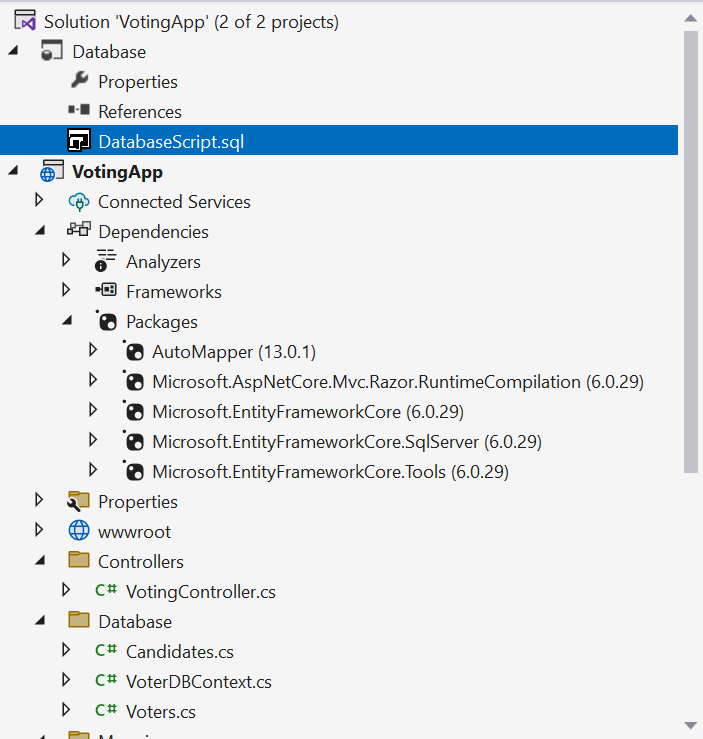
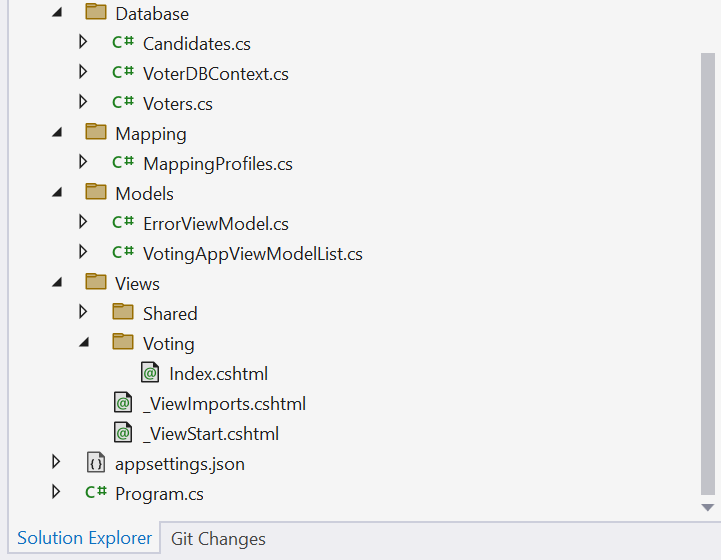
**VOTING APPLICATION**

**Index:**

1. Folder Structure of Voting Application – Page 2
2. Source Code – Page 3
   * 1. VotingController.cs
     2. Candidates.cs
     3. Voters.cs
     4. VoteingDBContext .cs
     5. MappingProfiles.cs
     6. VotingAppViewModelList.cs
     7. Program.cs
     8. appsettings.json
     9. Views/Voting/Index.cshtml
     10. SQL Database Script
3. Preview of Application - Page 24

**Below is the folder structure of Voting Application:**





**Source code of the Voting Applications:**

1. **VotingController.cs**

using AutoMapper;  
using Microsoft.AspNetCore.Mvc;  
using Microsoft.EntityFrameworkCore;  
using System;  
using System.Diagnostics;  
using VotingApp.Database;  
using VotingApp.Models;  
  
namespace VotingApp.Controllers  
{  
    public class VotingController : Controller  
    {  
        private readonly ILogger<VotingController> \_logger;  
        private readonly VoteingDBContext \_context;  
        private readonly IMapper \_mapper;  
        public VotingController(VoteingDBContext context, IMapper mapper, ILogger<VotingController> logger)  
        {  
            \_logger = logger;  
            \_context = context;  
            \_mapper = mapper;  
        }  
  
