

Universidad ORT Uruguay
Facultad de Ingeniería
Escuela de Tecnología

OBLIGATORIO 2 PROGRAMACIÓN 2

DOCUMENTO DE ANÁLISIS



Juan Sánchez – 319487



Pablo Padilla– 292209

N2D

Docente: Luis Dentone

Analista en tecnologías de la información

23-11-2023

Indice

Diagrama de clases	3
Diagrama de casos	4
Evidencia de Testing	5
Casos de prueba: Login	5
Casos de prueba: Registrar	5
Casos de prueba: Administrador	7
Casos de prueba: Miembro.....	8
Código	14
Controladores:	14
Filtros:	21
Views:.....	22
Dirección URL del sitio	32

Diagrama de clases

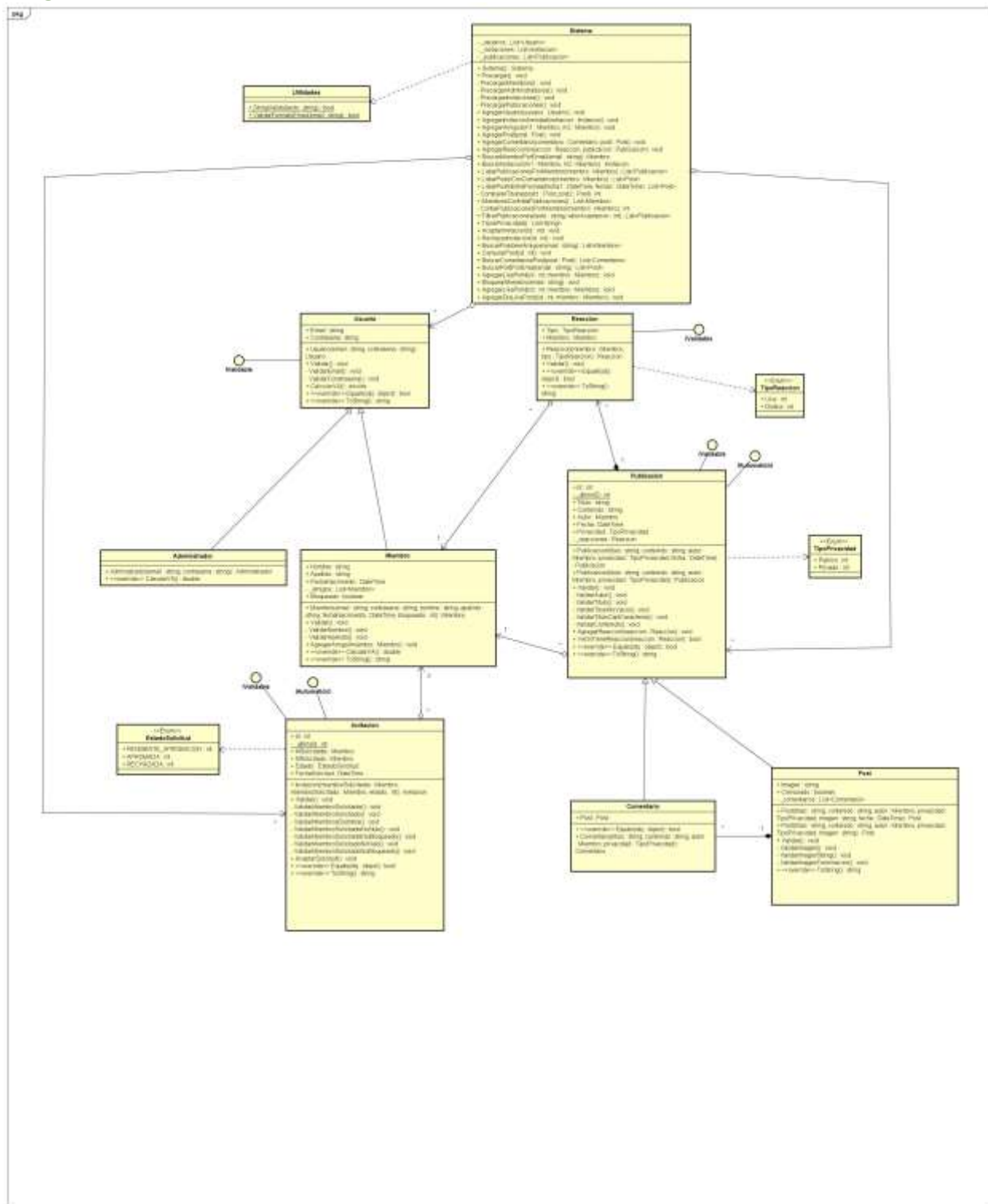
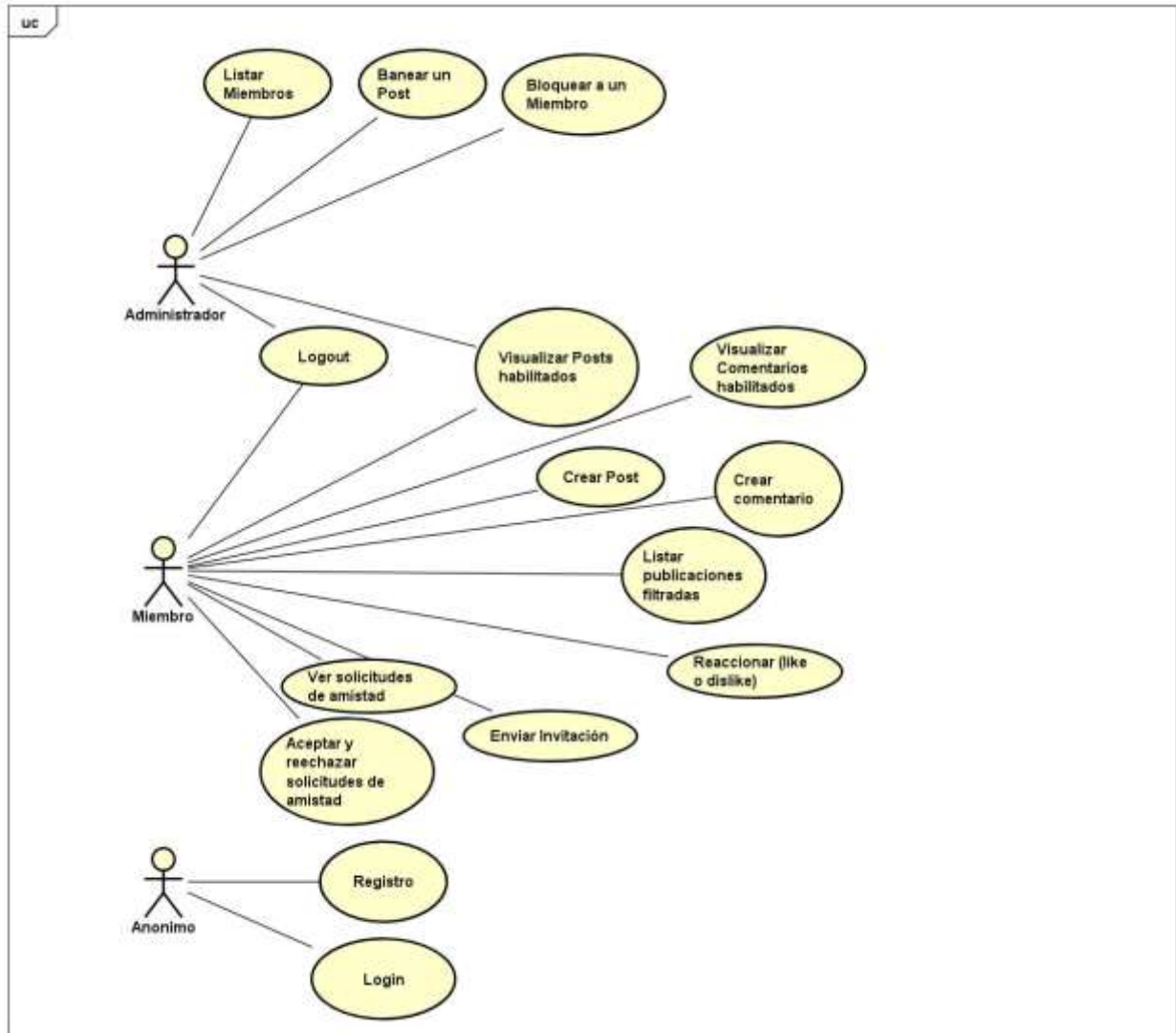


