**DESIGN A DATA BASE SYSTEM THAT TAKES INFORMATION OF CITIZENS NAME, PHONE NUMBER, GENDER AND ADDRESS DURING A GENERAL ELECTION AND STORE IT ON A DATA BASE THAT CAN BE ACCESS**

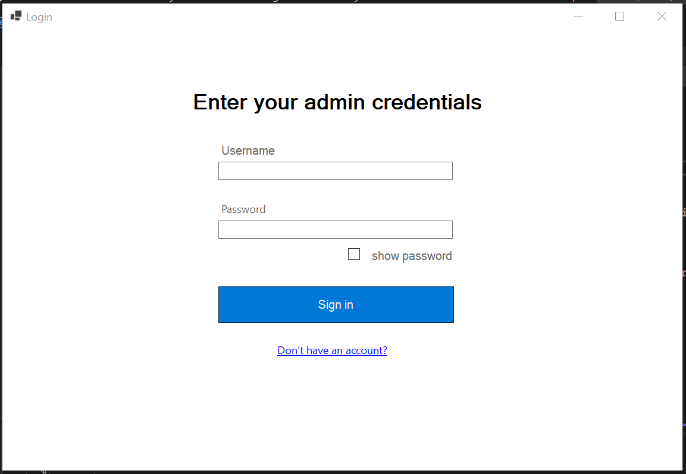
**GROUP 3**

|  |  |
| --- | --- |
| OLUFUNKE OJIKUTU | 20003383 |
| ISRAEL DAVID | 20002359 |
| AYOOLA EBENEZER OLUWASEGUN | 20003028 |
| ALEBIOSU AANU KENNETH | 20001751 |
| AKANDE AYOMIDE YUSUF | 20002992 |
| AYOADE AKINDOLA SOLOMON | 20005011 |
| BENSON DEBORAH OLUWASEUN | 20004552 |
| OLADEPO RIDWAN KAYODE | 20002362 |
| OLADOSU FUNKE ESTHER | 20004194 |
| IDOWU OLALEYE STEPHEN | 20003427 |
| ADEBOWALE PHILLIP | 20005537 |
| FOLAYAN ADESEWA ELIZABETH | 20005112 |
| ALABI ABIDEMI FOLORUNSHO | 20003500 |
| OTUNLA BARAKAT OMOTANWA | 20005109 |
| OYEBAMIJI EBENEZER | 20005055 |
| OLALERE DANIEL | 20004314 |
| OLAOSUN PEACE | 20002891 |
| OWOLABI MARY | 20003979 |
| AJETUNMOBI OLUWAFISAYOMI EMMANUELLA | 20006068 |
| MUSTAPHA ADEKUNLE HAZZAN | 20003766 |
| OLANIYI HELEN HANNAH | 20006047 |
| IBRAHIM MUBARAK OLADIMEJI | 20003091 |
| OLANIRAN STEPHEN TAIWO | 20003475 |
| ADEMILOLA MICHAEL A. | 20004821 |
| ADENIYI PHILIP AYOMIDE | 20005256 |
| AKINWUMI BLESSING GRACE | 20005217 |
| ADELEKE ILIASU | 20005959 |
| ADEBAYO VICTORIA FERANMI | 20003191 |
| OBA ADEDAYO OLUYOMI | 20003921 |
| ONWODI KINGSLEY ECHEZONA | 20003573 |
| BABALOLA ISAAC ADESHINA | 20001022 |
| OLAGUNJU NAIMOT OLAJUMOKE | 20002696 |
| ALIYU JAMIU OLASUNKANMI | 20005874 |

