DEMOGRAPHIC STATISTICS

A PROJECT REPORT
Submitted by

CB.EN.U4CSE16617	KANNAN MANI. S. M
CB.EN.U4CSE16620	MURALI KRISHTNA. J
CB.EN.U4CSE16634	RAAJESH. M

in partial fulfilment for the award
of grade for the course
15CSE302 Database Management Systems

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
AMRITA SCHOOL OF ENGINEERING, COIMBATORE
AMRITA VISHWA VIDYAPEETHAM
COIMBATORE 641 112
October 2018

ACKNOWLEDGEMENT

We would like to thank the Pro Chancellor **Brahmachari Abhayamrita Chaitanya** Vice Chancellor **Dr. P. Venkat Rangan and Dean Dr. Sasangan Ramanathan** for having provided the necessary infrastructure required for the completion of the project.

We are grateful to **Dr. Latha Parameswaran**, Chairperson, Department of Computer Science and Engineering for providing an opportunity to do this project.

We express our deepest gratitude to our faculty **Ms. Anupa Vijay** and panel members' **Ms. T Bagyammal** and **Ms. G R Ramya** for the invaluable guidance and encouragement throughout the project. Without their support and helping hand, this project would never have been a success.

We sincerely thank all the teaching and non-teaching staff members who have helped us in our work during this project.

We owe all our knowledge to all the teachers who taught us and cared for us, they are our true well-wishers. It's their blessing that catapulted us through the way of success. We owe them all a piece of our heart. We wouldn't be happier than to be known as their student.

Last but not the least, our sincere thanks to all of our parents and friends who have patiently extended all sorts of help for completing this project. We could not have successfully completed our project without the blessings of our respected chancellor beloved **Amma**.

TABLE OF CONTENTS

ABSTRACT

Demographic Statistics is an online system or agency to help take a census of Indian citizens to connect them to government records and to read from this data to help the government make policies and national decisions based on the statistics derived from the census.

Today there is a new ID or record called the Aadhaar that almost each and every citizen in India has .Our main motto is to create a website to register the census details and other statistics to the Aadhaar ID associated with a person.

The primary objective of this web site is to take a census and display Demographic Statistics using the help of Bar charts, Pie charts etc .This statistics data will be a summary of the census taken and will be available to the public. The main purpose of the web site is to help the government officials while deciding policies that affects the nation.

This project still requires more development of attributes and needs to broaden the scope of data that it can obtain from citizens .The more the attributes and the more the citizens are connected, the better we can produce statistics that will help change their lives .

Summarizing all the data from the census into various charts generated by the front end can make life easier not only for politicians but also those like journalists. The pie charts can be used to make info graphics and other forms of data representation.

PREVIEW OF THE PROJECT

INTRODUCTION

Basically our website starts with a login/signup page where admins can sign in to register citizens along with their basic details. Admin’s have the privilege to modify user data and approve the details via records physically first before entering in the system.

After signing in the registration page is displayed. The registration page starts with basic mandatory details like Name,Aadhaar ID, DOB etc .After basic detail registrations, we move on to other registration like Car, Education, Vehichle, Marital Status, Kids Details etc which not all citizens might have.

Add / View / Modify basic Citizen Details
Other pages include like,

1. Vehicle Registration
2. Education
3. Kids
4. Employment and Tax

NEED OF THE PROJECT AND MOTIVATION

To connect a majority of the citizens life to Aadhaar and to make decisions out of the aggregated data to help make the citizens life better .

With continued declines in fertility and mortality, the global population's shift toward an older age structure, known as population aging, will accelerate. Older adults' (ages 65+) share of the global population increased from 5 percent in 1960 to 9 percent in 2018 and is projected to rise to 16 percent by 2050, with the segment ages 85 and older growing the fastest. Children's (ages 0 to 14) share is falling, from 37 percent in 1960, to 26 percent in 2018, with a projected decrease to 21 percent by 2050.

