStringExtra("name");
        uiddataFromIntent = getIntent().getStringExtra("id");
        uvaluedataFromIntent = getIntent().getStringExtra("value");
        upassworddataFromIntent = getIntent().getStringExtra("password");
        uphonenodataFromIntent = getIntent().getStringExtra("phoneno");



        //Showing progress bar just after button click.
        progressBar.setVisibility(View.VISIBLE);

        // Creating StringRequest and set the JSON server URL in here.
        StringRequest stringRequest = new StringRequest(HTTP_URL+Rg,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {

                        // After done Loading store JSON response in FinalJSonObject
string variable.
                        FinalJSonObject = response ;

                        // Calling method to parse JSON object.
                        new ParseJSonDataClass(bjpnominatorlist.this).execute();

                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {

                        // Showing error message if something goes wrong.

Toast.makeText(bjpnominatorlist.this,error.getMessage(),Toast.LENGTH_LONG).show();

                    }
                });

        // Creating String Request Object.
        RequestQueue requestQueue = Volley.newRequestQueue(bjpnominatorlist.this);

        // Passing String request into RequestQueue.
        requestQueue.add(stringRequest);


        listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position,
long id) {
                ViewItem view1;
                view1 = new ViewItem();
                Subject sub;
                sub= new Subject();
                ListViewAdapter list1;
                // Name1=sub.Subject_Name;
                Name1=CustomSubjectNamesList.get(position).Subject_Name;
                id1=CustomSubjectNamesList.get(position).Subject_ID;
                pin1=CustomSubjectNamesList.get(position).Subject_Pin;
                party1=CustomSubjectNamesList.get(position).Subject_Party;
                vote1=CustomSubjectNamesList.get(position).Subject_Vote;
                Intent a = new Intent(bjpnominatorlist.this,vote.class);
                a.putExtra("name1",Name1);
                a.putExtra("id1",id1);
                a.putExtra("pin1",pin1);
                a.putExtra("party1",party1);
                a.putExtra("vote1",vote1);


                //User Data Get Through Intent.
                a.putExtra("pin",Rg);
                a.putExtra("name",unamedataFromIntent);
                a.putExtra("id",uiddataFromIntent);
                a.putExtra("phoneno",uphonenodataFromIntent);
                a.putExtra("value",uvaluedataFromIntent);
                a.putExtra("password",upassworddataFromIntent);


                startActivity(a);
                // Toast.makeText(MainActivity.this,""+Name1,
Toast.LENGTH_SHORT).show();
            }
        });
    }


    private class ParseJSonDataClass extends AsyncTask<Void, Void, Void> {

        public Context context;

        // Creating List of Subject class.


        public ParseJSonDataClass(Context context) {

            this.context = context;
        }

        @Override
        protected void onPreExecute() {

            super.onPreExecute();
        }

        @Override
        protected Void doInBackground(Void... arg0) {

            try {

                // Checking whether FinalJSonObject is not equals to null.
                if (FinalJSonObject != null) {

                    // Creating and setting up JSON array as null.
                    JSONArray jsonArray = null;
                    try {

                        // Adding JSON response object into JSON array.
                        jsonArray = new JSONArray(FinalJSonObject);

                        // Creating JSON Object.
                        JSONObject jsonObject;

                        // Creating Subject class object.
                        Subject subject;

                        // Defining CustomSubjectNamesList AS Array List.
                        CustomSubjectNamesList = new ArrayList<Subject>();

                        for (int i = 0; i < jsonArray.length(); i++) {

                            subject = new Subject();

                            jsonObject = jsonArray.getJSONObject(i);

                            //Storing ID into subject list.
                            subject.Subject_ID = jsonObject.getString("nomini_id");

                            //Storing Subject name in subject list.
                            subject.Subject_Name = jsonObject.getString("name");
                            subject.Subject_Pin = jsonObject.getString("pin");
                            subject.Subject_Vote = jsonObject.getString("vote");
                            subject.Subject_Party = jsonObject.getString("party");


                            // Adding subject list object into CustomSubjectNamesList.
                            CustomSubjectNamesList.add(subject);
                        }
                    } catch (JSONException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                    }
                }
            } catch (Exception e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
            return null;
        }

        @Override
        protected void onPostExecute(Void result)

        {
            // After all done loading set complete CustomSubjectNamesList with
application context to ListView adapter.
            ListViewAdapter adapter = new ListViewAdapter(CustomSubjectNamesList,
context);

            // Setting up all data into ListView.
            listView.setAdapter(adapter);

            // Hiding progress bar after all JSON loading done.
            progressBar.setVisibility(View.GONE);

        }
    }



}
```

34.Php Codes :

public_html/bjpnominationlist.php

```
<?php
$servername = "localhost";
$username = "id2339034_aptron";
$password = "aptron";
$dbname = "id2339034_bibek";
// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$id =$_GET['Rg'];
$sql = "SELECT * FROM nomini_list   WHERE pin='".$id."'AND party='BJP'";
$result = $conn->query($sql);
if ($result->num_rows >0) {
 while($row[] = $result->fetch_assoc()) {
 $tem = $row;
 $json = json_encode($tem); }
 } else {
 echo "No Results Found.";
}
 echo $json;
$conn->close();?>
```

notice.php

```
<?php
$servername = "localhost";
$username = "id2339034_aptron";
$password = "aptron";
$dbname = "id2339034_bibek";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$sql = "SELECT * FROM notice";

$result = $conn->query($sql);

if ($result->num_rows >0) {


 while($row[] = $result->fetch_assoc()) {

 $tem = $row;

 $json = json_encode($tem);


 }

} else {
 echo "No Results Found.";
}
 echo $json;
$conn->close();

?>
```

voted1.php

```
<?php
$servername = "localhost";
$username = "id2339034_aptron";
$password = "aptron";
$dbname = "id2339034_bibek";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$v_value =$_POST['vote2'];
$id =$_GET['Rg1'];
```

```
$sql = "UPDATE nomini_list SET vote='$v_value' WHERE vote='$id'";

if (mysqli_query($conn, $sql)) {
    echo "Record updated successfully";
} else {
    echo "Error updating record: " . mysqli_error($conn);
}

mysqli_close($conn);
?>
```

winerlist.php

```
<?php
$servername = "localhost";
$username = "id2339034_aptron";
$password = "aptron";
$dbname = "id2339034_bibek";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$id =$_GET['Rg'];
$sql = "SELECT * FROM nomini_list   WHERE pin='".$id."'";

$result = $conn->query($sql);

if ($result->num_rows >0) {


 while($row[] = $result->fetch_assoc()) {

 $tem = $row;

 $json = json_encode($tem);


 }

} else {
 echo "No Results Found.";
}
 echo $json;
$conn->close();

?>
```

36.CONCLUSION

In this paper, an idea for voting online, directly from the voter’s home is presented, keeping in mind, the present day Indian Voting System scenarios, along with its real time implementation problems. It overcomes problems of rigging, taking a step towards, making the voting system secure, and, introduces OPEN PUBLIC VOTING , where everyone would be allowed to vote in front of other people .This paper also aims to promote voting, by saving the voters, from huge waiting time in the queues.

37.REFERENCES :

- **Software Engineering (Rajib Mall)**
- **Google (www.google.co.in)**

THANK YOU