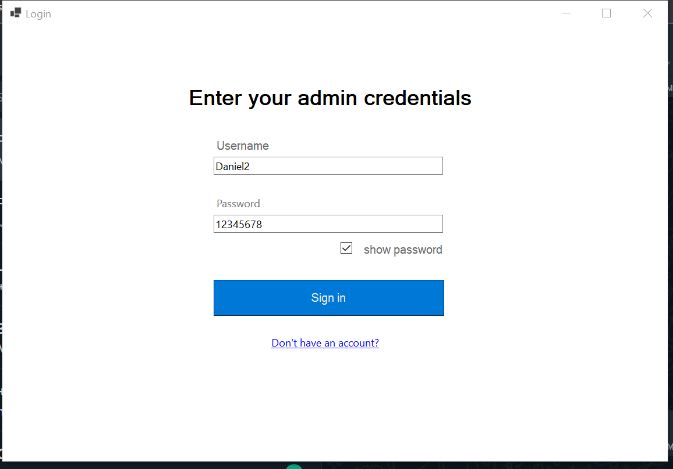
**NAMES AND MATRIC NUMBER**

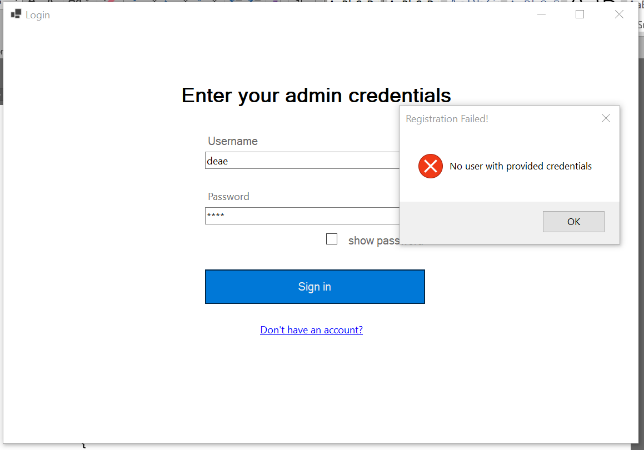
SUBMITTED TO: **AKANDE NOAH O.**

1. **ADMIN LOGIN FORM**

****

1. Show if password is correct





1. Login form Data Validation

public partial class Login : Form

{

public static IMongoClient client = new MongoClient("mongodb+srv://junior:junior22@cluster0.ctxr861.mongodb.net/?retryWrites=true&w=majority");

public static IMongoDatabase db = client.GetDatabase("election\_data");

public static IMongoCollection<User> collection = db.GetCollection<User>("user");

public Login()

{

InitializeComponent();

}

public class User

{

[BsonId]

public ObjectId Id { get; set; }

[BsonElement("Username")]

public string Username { get; set; }

[BsonElement("Password")]

public string Password { get; set; }

public User(string username, string password)

{

Username = username;

Password = password;

}

}

private void linkLabel1\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

new Register().Show();

this.Hide();

}

private void signinBtn\_Click(object sender, EventArgs e)

{

if(txtUsername.Text == "" && txtPassword.Text == "")

{

MessageBox.Show(

"Username and Password fields are empty",

"Registration Failed!",

MessageBoxButtons.OK,

MessageBoxIcon.Error

);

}

else

{

var filter = Builders<User>.Filter.Eq("Username", txtUsername.Text);

var validateUser = collection.Find(filter).FirstOrDefault();

if(validateUser == null)

{

MessageBox.Show(

"No user with provided credentials",

"Registration Failed!",

MessageBoxButtons.OK,

MessageBoxIcon.Error

);

}

else if(validateUser != null)

{

new InputForm().Show();

this.Hide();

}

};

}

private void checkInput\_CheckedChanged(object sender, EventArgs e)

{

if(checkInput.Checked == true)

{

txtPassword.PasswordChar = '\0';

}

else

{

txtPassword.PasswordChar = '\*';

}

}

private void Login\_Load(object sender, EventArgs e)