        public IActionResult Index()  
        {  
            return View(getVotingAppModel());  
        }  
  
        private VotingAppViewModelList getVotingAppModel()  
        {  
            var voters = \_context.Voters.ToList();  
            var candidates = \_context.Candidates.ToList();  
            var votingAppModel = new VotingAppViewModelList();  
            if (voters != null && candidates != null)  
            {  
                var lstVoters = \_mapper.Map<List<Database.Voters>, List<Models.Voter>>(voters);  
                var lstCandidates = \_mapper.Map<List<Database.Candidates>, List<Models.Candidate>>(candidates);  
  
                votingAppModel = new VotingAppViewModelList  
                {  
                    VotersList = lstVoters,  
                    CandidatesList = lstCandidates  
                };  
            }  
            return votingAppModel;  
        }  
        public IActionResult AddVoter([FromBody] Voter model)  
        {  
            if (model.Name != null)  
            {  
                \_context.Voters.Add(new Database.Voters() { VoterId = new Guid(), Name = model.Name, HasVoted = model.HasVoted });  
                \_context.SaveChanges();  
            }  
            return RedirectToActionPermanent("Index", "Voting", getVotingAppModel());  
        }  
  
        public IActionResult AddCandidate([FromBody] Candidate model)  
        {  
            if (model.Name != null)  
            {  
                \_context.Candidates.Add(new Database.Candidates() { CandidateId = new Guid(), Name = model.Name, Votes = 0 });  
                \_context.SaveChanges();  
            }  
            return RedirectToActionPermanent("Index", "Voting", getVotingAppModel());  
        }  
  
        public IActionResult SubmitVote([FromBody] Vote model)  
        {  
            if (model != null)  
            {  
                //if not voted before  
                var voter = \_context.Voters.FirstOrDefault(item => item.VoterId == model.VoterId && item.HasVoted == false);  
                if (voter != null)  
                {  
                    voter.HasVoted = true;  
  
                    var candidate = \_context.Candidates.FirstOrDefault(item => item.CandidateId == model.CandidateId);  
                    if (candidate != null)  
                    {  
                        candidate.Votes = candidate.Votes + 1;  
                    }  
  
                    \_context.SaveChanges();  
                }  
            }  
            return RedirectToActionPermanent("Index", "Voting", getVotingAppModel());  
        }  
  
        [ResponseCache(Duration = 0, Location = ResponseCacheLocation.None, NoStore = true)]  
        public IActionResult Error()  
        {  
            return View(new ErrorViewModel { RequestId = Activity.Current?.Id ?? HttpContext.TraceIdentifier });  
        }  
    }  
}

**2. Candidates.cs**

using System.ComponentModel.DataAnnotations.Schema;  
using System.ComponentModel.DataAnnotations;  
  
namespace VotingApp.Database  
{  
    public class Candidates  
    {  
        [Key]  
        [DatabaseGenerated(DatabaseGeneratedOption.Identity)]  
        public Guid CandidateId { get; set; }  
        public string Name { get; set; }  
        public int Votes { get; set; }  
  
    }  
}

**3. Voters.cs**

using System.ComponentModel.DataAnnotations.Schema;  
using System.ComponentModel.DataAnnotations;  
  
namespace VotingApp.Database  
{  
    public class Voters  
    {  
        [Key]  
        [DatabaseGenerated(DatabaseGeneratedOption.Identity)]  
        public Guid VoterId { get; set; }  
        public string Name { get; set; }  
        public bool HasVoted { get; set; }  
  
    }  
}

**4. VoteingDBContext .cs**

using Microsoft.EntityFrameworkCore;  
  
namespace VotingApp.Database;  
public class VoteingDBContext : DbContext  
{  
    public DbSet<Voters> Voters { get; set; }  
    public DbSet<Candidates> Candidates { get; set; }  
    public VoteingDBContext(DbContextOptions<VoteingDBContext> options) : base(options) { }  
}

**5. MappingProfiles.cs**

using AutoMapper;  
using System.Net;  
  
namespace VotingApp.Mapping  
{  
    public class MappingProfiles : Profile  
    {  
        public MappingProfiles()  
        {  
            CreateMap<Database.Voters, Models.Voter>();  
            CreateMap<Database.Candidates, Models.Candidate>();  
        }  
    }  
}