Diagrama de casos



Evidencia de Testing

Casos de prueba: Login

Precondiciones:

No existen precondiciones

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F01-T01	Ingreso a la aplicación exitoso.	Ingresamos mail y contraseña. Clickeamos el botón de "Login".	Mail: email1@gmail.com Contraseña: contrasena1	Se ingresa al sistema y el usuario es redirigido a la página principal según si es administrador o miembro.	Se ingresa al sistema y el usuario es redirigido a la página principal según si es administrador o miembro.	P
F01-T02	Contraseña y/o Usuario incorrecto.	Ingresamos mail y contraseña. Clickeamos el botón de "Login".	Usuario: usu1 Contraseña: aB123	No se logra ingresar al Sistema y se visualiza un mensaje de error "Usuario y/o contraseña incorrecta"	No se logra ingresar al Sistema y se visualiza un mensaje de error "Usuario y/o contraseña incorrecta"	P

Casos de prueba: Registrar

Precondiciones:

No existe precondiciones

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F02-T01	Registro exitoso de nuevo Miembro	Al entrar a la aplicación utilizamos el botón "Registrarse" Ingresamos todos los campos	Nombre: Juan Apellido: Perez Mail: email30@gmail.com	Se registra el nuevo Miembro y se lo redirige a la página de Login.	Se registra el nuevo Miembro y se lo redirige a la página de Login.	P

		correctamente Clickeamos el botón de “Registrarse”.	Fecha de Nacimiento: 01/01/1985 Contraseña: contrasena1			
F02 - T02	Registro fallido – mail duplicado .	Al entrar a la aplicación utilizamos el botón “Registrarse” Ingresamos el campo mail de un usuario ya registrado. Clickeamos el botón de “Registrarse”.	Nombre: Juan Apellido: Perez Mail: email1@gmail.com Fecha de Nacimiento: 01/01/1985 Contraseña: contrasena1	No se registra el nuevo Miembro y se le informa a través de un texto que “El miembro con el email email1@gmail.com ya fué registrado anteriormente.”	No se registra el nuevo Miembro y se le informa a través de un texto que “El miembro con el email email1@gmail.com ya fué registrado anteriormente.”	P
F02 - T03	Registro fallido – campo vacío.	Al entrar a la aplicación utilizamos el botón “Registrarse” Ingresamos un campo vacío. Clickeamos el botón de “Registrarse”.	Nombre: Juan Apellido: Mail: email145@gmail.com Fecha de Nacimiento: 01/01/1985 Contraseña: contrasena1	No se registra el nuevo Miembro y se le informa a través de un texto que “Apellido no puede ser un dato vacío.”	No se registra el nuevo Miembro y se le informa a través de un texto que “Apellido no puede ser un dato vacío.”	P

Casos de prueba: Administrador

Precondiciones:

Administrador logueado.