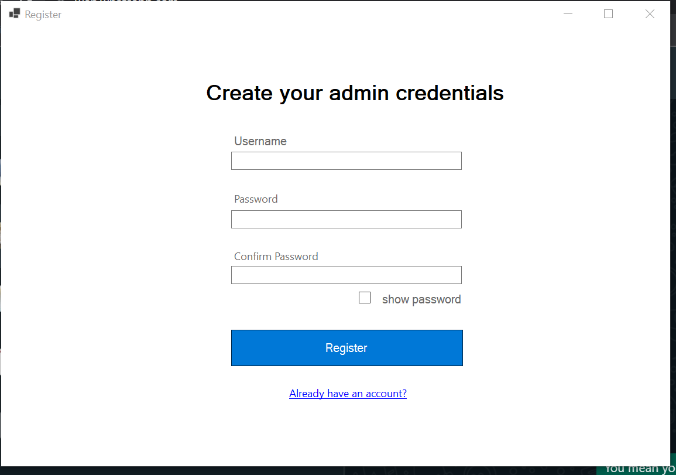
{

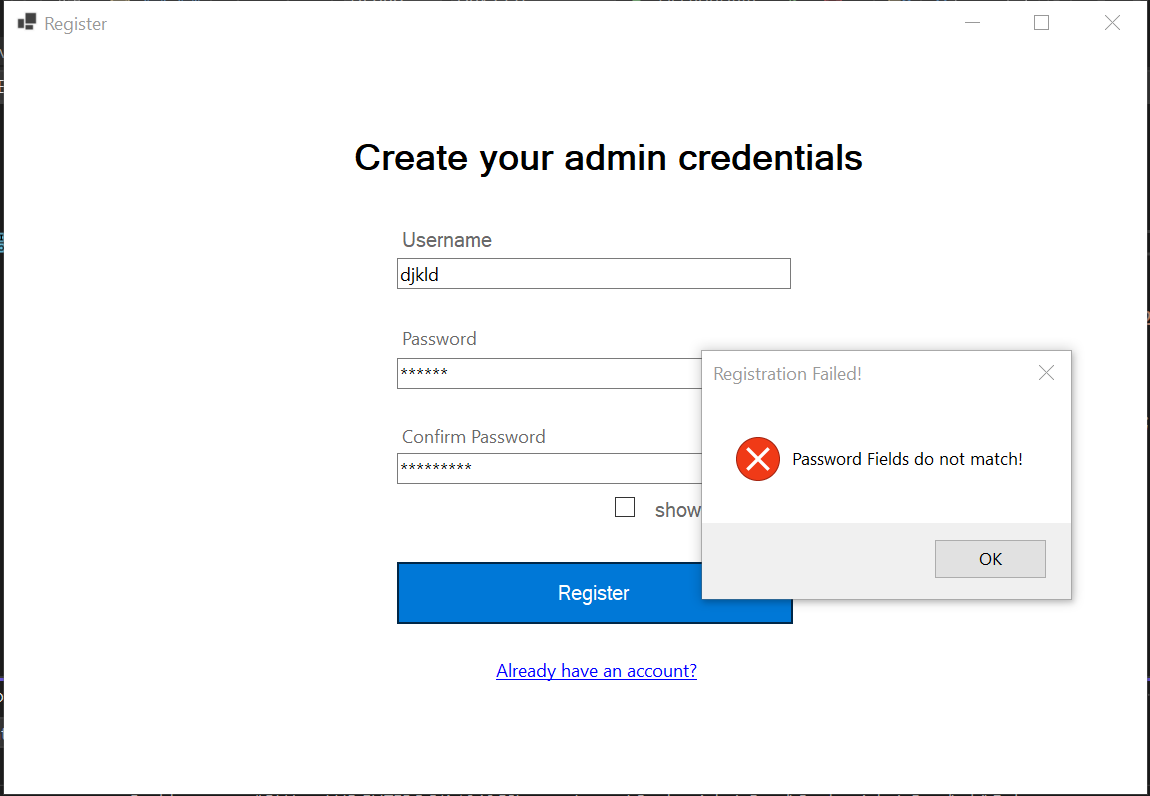
}

}

}

1. **ADMIN REGISTRATION FORM**

****



1. Admin Registration Form Data Validation

public partial class Register : Form

{

public static IMongoClient client = new MongoClient("mongodb+srv://junior:junior22@cluster0.ctxr861.mongodb.net/?retryWrites=true&w=majority");

public static IMongoDatabase db = client.GetDatabase("election\_data");

public static IMongoCollection<User> collection = db.GetCollection<User>("user");

public Register()

{

InitializeComponent();

}

public class User

{

[BsonId]

public ObjectId Id { get; set; }

[BsonElement("Username")]

public string Username { get; set; }

[BsonElement("Password")]

public string Password { get; set; }

public User(string username, string password)