**6. VotingAppViewModelList.cs**

using System.ComponentModel.DataAnnotations.Schema;  
using System.ComponentModel.DataAnnotations;  
  
namespace VotingApp.Models  
{  
    public class VotingAppViewModelList  
    {  
        public Voter Voter { get; set; }  
        public Candidate Candidate { get; set; }  
        public List<Voter> VotersList { get; set; }  
        public List<Candidate> CandidatesList { get; set; }  
    }  
  
    public class Voter  
    {  
        public Guid VoterId { get; set; }  
        public string Name { get; set; }  
        public bool HasVoted { get; set; }  
    }  
  
    public class Candidate  
    {  
        public Guid CandidateId { get; set; }  
        public string Name { get; set; }  
        public int Votes { get; set; }  
  
    }  
  
    public class Vote  
    {  
        public Guid VoterId { get; set; }  
        public Guid CandidateId { get; set; }  
    }  
}

**7. Program.cs**

using Microsoft.EntityFrameworkCore;  
using Microsoft.Extensions.Configuration;  
using VotingApp.Database;  
  
var builder = WebApplication.CreateBuilder(args);  
  
// Add services to the container.  
builder.Services.AddControllersWithViews();  
builder.Services.AddDbContext<VoteingDBContext>(  
    options => options.UseSqlServer(  
    builder.Configuration.GetConnectionString("VotingDBConnetionString")  
));  
builder.Services.AddAutoMapper(AppDomain.CurrentDomain.GetAssemblies());  
builder.Services.AddControllersWithViews().AddRazorRuntimeCompilation();  
var app = builder.Build();  
  
// Configure the HTTP request pipeline.  
if (!app.Environment.IsDevelopment())  
{  
    app.UseExceptionHandler("/Voting/Error");  
    // The default HSTS value is 30 days. You may want to change this for production scenarios, see <https://aka.ms/aspnetcore-hsts>

|  |  |
| --- | --- |
|  | [Enforce HTTPS in ASP.NET Core](https://aka.ms/aspnetcore-hsts)  Learn how to require HTTPS/TLS in an ASP.NET Core web app.  aka.ms |

.  
    app.UseHsts();  
}  
  
app.UseHttpsRedirection();  
app.UseStaticFiles();  
  
app.UseRouting();  
  
app.UseAuthorization();  
  
app.MapControllerRoute(  
    name: "default",  
    pattern: "{controller=Voting}/{action=Index}/{id?}");  
  
app.Run();

**8. appsettings.json**

{  
  "ConnectionStrings": {  
    "VotingDBConnetionString": "Data Source=MMD5CG25160YF\\SQLEXPRESS;Integrated Security=true;Initial Catalog=Voting;TrustServerCertificate=True;MultipleActiveResultSets=true;"  
  },  
  "Logging": {  
    "LogLevel": {  
      "Default": "Information",  
      "Microsoft.AspNetCore": "Warning"  
    }  
  },  
  "AllowedHosts": "\*"  
}

**9.  Views/Voting/Index.cshtml**

@model VotingApp.Models.VotingAppViewModelList  
@{  
    ViewData["Title"] = "Voting App";  
    Layout = "~/Views/Shared/\_Layout.cshtml";  
}  
  
<script src="~/lib/jquery/dist/jquery.js"></script>  
<script src="~/lib/bootstrap/dist/js/bootstrap.js"></script>  
<link href="~/lib/bootstrap/dist/css/bootstrap.css" rel="stylesheet" />  
  
<div style="width:100%">  
<div class="table-responsive column">  
    <div>  
        <div style="text-align:center; float:left; width:80%">  
            <h4>  
                Voters  
            </h4>  
        </div>  
        <div style="float:right">  
            <span style="text-align:right;margin-right:20px;">  
                <a href="#" id="btnAddVoter" class="btn btn-outline-primary">+</a>  
            </span>  
        </div>  
    </div>  
   
    <table class="table">  
        <thead>  
            <tr>  
                <th>  
                    @Html.DisplayNameFor(model => model.VotersList.FirstOrDefault().Name)  
                </th>  
                <th>  
                    @Html.DisplayNameFor(model => model.VotersList.FirstOrDefault().HasVoted)  
                </th>  
                 