Email: admin@admin.com

Contraseña: Obligatorio1

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F03-T01	Listar miembros.	Loguearse a la aplicación o ya logueado clicar la opción "Listar Miembros"		Se listan todos los miembros guardados en el Sistema.	Se listan todos los miembros guardados en el Sistema.	P
F03-T02	Bloquear miembros	Loguearse a la aplicación o ya logueado clicar la opción "Listar Miembros" Clicar en el botón "Bloquear" de un miembro no bloqueado.		Se bloquea al Miembro correctamente y se deshabilita el botón.	Se bloquea al Miembro correctamente y se deshabilita el botón.	P
F03-T03	Listar Posts.	Ya logueado clicar la opción "Listar Posts"		Se listan todos los Posts guardados en el Sistema.	Se listan todos los Posts guardados en el Sistema.	P
F03-T04	Censurar Posts	Loguearse a la aplicación y ya logueado clicar la opción "Listar Posts" Clicar en el botón "Censurar" de un Post no bloqueado.		Se bloquea el Post correctamente y se deshabilita el botón.	Se bloquea el Post correctamente y se deshabilita el botón.	P
F03-T05	Logout	Loguearse a la aplicación y ya logueado clicar en la opción Logout.		Se desloguea correctamente y se lo redirige a la página de Login.	Se desloguea correctamente y se lo redirige a la página de Login.	P

Casos de prueba: Miembro

Precondiciones:

Miembro logueado.

Email: email1@gmail.com

Contraseña: contrasena1

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla , P=pasa)
F04 - T01	Listar Publicaciones.	Loguearse a la aplicación o ya logueado clickear la opción "Listar Publicaciones"		Se listan todas las publicaciones públicas, propias o generadas por un amigo.	Se listan todas las publicaciones públicas, propias o generadas por un amigo.	P
F04 - T02	Comentar Post.	Loguearse a la aplicación o ya logueado clickear la opción "Listar Publicaciones" Clickear en el botón "Comentar" de un Post no censurado. Nos redirige hacia "Comentario:" Ingresamos los datos requeridos y clickeamos en el botón "Comentar"	Título: Hola Contenido: Muy buenas	Se comenta el Post correctamente y se lo redirige hacia "Listar publicaciones"	Se comenta el Post correctamente	P
F04 - T03	Comentar Post – Miembro bloqueado	Loguearse a la aplicación o ya logueado clickear la	Email: email13@gmail.com Contraseña: contrasena13	Visualiza mensaje "Este miembro no puede comentar ya	Visualiza mensaje "Este miembro no puede comentar ya	P

		opción “Listar Publicaciones”		que se encuentra bloqueado por un administrador”	que se encuentra bloqueado por un administrador”	
F04 - T04	Comentar Post. – Título menor a 3 caracteres	<p>Loguearse a la aplicación o ya logueado clicar la opción “Listar Publicaciones”</p> <p>Clickear en el botón “Comentar” de un Post no censurado.</p> <p>Nos redirige hacia “Comentario:”</p> <p>No Ingresamos alguno de los datos requeridos y clickeamos en el botón “Comentar”</p>	Título: Ho Contenido:dfgfdgdf g	Se le informa al usuario del error con un texto “El Título no puede contener menos de 3 caracteres” y no se lo redirige ni se crea el comentario.	Se le informa al usuario del error con un texto “El Título no puede contener menos de 3 caracteres” y no se lo redirige ni se crea el comentario.	P
F04 - T05	Reaccionar un Post o Comentario	<p>Loguearse a la aplicación o ya logueado clicar la opción “Listar Publicaciones”</p> <p>Clickear en el botón “Like” o “Dislike” de una publicación (Post o Comentario) no censurado.</p>		Se reacciona a la Publicación deseada y se realiza el cambio en “Valor de aceptación”	Se reacciona a la Publicación deseada y se realiza el cambio en “Valor de aceptación”	P
F04 - T06	Filtrar Publicaciones	Ya logueado clicar la	<p>Texto a buscar: cont</p> <p>Valor de aceptación: 5</p>	Se listan todos los Post y Comentarios que cumplan	Se listan todos los Post y Comentarios que cumplan	P

		<p>opción “Filtrar Publicaciones”</p> <p>Nos redirige hacia “Filtrar publicaciones por texto y valor de aceptación:”</p> <p>Ingresamos los datos requeridos y clickeamos en el botón Filtrar.</p>		con los filtros junto con su “tipo”	con los filtros junto con su “tipo”	
F04 - T07	Crear Post	<p>Ya logueado clickear la opción “Crear Post”</p> <p>Nos redirige hacia “Crear Post”</p> <p>Ingresamos los datos requeridos y clickeamos en el botón Publicar.</p>	<p>Título: hola</p> <p>Contenido: comoestas</p> <p>Imagen: ima.jpg</p> <p>Privacidad: Publico</p>	Se crea el Post correctamente y se lo redirige hacia “Listar publicaciones”	Se crea Post correctamente y se lo redirige hacia “Listar publicaciones”	P
F04 - T08	Crear Post – Miembro bloqueado	<p>Ya logueado clickear la opción “Crear Post”</p> <p>Nos redirige hacia “Crear Post”</p>	<p>Email: email13@gmail.com</p> <p>Contraseña: contrasena13</p>	Se visualiza el mensaje “El usuario no tiene permitido crear Posts porque fué bloqueado por un Administrador ”	Se visualiza el mensaje “El usuario no tiene permitido crear Posts porque fué bloqueado por un Administrador ”	P

F04 - T09	Crear Post – Campo incorrecto	<p>Ya logueado clickear la opción “Crear Post”</p> <p>Nos redirige hacia “Crear Post”</p> <p>Ingresamos los datos requeridos con el formato de imagen erróneo y clickeamos en el botón Publicar.</p>	<p>Título: hola</p> <p>Contenido: comoestas</p> <p>Imagen: imapg</p> <p>Privacidad: Publico</p>	No se crea el Post y se le informa al usuario el error con el texto “La imagen debe terminar en .png o .jpg”.	No se crea el Post y se le informa al usuario el error con el texto “La imagen debe terminar en .png o .jpg”.	P
F04 - T10	Crear Solicitud de Amistad	<p>Ya logueado clickear la opción “Solicitud de amistad”</p> <p>Nos redirige hacia “Solicitud de amistad”</p> <p>Seleccionamo s al Miembro que enviaremos la solicitud de amistad y clickeamos en el botón “Agregar”</p>	Amigo a agregar: Ronaldo Neymar	Se crea la invitación correctamente y se elimina al Miembro solicitado de las opciones a elegir.	Se crea la invitación correctamente y se elimina al Miembro solicitado de las opciones a elegir.	P
F04 - T11	Crear Solicitud de Amistad – Miembro bloqueado	<p>Ya logueado clickear la opción “Solicitud de amistad”</p> <p>Nos redirige hacia</p>		Se visualiza el mensaje “El miembro solicitante fué bloqueado por un administrador”	Se visualiza el mensaje “El miembro solicitante fué bloqueado por un administrador”	P