{

Username = username;

Password = password;

}

}

private void checkBox1\_CheckedChanged(object sender, EventArgs e)

{

if(checkInput.Checked == false)

{

txtPassword.PasswordChar = '\0';

txtConfirmPassword.PasswordChar = '\0';

}

else

{

txtPassword.PasswordChar = '\*';

txtConfirmPassword.PasswordChar = '\*';

}

}

private void button1\_Click(object sender, EventArgs e)

{

if(txtUsername.Text == "" && txtPassword.Text == "" && txtConfirmPassword.Text == "")

{

MessageBox.Show(

"Username and Password fields are empty",

"Registration Failed!",

MessageBoxButtons.OK,

MessageBoxIcon.Error

);

}

else if(txtPassword.Text != txtConfirmPassword.Text)

{

MessageBox.Show(

"Password Fields do not match!",

"Registration Failed!",

MessageBoxButtons.OK,

MessageBoxIcon.Error

);

}

else

{

User user = new User(txtUsername.Text, Cryptography.Encrypt(txtPassword.Text.ToString()));

collection.InsertOne(user);

MessageBox.Show(

"Registration Successful please click ok to be redirected to login page!",

"Registration Success!",

MessageBoxButtons.OK,

MessageBoxIcon.None

);

new Login().Show();

this.Hide();

}

}

private void linkLabel1\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

new Login().Show();

this.Hide();

}

private void Register\_Load(object sender, EventArgs e)

{

}

private void checkInput\_CheckedChanged(object sender, EventArgs e)

{

if (checkInput.Checked == true)

{

txtPassword.PasswordChar = '\0';

txtConfirmPassword.PasswordChar = '\0';

}

else

{

txtPassword.PasswordChar = '\*';

txtConfirmPassword.PasswordChar = '\*';

}

}

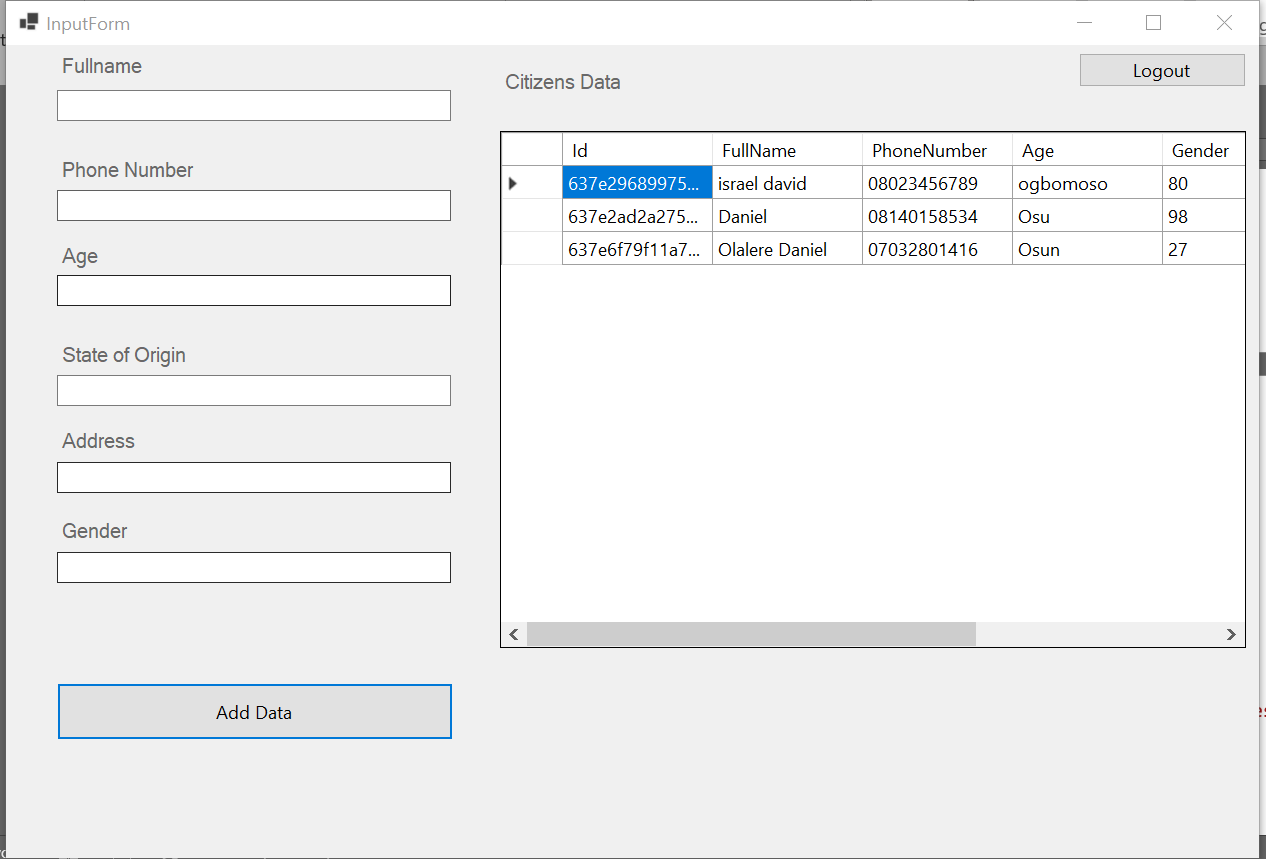
private void txtUsername\_TextChanged(object sender, EventArgs e)