TOOLS USED

FRONT END :

CSS
Bootstrap
Embedded JavaScript
Angular JS

MIDDLEWARE :

Express JS
Mongoose

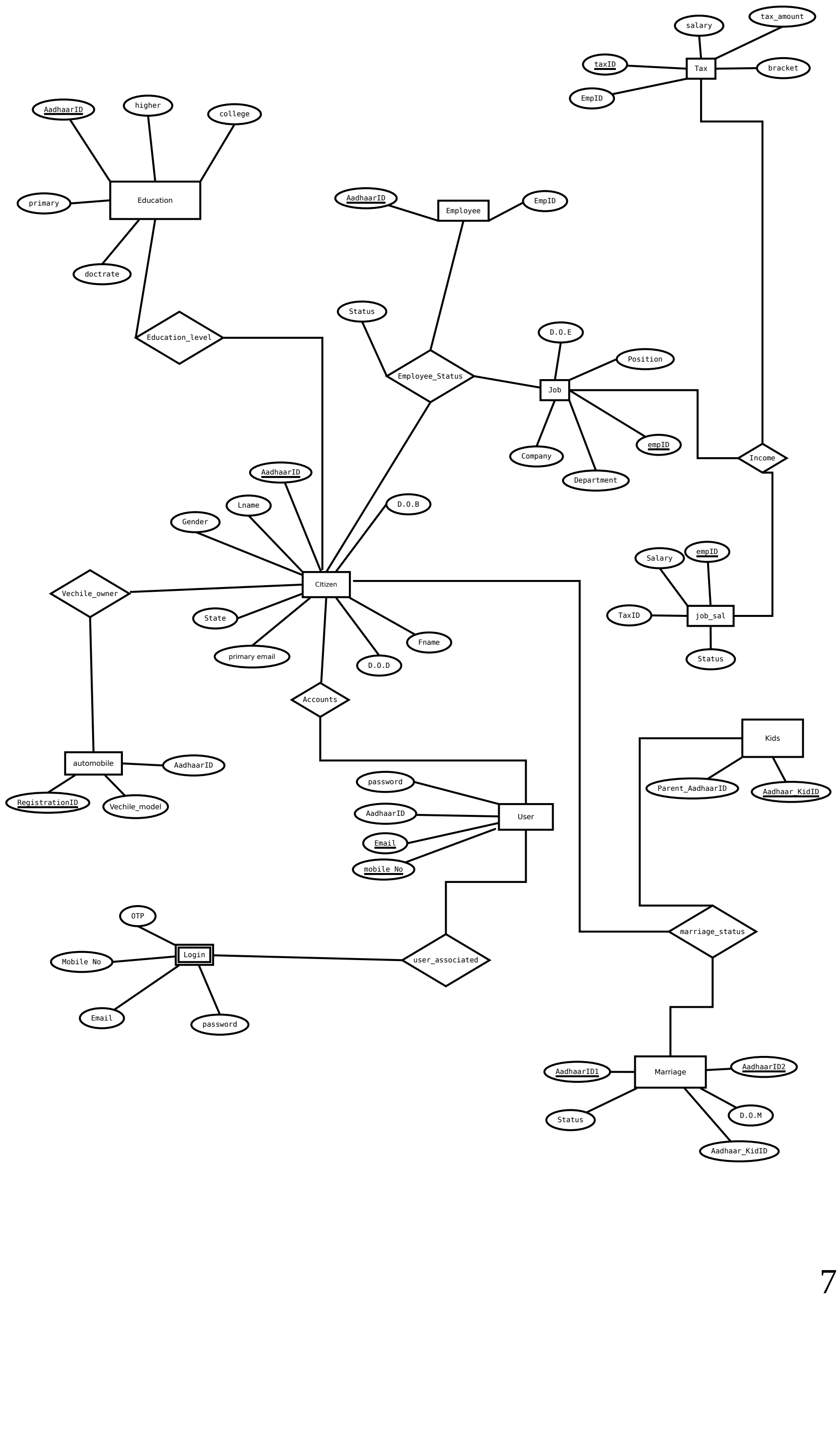
BACK END:

Node JS
MongoDB

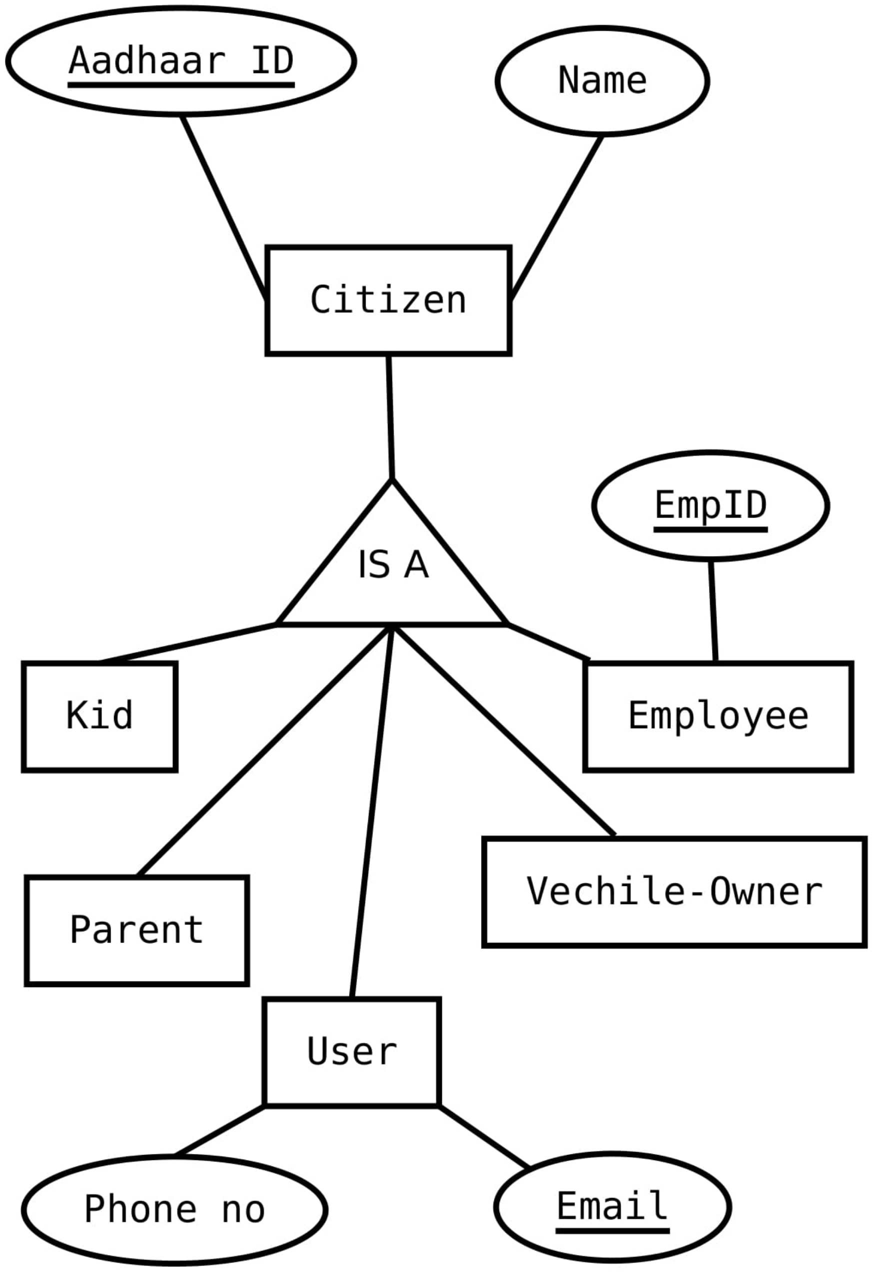
SOURCE CONTROL:

Github

ENTITY RELATIONSHIP MODEL (ERM)



EXTENDED ENTITY RELATIONSHIP MODEL (EERM)



NORMALIZATION

0NF:

citizen(aadhaarID,gender,DOB,name,email,state,DOD,education)

automobile(aadhaarID,vehicle_model,registration_no)

marriage(aadharID1,aadharID2,DOM,kids,status)

user(aadhaarID,username,email,password)

login(username,password)

employment(empID,aadhaarID,company,DOE,status,salary,position,taxID)

tax(taxID,salary,bracket,tax_amount);

1NF:

1)It should only have single(atomic) valued attributes/columns.