            </tr>  
        </thead>  
        <tbody>  
            @foreach (var item in Model.VotersList)  
            {  
            <tr>  
                <td>  
                    @Html.DisplayFor(modelItem => item.Name)  
                </td>  
                <td>  
                    @Html.DisplayFor(modelItem => item.HasVoted)  
                </td>  
            </tr>  
            }  
        </tbody>  
    </table>  
</div>  
<div class="table-responsive column">  
    <div>  
        <div style="text-align:center;float:left; width:80%">  
            <h4>  
                Candidates  
            </h4>  
        </div>  
        <div style="float:right">  
            <span style="text-align:right;margin-right:20px;">  
                <a href="#" id="btnAddCandidate" class="btn btn-outline-primary">+</a>  
            </span>  
        </div>  
    </div>  
     
  
    <table class="table">  
        <thead>  
            <tr>  
                <th>  
                    @Html.DisplayNameFor(model => model.CandidatesList.FirstOrDefault().Name)  
                </th>  
                <th>  
                    @Html.DisplayNameFor(model => model.CandidatesList.FirstOrDefault().Votes)  
                </th>  
            </tr>  
        </thead>  
        <tbody>  
            @foreach (var item in Model.CandidatesList)  
            {  
                <tr>  
                    <td>  
                        @Html.DisplayFor(modelItem => item.Name)  
                    </td>  
                    <td>  
                        @Html.DisplayFor(modelItem => item.Votes)  
                    </td>  
                </tr>  
            }  
        </tbody>  
    </table>  
</div>  
</div>  
  
<div class="row" style="width:100%">  
    <form asp-action="SubmitVote" asp-controller="Voting" method="post">  
  
        <div class="row">  
            <div class="col-md-4">  
                <label>I am</label>  
                <select id="drpVoterList" class="form-control" asp-for="Voter.VoterId" asp-items="@(new SelectList(Model.VotersList, "VoterId", "Name",1))">  
                    <option value="">--Select--</option>  
                </select>  
                <input type="hidden" id="hdnVoterId" asp-for="Voter.VoterId" />  
  
            </div>  
            <div class="col-md-4">  
                <label>I vote for</label>  
                <select id="drpCandidateList" class="form-control" asp-for="Candidate.CandidateId" asp-items="@(new SelectList(Model.CandidatesList, "CandidateId", "Name",1))">  
                    <option value="">--Select--</option>  
                </select>  
                <input type="hidden" id="hdnCandidateId" asp-for="Candidate.CandidateId" />  
  
            </div>  
            <div class="col-md-2">  
                <input type="button" id="btnSubmitVote" name="btnSubmit" value="Submit" class="btn btn-success" />  
            </div>  
        </div>  
  
    </form>  
</div>  
  
<div id="MyPopup" class="modal fade" role="dialog">  
    <div class="modal-dialog">  
        <div class="modal-content">  
            <div class="modal-header">  
                <h4 class="modal-title">Add New Voter</h4>  
                <button type="button" id="btnClosePopup" class="close" data-dismiss="modal">  
                    &times;  
                </button>  
            </div>  
            <div class="modal-body">  
                <div class="container">  
                    <div class="row">  
                        <div class="col-md-8">  
                            <form asp-action="AddVoter" asp-controller="Voting">  
                                <div class="form-group">  
                                    Name: <input id="txtName" asp-for="Voter.Name" class="form-control" />  
                                </div>  
                                <br />  
                                <br />  
                                <div class="form-group">  
                                    <a id="btnSubmitVoter" class="btn btn-success btn-sm">Submit</a>  
                                </div>  
                            </form>  
                        </div>  
                    </div>  
                </div>  
            </div>  
            <div class="modal-footer">  
                <input type="button" id="btnClosePopup" value="Close" data-dismiss="modal" class="btn btn-danger" />  
            </div>  
        </div>  
    </div>  
</div>  
   