{

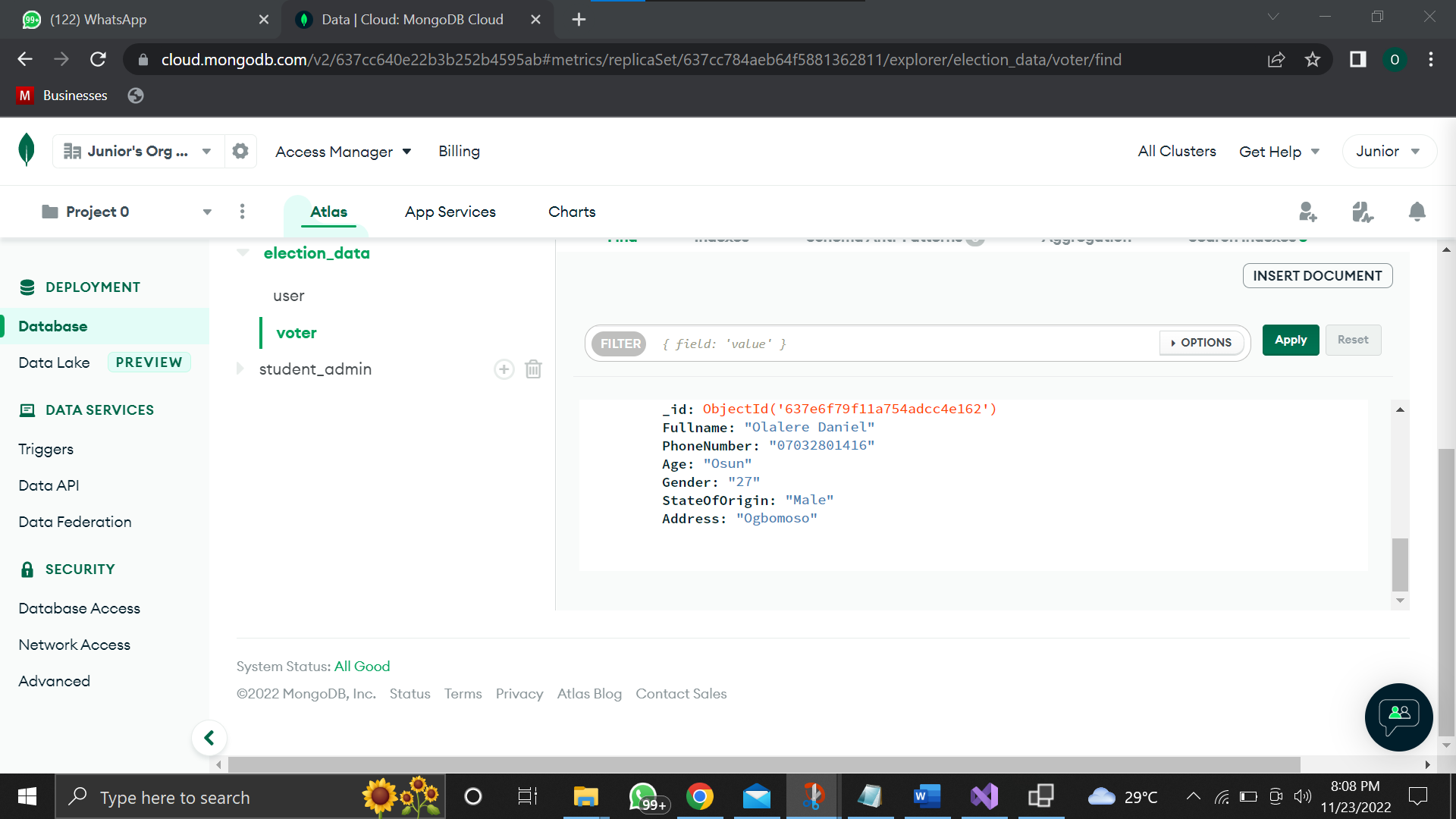
}

}

1. INPUT FORM PANEL THAT SAVES DETAILS ON DATA BASE



This updated the data in the Data Grid on the Cloud Data base automatically and can be accessed.



public partial class InputForm : Form

{

public static IMongoClient client = new MongoClient("mongodb+srv://junior:junior22@cluster0.ctxr861.mongodb.net/?retryWrites=true&w=majority");

public static IMongoDatabase db = client.GetDatabase("election\_data");

public static IMongoCollection<Voters> collection = db.GetCollection<Voters>("voter");

public InputForm()

{

InitializeComponent();

}

public class Voters

{

[BsonId]

public ObjectId Id { get; set; }

[BsonElement("Fullname")]

public string FullName { get; set; }

[BsonElement("PhoneNumber")]

public string PhoneNumber { get; set; }

[BsonElement("Age")]

public string Age { get; set; }

[BsonElement("Gender")]

public string Gender { get; set; }

[BsonElement("StateOfOrigin")]

public string StateOfOrigin { get; set; }

[BsonElement("Address")]

public string Address { get; set; }

public Voters(string fullname, string phone\_number, string age, string gender, string state\_of\_origin, string address)

{

FullName = fullname;

PhoneNumber = phone\_number;

Age = age;

Gender = gender;

StateOfOrigin = state\_of\_origin;