2)Values stored in a column should be of the same domain

3)All the columns in a table should have unique names

4)The order in which data is stored, does not matter.

citizen(aadhaarID,gender,DOB,Fname,Lname,primary_email,state,DOD)

//add new table for education as its not atomic

//email to primary_email

automobile(registrationID,aadhaarID,vehicle_model)

marriage(aadharID1,aadharID2,DOM,kidID,status)

//Change kids to kidID as a foreign key to reference another table exclusively for kids

user(aadhaarID,username,email,password)

login(username,password)

employment(empID,aadhaarID,company,DOE,status,salary,position,department,taxID)

tax(taxID,salary,bracket,tax_amount)

education(aadhaarID,primary,higher,college,higher,doctorate)

kids(parent_aadhaarID,aadhaarID,name)

2NF

No Partial Deps

citizen(aadhaarID,gender,DOB,Fname,Lname,primary_email,state,DOD)

//automobile(registrationID,aadhaarID,vehicle_model)

//vehicle model depends on registrationID and not aadhaarID. =>split table.

automobile_ownership(registrationID,aadhaarID)

automobile_identification(registrationID,vehicle_model)

marriage(aadharID1,aadharID2,DOM,kidID,status)

user(aadhaarID,username,email,password)

login(username,password)

//employment(empID,aadhaarID,company,DOE,status,salary,position,department,taxID)

//Here , company,DOE,status,salary and etc doesn't depend on aadhaarID and hence we have to split the table.

employment(aadhaarID,empID)

job(empID,company,DOE,status,salary,position,department,taxID)

tax(taxID,salary,bracket,tax_amount)

education(aadhaarID,primary,higher,college,higher,doctorate)

//kids(parent_aadhaarID,aadhaarID,name)

//name depends on aadhaarID alone and not parent_aadhaarID

=>remove name

kids(parent_aadhaarID,aadhaarID)

3NF:

citizen(aadhaarID,gender,DOB,Fname,Lname,primary_email,state,DOD)

automobile_identification(registrationID,vehicle_model)

marriage(aadharID1,aadharID2,DOM,kidID,status)

//user(aadhaarID,username,email,password)

//assume your email name is username

// all emails are unique assuming under same domain

// email → username

user(aadhaarID, email)

user_id(email, username, password)

login(username,password)

employment(aadhaarID,empID)

//job(empID,company,DOE,status,salary,position,department,taxID)

// breaking this table into 2, Assume this

// empID → company

//salary → company

//breaking this table further helps

job(empID, company, DOE, position, department)

job_sal(company, salary, status, taxID)

tax(taxID,salary,bracket,tax_amount)

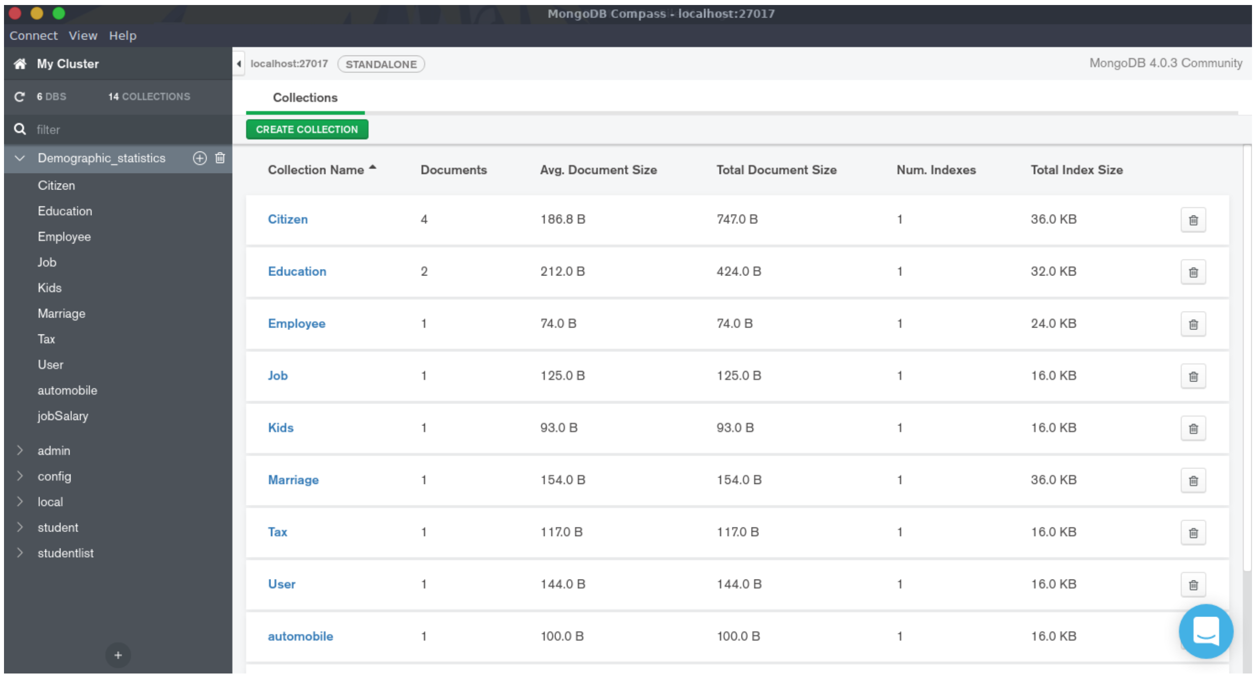
education(aadhaarID,primary,higher,college,doctorate)