		"Solicitud de amistad"				
F04 - T12	Ver solicitudes pendientes	<p>Ya logueado clicar la opción "Solicitud de amistad"</p> <p>Nos redirige hacia "Solicitud de amistad"</p> <p>Seleccionamos al Miembro que enviaremos la solicitud de amistad y clickeamos en el botón "Ver solicitudes"</p>		Se listan todas las invitaciones pendientes de aprobación que tenga el usuario, si no tiene ninguna solicitud de amistad se visualizará el mensaje "No tienes solicitudes pendientes"	Se listan todas las invitaciones pendientes de aprobación que tenga el usuario, si no tiene ninguna solicitud de amistad se visualizará el mensaje "No tienes solicitudes pendientes"	P
F04 - T13	Aceptar/Rechazar solicitudes pendientes	<p>Ya logueado clicar la opción "Solicitud de amistad"</p> <p>Nos redirige hacia "Solicitud de amistad"</p> <p>Seleccionamos al Miembro que enviaremos la solicitud de amistad y clickeamos en el botón "Ver solicitudes"</p> <p>Seleccionamos la opción deseada (Aceptar o Rechazar)</p>		Se crea el vínculo de amistad entre los miembros y se elimina la invitación seleccionada de la lista	Se crea el vínculo de amistad entre los miembros y se elimina la invitación seleccionada de la lista	P

F04 - T14	Logout	Loguearse a la aplicación y ya logueado clicar en la opción Logout.		Se desloguea correctamente y se lo redirige a la página de Login.	Se desloguea correctamente y se lo redirige a la página de Login.	P
-----------------	--------	---	--	---	---	---

Código

Controladores:

AdministradorController:

```
using AppWeb.Filtros;
using Dominio;
using Microsoft.AspNetCore.Mvc;

namespace Appweb1.Controllers
{
    [Admin]
    public class AdministradorController : Controller
    {
        private Sistema _sistema = Sistema.Instancia;

        public IActionResult Index()
        {
            try {
                ViewBag.Miembros = _sistema.ListarMiembrosOrdenados();
            } catch (Exception e)
            {
                ViewBag.Error = e.Message;
            }
            return View();
        }

        public IActionResult BloquearMiembro(string email)
        {
            try {
                _sistema.BloquearMiembro(email);
            }
            catch(Exception e) { ViewBag.Error = e.Message; }

            return RedirectToAction("Index");
        }

        public IActionResult ListarPosts()
        {
            try
            {
                ViewBag.Posts = _sistema.ListarPosts();
            }
            catch (Exception e)
            {
                ViewBag.Error = e.Message;
            }
            return View();
        }

        public IActionResult CensurarPost(int id)
        {
            try
            {
                _sistema.CensurarPost(id);
            }
        }
    }
}
```

```

    }
    catch (Exception e) { ViewBag.Error = e.Message; }

    return RedirectToAction("ListarPosts");
}
}
}

```

Login Controller:

```

using Dominio;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using AppWeb.Filtros;

namespace Appweb1.Controllers
{
    public class LoginController : Controller
    {
        private Sistema _sistema = Sistema.Instancia;
        public IActionResult Index()
        {
            return View();
        }

        [HttpPost]
        public IActionResult Index(string email, string contrasena)
        {
            try{
                Usuario usuario = _sistema.BuscarUsuarioPorEmail(email);
                if (usuario == null || contrasena != usuario.Contrasena)
                {
                    ViewBag.Error = "Usuario y/o contraseña incorrecta";
                    return View("Index");
                }

                HttpContext.Session.SetString("email", email);
                HttpContext.Session.SetString("tipo", usuario.GetType().Name);

                if(HttpContext.Session.GetString("tipo") == "Administrador")
                { return Redirect("/Administrador/Index"); }
                else
                {
                    return Redirect("/Miembro/Index");
                }
            }
            catch {
                return View("Index");
            }
        }
    }
}

```

```

    }

    public IActionResult Logout()
    {
        HttpContext.Session.Clear();
        return RedirectToAction("Index");
    }
}
}

```

MiembroController:

```

using Dominio;
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Mvc.Formatters;
using Microsoft.Extensions.Hosting;
using AppWeb.Filtros;

namespace Appweb1.Controllers
{
    [Member]
    public class MiembroController : Controller
    {
        private Sistema _sistema = Sistema.Instancia;

        public IActionResult Index()
        {
            try
            {
                ViewBag.Posts =
                _sistema.BuscarPostPorEmail(HttpContext.Session.GetString("email"));
                string email = HttpContext.Session.GetString("email");
                Miembro autor = _sistema.BuscarMiembroPorEmail(email);
                ViewBag.Bloqueado = autor.Bloqueado;
            }
            catch (Exception e)
            {
                ViewBag.Error = e.Message;
            }
            return View();
        }

        public IActionResult Amigos()
        {
            try
            {
                ViewBag.PosiblesAmigos =
                _sistema.BuscarPosiblesAmigos(HttpContext.Session.GetString("email"));
            }
            catch (Exception e)
            {
                ViewBag.Error = e.Message;
            }
        }
    }
}