Address = address;

}

}

private void InputForm\_Load(object sender, EventArgs e)

{

}

private void addData\_Click(object sender, EventArgs e)

{

if(txtFullname.Text == "" && txtPhoneNumber.Text == "" && txtStateOfOrigin.Text == "" && txtAge.Text == "" && txtGender.Text == "" && txtAddress.Text == "")

{

MessageBox.Show(

"Please fill a input fields",

"Data Registration Failed!",

MessageBoxButtons.OK,

MessageBoxIcon.Error

);

}

else

{

Voters student = new Voters(

txtFullname.Text,

txtPhoneNumber.Text,

txtStateOfOrigin.Text,

txtAge.Text,

txtGender.Text,

txtAddress.Text

);

collection.InsertOne(student);

MessageBox.Show(

"Registration Successful please click ok to be redirected to login page!",

"Registration Success!",

MessageBoxButtons.OK,

MessageBoxIcon.None

);

ReadData();

txtFullname.Text = "";

txtPhoneNumber.Text = "";

txtStateOfOrigin.Text = "";

txtAge.Text = "";

txtGender.Text = "";

txtAddress.Text = "";

}

}

public void ReadData()

{

List<Voters> list = collection.AsQueryable().ToList();

dataGridView1.DataSource = list;

txtFullname.Text = dataGridView1.Rows[0].Cells[0].Value.ToString();

}

private void button1\_Click(object sender, EventArgs e)

{

new Login().Show();

this.Hide();

}

private void txtFullname\_TextChanged(object sender, EventArgs e)

{

}

}

}