kids(parent_aadhaarID,aadhaarID)

TABLE CREATION

Using Mongoose Middleware in javascript, Tables are created

The visualization is from MongoDB Compass.



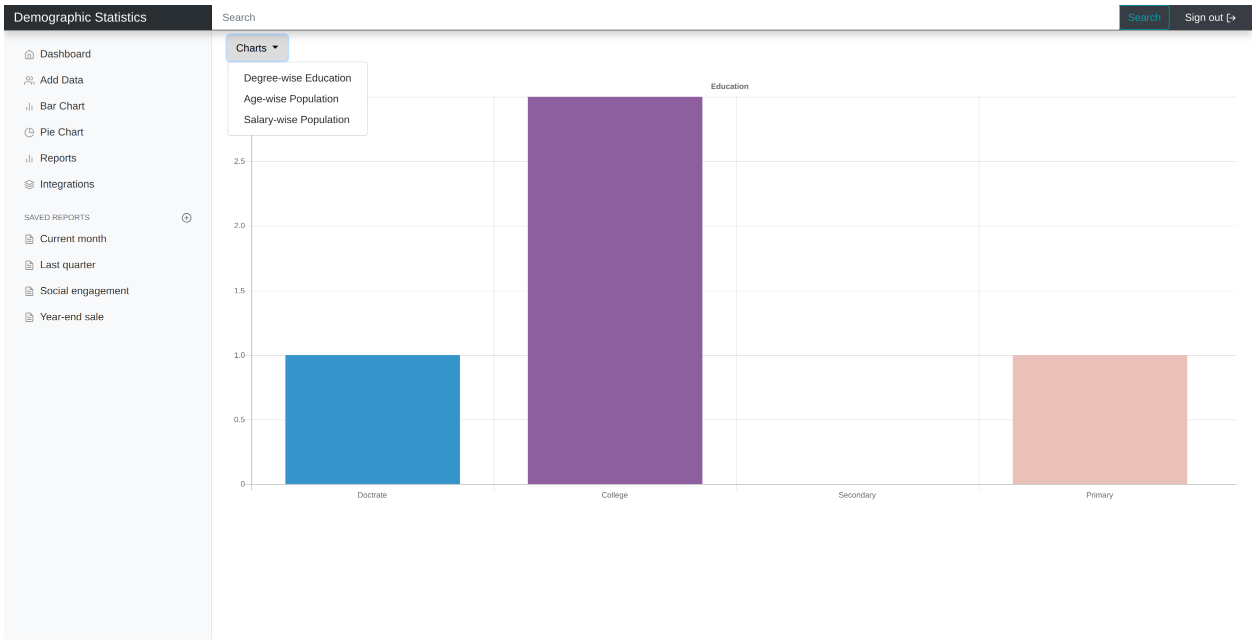
Collection Name	Documents	Avg. Document Size	Total Document Size	Num. Indexes	Total Index Size
Citizen	4	186.8 B	747.0 B	1	36.0 KB
Education	2	212.0 B	424.0 B	1	32.0 KB
Employee	1	74.0 B	74.0 B	1	24.0 KB
Job	1	125.0 B	125.0 B	1	16.0 KB
Kids	1	93.0 B	93.0 B	1	16.0 KB
Marriage	1	154.0 B	154.0 B	1	36.0 KB
Tax	1	117.0 B	117.0 B	1	16.0 KB
User	1	144.0 B	144.0 B	1	16.0 KB
automobile	1	100.0 B	100.0 B	1	16.0 KB