```



```

        return View(new Invitacion());
    }
    [HttpPost]
    public IActionResult Amigos(string email)
    {
        try
        {
            Miembro miembro =
            _sistema.BuscarMiembroPorEmail(HttpContext.Session.GetString("email"));
            Miembro nuevoAmigo = _sistema.BuscarMiembroPorEmail(email);
            _sistema.CrearInvitacionAmistad(miembro, nuevoAmigo);

            return RedirectToAction("amigos");
        }
        catch (Exception e)
        {
            ViewBag.Error = e.Message;
        }
        ViewBag.PosiblesAmigos =
        _sistema.BuscarPosiblesAmigos(HttpContext.Session.GetString("email"));
        return View();
    }

    public IActionResult Invitaciones()
    {
        try
        {
            ViewBag.InvitacionesPendientes =
            _sistema.BuscarInvitacionesPendientes(HttpContext.Session.GetString("email"));
        }
        catch (Exception e)
        {
            ViewBag.Error = e.Message;
        }
        return View();
    }

    public IActionResult AceptarInvitacion(int id)
    {
        try
        {
            _sistema.AceptarInvitacion(id);
        }
        catch (Exception e) {
            ViewBag.Error = e.Message;
        }

        return RedirectToAction("Invitaciones");
    }

    public IActionResult RechazarInvitacion(int id)
    {
        try
        {
            _sistema.RechazarInvitacion(id);
        }
    }

```

```

    }
    catch (Exception e) {
        ViewBag.Error = e.Message;
    }

    return RedirectToAction("Invitaciones");
}

public IActionResult CrearPost()
{
    Miembro miembro =
_sistema.BuscarMiembroPorEmail(HttpContext.Session.GetString("email"));
    ViewBag.Bloqueado = miembro.Bloqueado;
    ViewBag.Privacidades = _sistema.TiposPrivacidad();
    return View(new Post());
}

[HttpPost]
public IActionResult CrearPost(Post post)
{
    Miembro miembro =
_sistema.BuscarMiembroPorEmail(HttpContext.Session.GetString("email"));
    if (miembro.Bloqueado == false)
    {
        try {
            string email = HttpContext.Session.GetString("email");
            Miembro autor = _sistema.BuscarMiembroPorEmail(email);
            post.Autor = autor;
            post.Fecha = DateTime.Now;
            post.Censurado = false;
            _sistema.AgregarPost(post);
            ViewBag.Posts =
_sistema.BuscarPostPorEmail(HttpContext.Session.GetString("email"));
            return RedirectToAction("index");
        } catch (Exception e)
        {
            ViewBag.Error = e.Message;
        }
    }
    else
    {
        ViewBag.Error = "Este usuario se encuentra bloqueado por lo tanto no
puede crear Posts";
    }
    ViewBag.Bloqueado = miembro.Bloqueado;
    ViewBag.Privacidades = _sistema.TiposPrivacidad();
    return View(post);
}

public IActionResult CrearComentario(int idPost)
{
    ViewBag.idPost = idPost;
    return View(new Comentario());
}

[HttpPost]

```

```

public IActionResult CrearComentario(Comentario comentario, int idPost) {
    try {
        Post post = _sistema.BuscarPostPorId(idPost);
        Miembro miembro =
_sistema.BuscarMiembroPorEmail(HttpContext.Session.GetString("email"));

        if (miembro.Bloqueado == false && post != null)
        {
            try
            {
                comentario.Privacidad = post.Privacidad;
                comentario.Post = post;
                comentario.Fecha = DateTime.Now;
                comentario.Autor = miembro;
                _sistema.AgregarComentario(comentario, post);
                return RedirectToAction("index");
            }
            catch (Exception e)
            {
                ViewBag.Error = e.Message;
            }
        }
    }
    catch (Exception e)
    {
        ViewBag.Error = e.Message;
    }
    ViewBag.idPost = idPost;
    return View(comentario);
}

public IActionResult DarLike(int id)
{
    Miembro miembro =
_sistema.BuscarMiembroPorEmail(HttpContext.Session.GetString("email"));

    _sistema.AgregarLikePorId(id, miembro);

    ViewBag.Posts =
_sistema.BuscarPostPorEmail(HttpContext.Session.GetString("email"));
    string email = HttpContext.Session.GetString("email");
    Miembro autor = _sistema.BuscarMiembroPorEmail(email);
    ViewBag.Bloqueado = autor.Bloqueado;

    return View("Index");
}

public IActionResult DarDisLike(int id)
{
    Miembro miembro =
_sistema.BuscarMiembroPorEmail(HttpContext.Session.GetString("email"));

```

```

        _sistema.AgregarDisLikePorId(id, miembro);

        ViewBag.Posts =
_sistema.BuscarPostPorEmail(HttpContext.Session.GetString("email"));
        string email = HttpContext.Session.GetString("email");
        Miembro autor = _sistema.BuscarMiembroPorEmail(email);
        ViewBag.Bloqueado = autor.Bloqueado;

        return View("Index");
    }

    public IActionResult FiltrarPublicaciones() {

        return View();
    }

    [HttpPost]
    public IActionResult FiltrarPublicaciones(string texto, int valorAceptacion)
    {
        ViewBag.PublicacionesFiltradas = _sistema.FiltrarPublicaciones(texto,
valorAceptacion);

        return View();
    }
}
}

```

RegistroController:

```

using Dominio;
using Microsoft.AspNetCore.Mvc;
using AppWeb.Filtros;

namespace Appweb1.Controllers
{
    public class RegistroController : Controller
    {
        private Sistema _sistema = Sistema.Instancia;
        public IActionResult Index()
        {

            return View(new Miembro());
        }

        [HttpPost]
        public IActionResult Index(Miembro miembro)
        {
            try
            {
                _sistema.AgregarMiembro(miembro);
                return Redirect("/Login/Index");
            }
            catch (Exception e)
            {
                ViewBag.Error = e.Message;
            }
            return View(miembro);
        }
    }
}

```

```

    }
}

```

Filtros:

Administrador:

```

using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Mvc.Filters;
namespace AppWeb.Filtros
{
    public class Admin : Attribute, IAuthorizationFilter
    {
        public void OnAuthorization(AuthorizationFilterContext context)
        {
            string rol = context.HttpContext.Session.GetString("tipo");
            if (string.IsNullOrEmpty(rol))
            {
                context.Result = new RedirectResult("/Login/Index");
                return;
            }
            if (rol != "Administrador")
            {
                context.Result = new RedirectResult("/Miembro/Index");
                return;
            }
        }
    }
}