  
<div id="MyPopupCandidate" class="modal fade" role="dialog">  
    <div class="modal-dialog">  
        <div class="modal-content">  
            <div class="modal-header">  
                <h4 class="modal-title">Add New Candidate</h4>  
                <button type="button" id="btnClosePopup" class="close" data-dismiss="modal">  
                    &times;  
                </button>  
            </div>  
            <div class="modal-body">  
                <div class="container">  
                    <div class="row">  
                        <div class="col-md-8">  
                            <form asp-action="AddCandidate" asp-controller="Voting">  
                                <div class="form-group">  
                                    Name: <input id="txtCandidateName" asp-for="Candidate.Name" class="form-control" />  
                                </div>  
                                <br /><br />  
                                <div class="form-group">  
                                    <a id="btnSubmitCandidate" class="btn btn-success btn-sm">Submit</a>  
                                </div>  
                            </form>  
                        </div>  
                    </div>  
                </div>  
            </div>  
            <div class="modal-footer">  
                <input type="button" id="btnClosePopup" value="Close" data-dismiss="modal" class="btn btn-danger" />  
            </div>  
        </div>  
    </div>  
</div>  
  
  
<script type="text/javascript">  
    $(function () {  
        $("body").on('click', '#btnAddVoter', function () {  
            $('#txtName').val("");  
            $("#MyPopup").modal("show");  
        });  
  
        $("body").on('click', '#btnSubmitVoter', function () {  
            $("#MyPopup").modal("hide");  
            var obj = {};  
            obj.Name = $('#txtName').val();  
  
            $.ajax({  
                url: 'Voting/AddVoter',  
                data: JSON.stringify(obj),  
                type: 'POST',  
                dataType: 'html',  
                contentType: "application/json; charset=utf-8",  
                success: function (response) {  
                    window.location.reload();  
                }  
            });  
        });  
  
        $("body").on('click', '#btnAddCandidate', function () {  
            $('#txtCandidateName').val("");  
            $("#MyPopupCandidate").modal("show");  
        });  
  
        $("body").on('click', '#btnSubmitCandidate', function () {  
            $("#MyPopupCandidate").modal("hide");  
            var obj = {};  
            obj.Name = $('#txtCandidateName').val();  
  
            $.ajax({  
                url: 'Voting/AddCandidate',  
                data: JSON.stringify(obj),  
                type: 'POST',  
                dataType: 'html',  
                contentType: "application/json; charset=utf-8",  
                success: function (response) {  
                    window.location.reload();  
                }  
            });  
        });  
  
        $("body").on('click', '#btnClosePopup', function () {  
            $('#txtName').val("");  
            $('#txtCandidateName').val("");  
  
            $("#MyPopup").modal("hide");  
            $("#MyPopupCandidate").modal("hide");  
        });  
  
        $("body").on('click', '#btnSubmitVote', function () {  
            var obj = {};  
            obj.VoterId = $('#hdnVoterId').val();  
            obj.CandidateId = $('#hdnCandidateId').val();  
  
            $.ajax({  
                url: 'Voting/SubmitVote',  
                data: JSON.stringify(obj),  
                type: 'POST',  
                dataType: 'html',  
                contentType: "application/json; charset=utf-8",  
                success: function (response) {  
                    window.location.reload();  
                }  
            });  
        });  
  
        $("body").on('change', '#drpVoterList', function () {  
            $('#hdnVoterId').val($('#drpVoterList').val());  
        });  
  
        $("body").on('change', '#drpCandidateList', function () {  
            $('#hdnCandidateId').val($('#drpCandidateList').val());  
        });  
    });  
</script>