MongoDB gives out data in JSON format

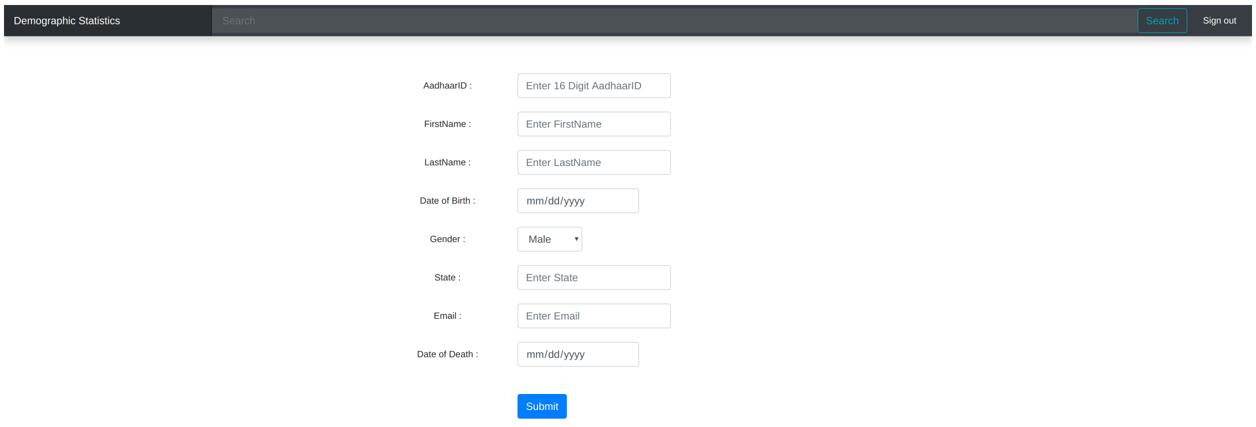
Converting these tables (JSON) into Java Script Objects and using them to display aggregated conclusions

UI DESIGN

VISUALIZING CHARTS



ADD USER DATA



AadhaarID :	Enter 16 Digit AadhaarID
FirstName :	Enter FirstName
LastName :	Enter LastName
Date of Birth :	mm/dd/yyyy
Gender :	Male
State :	Enter State
Email :	Enter Email
Date of Death :	mm/dd/yyyy

DATABASE CONNECTIVITY

```
var db = require('./app/models/db');



//mongoose starts here
var mongoose=require('mongoose');
mongoose.connect('mongodb://localhost/demograph-stats-project',{ useNewUrlParser: true } );

//Starting Connection
var db=mongoose.connection;
db.on('error', console.error.bind(console, 'connection error:')); //incase of error



var citizenSchema=new mongoose.Schema({
    fname: String,
    lname: String,
    dob: Date,
    dod: Date,          //date of death
    gender: String,
    state: String,
    email: String,
    aadhaarID: {
        type:String,
        required:true,
        unique:true
    }
});
var citizen=mongoose.model('citizen',citizenSchema);
```

SAMPLE UI CODE

index.ejs

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">
       <link  rel="icon"  href="http://icons.iconarchive.com/icons/kyo-tux/phuzion/128/Misc-Stats-
icon.png">

    <title>Demographic Statistics</title>

    <!-- Bootstrap core CSS -->
    <link href="https://getbootstrap.com/docs/4.1/dist/css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom styles for this template -->
                <link     href="https://getbootstrap.com/docs/4.1/examples/dashboard/dashboard.css"
rel="stylesheet">
  </head>
  <body>
    <nav class="navbar navbar-dark fixed-top bg-dark flex-md-nowrap p-0 shadow">
    <a class="navbar-brand col-sm-3 col-md-2 mr-0" href="/">Demographic Statistics</a>