```

Miembro:

```

using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Mvc.Filters;
namespace AppWeb.Filtros
{
    public class Member : Attribute, IAuthorizationFilter
    {
        public void OnAuthorization(AuthorizationFilterContext context)
        {
            string rol = context.HttpContext.Session.GetString("tipo");
            if (string.IsNullOrEmpty(rol))
            {
                context.Result = new RedirectResult("/Login/Index");
                return;
            }
            if (rol != "Miembro")
            {
                context.Result = new RedirectResult("/Administrador/Index");
                return;
            }
        }
    }
}

```

```
    }  
}
```

Views:

Administrador:

/Index

```
@using Dominio  
@model Miembro  
@*
```

For more information on enabling MVC for empty projects, visit
<https://go.microsoft.com/fwlink/?LinkID=397860>

```
*@  
@{  
}
```

@ViewBag.error

```
<table class="table table-striped">  
    <thead>  
        <tr>  
            <th scope="col">Email</th>  
            <th scope="col">Nombre</th>  
            <th scope="col">Apellido</th>  
            <th scope="col">Fec.Nacimiento</th>  
            <th scope="col">Amigos</th>  
            <th scope="col">Bloqueado</th>  
            <th scope="col">Bloquear Miembro</th>  
        </tr>  
    </thead>  
    <tbody>  
        @foreach (Miembro miembro in ViewBag.Miembros)  
        {  
            <tr>  
                <th scope="row">@miembro.Email</th>  
                <td>@miembro.Nombre</td>  
                <td>@miembro.Apellido</td>  
                <td>@miembro.FechaNacimiento.ToShortDateString()</td>  
                <td>@miembro.Amigos.Count</td>  
                <td>@miembro.Bloqueado</td>  
  
                <td> @if (miembro.Bloqueado == false)  
                {  
                    <a class="btn btn-danger"  
href="/Administrador/BloquearMiembro?email=@miembro.Email">Bloquear</a>  
                }  
                </td>  
            </tr>  
        }  
    </tbody>  
</table>
```

/ListarPosts

```

@using Dominio
@model Post
@*
    For more information on enabling MVC for empty projects, visit
    https://go.microsoft.com/fwlink/?LinkID=397860
*@
@{
}
@ViewBag.error

```

```

<table class="table table-striped">
    <thead>
        <tr>
            <th scope="col">ID</th>
            <th scope="col">Título</th>
            <th scope="col">Contenido</th>
            <th scope="col">Autor</th>
            <th scope="col">Privacidad</th>
            <th scope="col">Imagen</th>
            <th scope="col">Fecha</th>
            <th scope="col">Censurado</th>
            <th scope="col">Censurar Post</th>
        </tr>
    </thead>
    <tbody>
        @foreach (Post post in ViewBag.Posts)
        {
            <tr>
                <th scope="row">@post.Id</th>
                <td scope="row">@post.Título</td>
                <td>@post.Contenido</td>
                <td>@post.Autor.Nombre</td>
                <td>@post.Privacidad</td>
                <td>@post.Imagen</td>
                <td>@post.Fecha.ToShortDateString()</td>
                <td>@post.Censurado</td>

                <td>
                    @if (post.Censurado == false)
                    {
                        <a class="btn btn-danger"
href="/Administrador/CensurarPost?id=@post.Id">Censurar</a>
                    }
                </td>
            </tr>
        }
    </tbody>
</table>

```

Login

/Index

```
@using Dominio
```

```
@model Usuario
```

```
@*
```

```
    For more information on enabling MVC for empty projects, visit  
https://go.microsoft.com/fwlink/?LinkID=397860
```

```
*@
```

```
@{
```

```
}
```

```
@ViewBag.Error
```

```
<form action="/Login/Index" method="post">  
    <div class="container ">  
        <h1 >Login:</h1>  
        <div class="mb-3">  
            <input type="email" name="email" placeholder="Ingrese el email" />  
        </div>  
        <div class="mb-3">  
            <input type="password" name="contrasena" placeholder="Ingrese la contraseña" />  
        </div>  
        <div class="mb-3">  
            <button type="submit" class="btn btn-primary">Login</button>  
        </div>  
        <div class="mb-3">  
        <a class="btn btn-outline-info" href='/Registro/Index'>Registrarse</a>  
        </div>  
        </div>  
</form>
```

Miembro

/Amigos

```
@using Dominio
```

```
@model Invitacion
```

```
@*
```

```
    For more information on enabling MVC for empty projects, visit  
https://go.microsoft.com/fwlink/?LinkID=397860
```

```
*@
```

```
@{
```

```
}
```

```
<h2>Agregar Amigo</h2>
```

```
@ViewBag.error
```

```
@if (ViewBag.PosiblesAmigos != null)
```

```
{
```

```
    <p>Seleccione a quién quiere agregar como amigo:</p>
```

```
    <form action="/Miembro/Amigos" method="post">
```

```
        <select name="email">
```

```
            <option value="" disabled selected>Selecciona una opción</option>
```



```

        @foreach (Miembro miembro in ViewBag.PosiblesAmigos)
        {
            <option value="@miembro.Email">@miembro.Nombre
@miembro.Apellido</option>
        }

        </select>
        <button type="submit">Agregar</button>
    </form>
}
<br>
<h2>Solicitudes Pendientes</h2>
<a class="btn btn-outline-info" href="/Miembro/Invitaciones">Ver solicitudes</a>