**10. SQL Database Script**

USE [master]  
GO  
/\*\*\*\*\*\* Object:  Database [Voting]    Script Date: 08-05-2024 11:54:44 \*\*\*\*\*\*/  
CREATE DATABASE [Voting]  
 CONTAINMENT = NONE  
 ON  PRIMARY  
( NAME = N'Voting', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\Voting.mdf' , SIZE = 8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )  
 LOG ON  
( NAME = N'Voting\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\Voting\_log.ldf' , SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )  
 WITH CATALOG\_COLLATION = DATABASE\_DEFAULT  
GO  
ALTER DATABASE [Voting] SET COMPATIBILITY\_LEVEL = 150  
GO  
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))  
begin  
EXEC [Voting].[dbo].[sp\_fulltext\_database] @action = 'enable'  
end  
GO  
ALTER DATABASE [Voting] SET ANSI\_NULL\_DEFAULT OFF  
GO  
ALTER DATABASE [Voting] SET ANSI\_NULLS OFF  
GO  
ALTER DATABASE [Voting] SET ANSI\_PADDING OFF  
GO  
ALTER DATABASE [Voting] SET ANSI\_WARNINGS OFF  
GO  
ALTER DATABASE [Voting] SET ARITHABORT OFF  
GO  
ALTER DATABASE [Voting] SET AUTO\_CLOSE OFF  
GO  
ALTER DATABASE [Voting] SET AUTO\_SHRINK OFF  
GO  
ALTER DATABASE [Voting] SET AUTO\_UPDATE\_STATISTICS ON  
GO  
ALTER DATABASE [Voting] SET CURSOR\_CLOSE\_ON\_COMMIT OFF  
GO  
ALTER DATABASE [Voting] SET CURSOR\_DEFAULT  GLOBAL  
GO  
ALTER DATABASE [Voting] SET CONCAT\_NULL\_YIELDS\_NULL OFF  
GO  
ALTER DATABASE [Voting] SET NUMERIC\_ROUNDABORT OFF  
GO  
ALTER DATABASE [Voting] SET QUOTED\_IDENTIFIER OFF  
GO  
ALTER DATABASE [Voting] SET RECURSIVE\_TRIGGERS OFF  
GO  
ALTER DATABASE [Voting] SET  DISABLE\_BROKER  
GO  
ALTER DATABASE [Voting] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF  
GO  
ALTER DATABASE [Voting] SET DATE\_CORRELATION\_OPTIMIZATION OFF  
GO  
ALTER DATABASE [Voting] SET TRUSTWORTHY OFF  
GO  
ALTER DATABASE [Voting] SET ALLOW\_SNAPSHOT\_ISOLATION OFF  
GO  
ALTER DATABASE [Voting] SET PARAMETERIZATION SIMPLE  
GO  
ALTER DATABASE [Voting] SET READ\_COMMITTED\_SNAPSHOT OFF  
GO  
ALTER DATABASE [Voting] SET HONOR\_BROKER\_PRIORITY OFF  
GO  
ALTER DATABASE [Voting] SET RECOVERY SIMPLE  
GO  
ALTER DATABASE [Voting] SET  MULTI\_USER  
GO  
ALTER DATABASE [Voting] SET PAGE\_VERIFY CHECKSUM    
GO  
ALTER DATABASE [Voting] SET DB\_CHAINING OFF  
GO  
ALTER DATABASE [Voting] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF )  
GO  
ALTER DATABASE [Voting] SET TARGET\_RECOVERY\_TIME = 60 SECONDS  
GO  
ALTER DATABASE [Voting] SET DELAYED\_DURABILITY = DISABLED  
GO  
ALTER DATABASE [Voting] SET ACCELERATED\_DATABASE\_RECOVERY = OFF    
GO  
ALTER DATABASE [Voting] SET QUERY\_STORE = OFF  
GO  
USE [Voting]  
GO  
/\*\*\*\*\*\* Object:  Table [dbo].[Candidates]    Script Date: 08-05-2024 11:54:45 \*\*\*\*\*\*/  
SET ANSI\_NULLS ON  
GO  
SET QUOTED\_IDENTIFIER ON  
GO  
CREATE TABLE [dbo].[Candidates](  
[CandidateId] [uniqueidentifier] NOT NULL,  
[Name] [varchar](50) NOT NULL,  
[Votes] [int] NOT NULL,  
 CONSTRAINT [PK\_Candidates] PRIMARY KEY CLUSTERED  
(  
[CandidateId] ASC  
)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]  
) ON [PRIMARY]  
GO  
/\*\*\*\*\*\* Object:  Table [dbo].[Voters]    Script Date: 08-05-2024 11:54:45 \*\*\*\*\*\*/  
SET ANSI\_NULLS ON  
GO  
SET QUOTED\_IDENTIFIER ON  
GO  
CREATE TABLE [dbo].[Voters](  
[VoterId] [uniqueidentifier] NOT NULL,  
[Name] [varchar](50) NOT NULL,  
[HasVoted] [bit] NOT NULL,  
 CONSTRAINT [PK\_Voters] PRIMARY KEY CLUSTERED  
(  
[VoterId] ASC  
)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]  
) ON [PRIMARY]  
GO  
USE [master]  
GO  
ALTER DATABASE [Voting] SET  READ\_WRITE  
GO

**Preview of VotingApp:**