         <input  class="form-control  form-control-dark  w-100"  type="text"  placeholder="Search"
aria-label="Search">
      <button class="btn btn-outline-info my-2 my-sm-0" type="submit">Search</button>
    <ul class="navbar-nav px-4 ">
      <li class="nav-item text-nowrap">
        <a class="nav-link active" href="#">
         Sign out
         <span data-feather="log-out"></span></a>
      </li>
    </ul>
  </nav>

  <div class="container-fluid">
      <div class="row">
        <nav class="col-md-2 d-none d-md-block bg-light sidebar">
         <div class="sidebar-sticky">
           <ul class="nav flex-column">
             <li class="nav-item">
               <a class="nav-link active" href="/">
                 <span data-feather="home"></span>
                 Dashboard <span class="sr-only">(current)</span>
               </a>
             </li>
             <li class="nav-item">
               <a class="nav-link" href="/newcitizen">
                 <span data-feather="users"></span>
                 Add Data
               </a>
             </li>
             <li class="nav-item">
               <a class="nav-link" href="/bar-education">
                 <span data-feather="bar-chart-2"></span>
                 Bar Charts
               </a>
           </li>
           <li class="nav-item">
             <a class="nav-link" href="/pie-sex-ratio">
               <span data-feather="pie-chart"></span>
               Pie Chart
             </a>
           </li>
           <li class="nav-item">
             <a class="nav-link" href="#">
               <span data-feather="bar-chart-2"></span>
               Reports
             </a>
           </li>
           <li class="nav-item">
             <a class="nav-link" href="#">
               <span data-feather="layers"></span>
               Integrations
             </a>
           </li>
         </ul>

           <h6 class="sidebar-heading d-flex justify-content-between align-items-center px-3 mt-4
  mb-1 text-muted">
           <span>Saved reports</span>
           <a class="d-flex align-items-center text-muted" href="#">
             <span data-feather="plus-circle"></span>
           </a>
         </h6>
         <ul class="nav flex-column mb-2">
           <li class="nav-item">
             <a class="nav-link" href="#">
             <span data-feather="file-text"></span>
             Current month
           </a>
         </li>
         <li class="nav-item">
           <a class="nav-link" href="#">
             <span data-feather="file-text"></span>
             Last quarter
           </a>
         </li>
         <li class="nav-item">
           <a class="nav-link" href="#">
             <span data-feather="file-text"></span>
             Social engagement
           </a>
         </li>
         <li class="nav-item">
           <a class="nav-link" href="#">
             <span data-feather="file-text"></span>
             Year-end sale
           </a>
         </li>
        </ul>
      </div>
    </nav>

    <!-- Bootstrap core JavaScript
  ================================================== -->
  <!-- Placed at the end of the document so the pages load faster →
```

```
  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" </script>
         <script>window.jQuery || document.write('<script src="https://code.jquery.com/jquery-
3.3.1.slim.min.js"><\/script>')</script>
       <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js"></
script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"></script>

    <!-- Icons -->
    <script src="https://unpkg.com/feather-icons/dist/feather.min.js"></script>
    <script>
      feather.replace()
    </script>

  </body>
</html>
```

education-barchart.ejs // Chart code

```
<script>
        new Chart(document.getElementById("bar-chart"), {
          type: 'bar',
          data: {
            labels: ["Doctrate", "College", "Secondary", "Primary"],
            datasets: [
              {
                label: "No. of people",
                backgroundColor: ["#3e95cd", "#8e5ea2","#3cba9f","#e8c3b9"],
                data: [<%=doctrate%>,<%=college%>,<%=secondary%>,<%=primary%>]
```

```
      }
    ]
   },
   options: {
    legend: { display: false },
    title: {
      display: true,
      text: 'Education  '
    }
   }
 });
 </script>
```

CONCLUSION

Now a days manual process for the citizens to apply for their government records like passport, driving license, voters id, pan card etc has become a huge task. The main object of the website is to reduce the effort by the candidate and save his time and avoid unwanted rushes at the government offices and assure a smooth working schedule at government offices. The main features of this site includes flexibility, reduce manual work in an efficient manner, a quick, convenient, reliable and effective way to apply for their government records. The project could very well be enhanced further as per the requirements

REFERENCES

https://mongoosejs.com/docs/
https://expressjs.com/docs/
https://gitbootstrap.com/docs/