```

```

/CrearComentario
@using Dominio
@model Comentario
@*
    For more information on enabling MVC for empty projects, visit
    https://go.microsoft.com/fwlink/?LinkID=397860
*@
@{
}

@ViewBag.error
<h2>Comentario:</h2>
<form action="/Miembro/CrearComentario" method="post">
    <h4 style="margin-top:70px">Comentar:</h4>
    <input type="text" name="idPost" hidden value="@ViewBag.idPost" />
    <input type="text" name="titulo" placeholder="Titulo"
value="@Model.Titulo"></input>
    <br>
    <div class="form-floating">
        <input type="text" placeholder="Contenido" name="contenido"
style="width:400px; margin: 30px 0" value="@Model.Contenido"></input>
    </div>
    <br>
    <button type="submit">Comentar</button>
</form>

```

/CrearPost

```

@using Dominio
@model Post
@*
    For more information on enabling MVC for empty projects, visit
    https://go.microsoft.com/fwlink/?LinkID=397860
*@
@{
}
@ViewBag.error

<h2>Crear Post</h2>
@if (ViewBag.Bloqueado)
{

```

```

        <p>El usuario no tiene permitido crear Posts porque fué bloqueado por un
Administrador</p>
    }
    else
    {
        <form action="/Miembro/CrearPost" method="post">

            <input type="text" name="titulo" placeholder="Titulo" required
value="@Model.Titulo"></input>
            <br>
            <div class="form-floating">
                <input type="text" placeholder="Contenido" name="Contenido"
style="width:600px; margin: 30px 0" required value="@Model.Contenido"></input>
            </div>
            <br>
            <input type="text" name="Imagen" placeholder="Imagen formato .jpg o .png"
required value="@Model.Imagen"></input>
            <br>
            <select name="Privacidad">
                @foreach (string privacidad in ViewBag.Privacidades)
                {
                    <option value="@privacidad"> @privacidad </option>
                }

            </select>
            <button type="submit">Publicar Post </button>
        </form>
    }

```

```

/FiltrarPublicaciones
@using Dominio

```

```

@*
    For more information on enabling MVC for empty projects, visit
https://go.microsoft.com/fwlink/?LinkID=397860
*@
@{
}

```

```
@ViewBag.Error
```

```

<h2>Filtrar publicaciones por texto y valor de aceptación</h2>
<form action="/Miembro/FiltrarPublicaciones" method="post">

    <input type="text" name="texto" placeholder="Texto a buscar" required></input>
    <br>
    <br>
    <input type="number" name="ValorAceptacion" placeholder="Ingresa un valor de
aceptación mínimo" required styles="width:auto"></input>
    <br>

    <button type="submit" style="margin-bottom:20px">Filtrar </button>
</form>
@if (ViewBag.PublicacionesFiltradas != null)
{
    @foreach (Publicacion publicacion in ViewBag.PublicacionesFiltradas)
    {

```

```

        <div style="margin-bottom:250px; padding:50px " class="border border-1
container shadow-lg p-3 mb-5 bg-body-tertiary rounded">
            <h3 style=" font-size:25px; margin-bottom:10px">@publicacion.Titulo</h3>
            <p>Tipo: @publicacion.GetType().Name </p>
            <p>Autor: @publicacion.Autor.Nombre</p>
            <p>Fecha de publicación: @publicacion.Fecha.ToShortDateString() </p>
            <p style="font-weight:600">@publicacion.Contenido</p>
            <p>Likes: @publicacion.CantidadLikes</p>
            <button> <a href="/Miembro/DarLike?id=@publicacion.Id" style="text-
decoration: none; color: black">Dar Like</a> </button>
            <p>Dislikes: @publicacion.CantidadDislikes</p>
            <button><a href="/Miembro/DarDislike?id=@publicacion.Id" style="text-
decoration: none; color: black">Dar Dislike</a></button>
            <p>Valor de aceptación: @publicacion.calcularVA</p>
        </div>
    }
}

```

```

/Index
@using Dominio
@model Post

```

```

@*
    For more information on enabling MVC for empty projects, visit
    https://go.microsoft.com/fwlink/?LinkID=397860
*@
@{
}

```

```
@ViewBag.error
```

```
<h2 style="margin-bottom:100px; font-size:32px; margin-top:70px" class="text-center">Posts</h2>
```

```

@foreach (Post post in ViewBag.Posts)
{
    <div style="margin-bottom:250px; padding:50px " class="border border-1 container
shadow-lg p-3 mb-5 bg-body-tertiary rounded">
        <h3 style=" font-size:25px; margin-bottom:10px">@post.Titulo</h3>
        <p>Autor: @post.Autor.Nombre</p>
        <p>Fecha de publicación: @post.Fecha.ToShortDateString() </p>
        <p style="font-weight:600">@post.Contenido</p>
        <p>Likes: @post.CantidadLikes</p>
        <a href="/Miembro/DarLike?id=@post.Id" class="btn btn-success">Dar Like</a>
        <p>Dislikes: @post.CantidadDislikes</p>
        <a href="/Miembro/DarDislike?id=@post.Id" class="btn btn-danger">Dar
Dislike</a>
        <p>Valor de aceptación: @post.calcularVA</p>
        <hr>

        <h4 style="margin-top:70px" class="text-center">Comentarios</h4>
        @if (post.Comentarios.Count() != 0)
        {
            @foreach (Comentario comentario in post.Comentarios)
            {
                <h5 style="margin-top:50px; margin-
bottom:25px">@comentario.Titulo</h5>
                <p> Autor: @comentario.Autor.Nombre</p>
            }
        }
    </div>
}

```

```

        <p style="font-weight:600">@comentario.Contenido</p>
        <p>Likes: @comentario.CantidadLikes()</p>
        <a href="/Miembro/DarLike?id=@comentario.Id" class="btn btn-
success">Dar Like</a>
        <p>Dislikes: @comentario.CantidadDislikes()</p>
        <a href="/Miembro/DarDislike?id=@comentario.Id" class="btn btn-
danger">Dar Dislike</a>
        <p>Valor de aceptación: @comentario.calcularVA()</p>
        <hr>
    }
}
else
{
    <p>Este post todavia no tiene comentarios</p>
}

@if (ViewBag.Bloqueado == true)
{
    <p>Este miembro no puede comentar ya que se encuentra bloqueado por un
administrador</p>
}
else
{
    <a href="/Miembro/CrearComentario?idPost=@post.Id" class="btn btn-
outline-primary">Comentar</a>
}
</div>
}

```

```

/Invitaciones
@using Dominio
@model Invitacion
@*

```

For more information on enabling MVC for empty projects, visit
<https://go.microsoft.com/fwlink/?LinkID=397860>

```

*@
@{
}

```

```
@ViewBag.error
```

```

<table class="table table-striped">
    <thead>
        <tr>
            <th scope="col">Solicitante</th>
            <th scope="col">Aceptar Solicitud</th>
            <th scope="col">Rechazar Solicitud</th>
        </tr>
    </thead>
    <tbody>
        @if(ViewBag.InvitacionesPendientes.Count==0){
            <tr>
                <td>No tienes solicitudes pendientes
            </td>
            <td></td>
            <td></td>
        }
    </tbody>
</table>

```

```

        </tr>
    }
    else{

        @foreach (Invitacion invitacion in ViewBag.InvitacionesPendientes)
        {
            <tr>
                <th scope="row">@invitacion.MiembroSolicitante.Nombre
@invitacion.MiembroSolicitante.Apellido</th>
                <td>
                    <a class="link-secondary"
href="/Miembro/AceptarInvitacion?id=@invitacion.Id">Aceptar</a>
                </td>
                <td>
                    <a class="link-secondary"
href="/Miembro/RechazarInvitacion?id=@invitacion.Id">Rechazar</a>
                </td>
            </tr>
        }
    }
</tbody>
</table>

```

Registro

/Index

```

@using Dominio
@model Miembro
@*
    For more information on enabling MVC for empty projects, visit
    https://go.microsoft.com/fwlink/?LinkID=397860
*@
@{
}
@ViewBag.Error

<form action="/Registro/Index" method="post">
    <h1>Registro:</h1>
    <input type="text" name="Nombre" placeholder="Ingrese su nombre"
value="@Model.Nombre" />
    <input type="text" name="Apellido" placeholder="Ingrese su apellido"
value="@Model.Apellido" />
    <input type="email" name="Email" placeholder="Ingrese su email"
value="@Model.Email" />
    <input type="date" placeholder="Ingrese fecha nacimiento" name="FechaNacimiento"
value="@Model.FechaNacimiento.ToString("dd-mm-yyyy")" />
    <input type="password" name="Contraseña" placeholder="Ingrese la contraseña" />
    <button type="submit">Registrarse</button>
</form>
<br>
<button> <a style="text-decoration: none; color: black" href='/Login/Index'>Volver
al Login</a> </button>

```

Shared

/_Layout

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>@ViewData["Title"] - Appweb1</title>
  <link rel="stylesheet" href="~/lib/bootstrap/dist/css/bootstrap.min.css" />
  <link rel="stylesheet" href="~/css/site.css" asp-append-version="true" />
  <link rel="stylesheet" href="~/Appweb1.styles.css" asp-append-version="true" />
</head>
<body>
  <header>
    <nav class="navbar navbar-expand-sm navbar-toggleable-sm navbar-light
border-bottom box-shadow mb-3" style="background-color:#ccea8">
      <div class="container-fluid">
        <h1>Obligatorio 2</h1>
        <button class="navbar-toggler" type="button" data-bs-
toggle="collapse" data-bs-target=".navbar-collapse" aria-
controls="navbarSupportedContent"
          aria-expanded="false" aria-label="Toggle navigation">
          <span class="navbar-toggler-icon"></span>
        </button>
        <div class="navbar-collapse collapse d-sm-inline-flex justify-
content-between">
          <ul class="navbar-nav flex-grow-1">
            <@if (@Context.Session.GetString("tipo") == "Administrador")>
            {
              <li class="nav-item">
                <a class="nav-link text-dark" asp-area="" asp-
controller="Administrador" asp-action="Index">Listar Miembros</a>
              </li>
              <li class="nav-item">
                <a class="nav-link text-dark" asp-area="" asp-
controller="Administrador" asp-action="ListarPosts">Listar Posts</a>
              </li>
            }
            <@if (@Context.Session.GetString("tipo") == "Miembro")>
            {
              <li class="nav-item">
                <a class="nav-link text-dark" asp-area="" asp-
controller="Miembro" asp-action="Index">Listar Publicaciones</a>
              </li>
              <li class="nav-item">
                <a class="nav-link text-dark" asp-area="" asp-
controller="Miembro" asp-action="FiltrarPublicaciones">Filtrar Publicaciones</a>
              </li>
              <li class="nav-item">
                <a class="nav-link text-dark" asp-area="" asp-
controller="Miembro" asp-action="CrearPost">Crear post</a>
              </li>
              <li class="nav-item">
                <a class="nav-link text-dark" asp-area="" asp-
controller="Miembro" asp-action="Amigos">Solicitudes de amistad</a>
              </li>
            }
          </ul>
        </div>
      </div>
    </nav>
  </header>
</body>
</html>
```

```

    }
    @if (@Context.Session.GetString("tipo") != null)
    {
        <li class="nav-item">
            <a class="nav-link text-dark" asp-area="" asp-
controller="Login" asp-action="Logout">Logout</a>
        </li>
    }
    </ul>
</div>
</div>
</nav>
</header>
<div class="container">
    <main role="main" class="pb-3">
        @RenderBody()
    </main>
</div>

<footer class="border-top footer text-muted">
    <div class="container">
        <p>Juan Sánchez, Pablo Padilla</p>
    </div>
</footer>
<script src="~/lib/jquery/dist/jquery.min.js"></script>
<script src="~/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
<script src="~/js/site.js" asp-append-version="true"></script>
@await RenderSectionAsync("Scripts", required: false)
</body>
</html>

```

Dirección URL del sitio

<http://obligatorio2sanchezpadilla.somee.com/>