UNIVERSIDAD TECNOLOGICA DE PUEBLA

TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN

QUINTO\_CUATRIMESTRE\_GRUPO\_A\_

**PROGRAMACIÓN DE APLICACIONES**

SALAS SOLIS J. CRUZ

ABRAHAM LORANCA ROSAS

GONZÁLEZ ROMERO JESÚS EDGAR

PAREDES GALVÁN ALEJANDRO

GONZÁLEZ REYES IDANIA

BERNARDO DOMÍNGUEZ CLAUDIA

**MANUAL TÉCNICO**

H\_PUEBLA\_DE\_ZARAGOZA\_A\_24\_DE\_ABRIL\_DE\_2017

INTRODUCCIÓN

La implementación del sistema automatizado para diplomas, se basa en una adaptación para los cursos que seran implementados y estos quieran entregar un certificado en el cual haga constar que el alumno ha concluido ese curso.

El software tambien tendrá la posibilidad de imprimir gafetes, los cuales serán utilizados para conferencias futuras; estos gafetes serán personalizados ya que llevaran el nombre de quien esté invitado al evento.

Este manual describe los pasos necesarios para cualquier persona que tenga ciertas bases de sistemas y pueda realizar un manejo adecuado de la página web creada, la cual está conectada a una base de datos remota. Se describen puntos como son la codiicación de clases, eventos y páginas maestras.

Es importante tener en cuenta que en el presente manual se hace mención a las especificaciones en las que se trabajó con Visual Studio.

INDICE

Contenido

[Alcance del problema 4](#_Toc480728073)

[ESPECIFICACIONES TECNICAS 4](#_Toc480728074)

[DICCIONARIO DE DATOS 5](#_Toc480728075)

[función de cada componente de la aplicación: 6](#_Toc480728076)

[Funcionamiento del sistema login 10](#_Toc480728077)

[Funcionamiento de la página Modificar 11](#_Toc480728078)

[Funcionamiento de la página Eliminar 12](#_Toc480728079)

[Funcionamiento de la pre visualización 13](#_Toc480728080)

[Funcionamiento de la página Enviar Diplomas 14](#_Toc480728081)

[Funcionamiento Método Enviar correos 15](#_Toc480728082)

[Funcionamiento página de Registro 16](#_Toc480728083)

[Funcionamiento página Gafete 17](#_Toc480728084)

[Mapa de navegación 18](#_Toc480728085)

[Descripción del mapa de navegación 19](#_Toc480728086)

[Base de datos 21](#_Toc480728087)

# Alcance del problema

El alcance del presente problema se atiene al desarrollo de una página web, que llevará el historial de visitas, así como el control y registro de los alumnos y/o público que asista. La página web se aplicará a público en general y está a su vez tendrá un administrador que lleve a cabo los cambios de los diplomas y gafetes que se enviarán a través del correo electrónico. La metodología propuesta está orientada al desarrollo de un sistema de Administración y registro de visitas, cursos y otras actividades; fundamentada en estructura de control interno.

# ESPECIFICACIONES TECNICAS

**Sistema operativo:** Microsoft 7 o versiones superiores

**Lenguaje de programación:** PHP, HTML 5, CSS 3, Framework (Bootstrap), JavaScript.

**Servidor de aplicaciones:** Servidor web Apache 2.0

**Navegador Web**: Cualquier versión de Google Chrome (Para trabajar en la aplicación) y funciona en todos los exploradores web.

**Plataforma utilizada:** Visual Studio 2015

# DICCIONARIO DE DATOS

Los datos a utilizar se manejaran en las tablas ya existentes de la base de datos, los cuales son “alumnos” y “universidades”

Estructura de la tabla alumnos:

|  |  |
| --- | --- |
| **Campo** | **Tipo** |
| Nombre | Varchar |
| ApellidoP | Varchar |
| ApellidoM | Varchar |
| CorreoEle | Varchar |
| FechaNacimeinto | Date |
| Curso | Varchar |
| Horario | Varchar |
| Universidad\_id | Int |
| Carrera | Varchar |
| Folio\_alumno | Int |

Estructura de la tabla Universidades:

|  |  |
| --- | --- |
| **Campo** | **Tipo** |
| Nombre | Varchar |
| Direccion | Varchar |
| Nomenclatura | Varchar |
| Id\_Universidad | Int |

A continuación, se describe brevemente la

Función de cada componente de la aplicación:









# Funcionamiento del sistema login

En el método de autenticar se pasarán dos parámetros el de contraseña y el de usuario cuando el usuario intente accesar el sistema mandará los dos datos ingresados por consulta para verificar si hay coincidencias entre usuario y contraseña, si encuentra un registro lo contará y al regresar un valor lo devolverá true así permitiendo el acceso a al panel de administrador

public bool Autenticar(string usuario, string password)

{

Diplomas.Models.Conexion Insertar = new Diplomas.Models.Conexion();

//consulta a la base de datos

string sql = @"SELECT COUNT(\*) FROM Administrador WHERE Usuario = @user AND Contra = @pass";

//cadena conexion

Insertar.AbrirConexion();

SqlCommand cmd = new SqlCommand(sql, Insertar.carretera); //ejecutamos la instruccion

cmd.Parameters.AddWithValue("@user", usuario); //enviamos los parametros

cmd.Parameters.AddWithValue("@pass", password);

int count = Convert.ToInt32(cmd.ExecuteScalar()); //devuelve la fila afectada

if (count == 0)

return false;

else

return true;

}

# Funcionamiento de la página Modificar

La modificación se creará a partir de buscar un registro por su id nos desplegara la información de la persona en textbox para modificarla cuando se modifique algún registro el botón llamará al método y mediante una consulta update tomará todos los datos de los textbox y los enviará a la base de datos

if (selecciona.AbrirConexion())

{

adapter = selecciona.construye\_adapter("select \* from Persona where IdPersona = @IdPersona");

adapter.SelectCommand.Parameters.Add(new SqlParameter("@IdPersona", SqlDbType.Int));

adapter.SelectCommand.Parameters["@IdPersona"].Value = txtId.Text;

fila = selecciona.extrae\_registro(adapter);

if (fila != null)

{

txtnom.Text = fila["Nombre"].ToString();

txtapp.Text = fila["App"].ToString();

txtapm.Text = fila["Apm"].ToString();

txtFec.Text = fila["FechaN"].ToString();

txtOc.Text = fila["Ocupacion"].ToString();

txtIdE.Text = fila["IdEve"].ToString();

txtCor.Text = fila["Correo"].ToString();

}

else Response.Write("Error al visualizar el registro");

}

else Response.Write("Error al conectar a la base de datos");

}

# Funcionamiento de la página Eliminar

El botón de eliminar llama al método y por medio de su id eliminara el registro de la base de datos

protected void btnEli\_Click(object sender, EventArgs e)

{

SqlCommand MyCommand;

Diplomas.Models.Conexion elimina = new Diplomas.Models.Conexion();

if (txtId.Text != "")

{

if (elimina.AbrirConexion())

{

MyCommand = elimina.construye\_command("Delete from Persona where IdPersona = @IdPersona");

MyCommand.Parameters.Add(new SqlParameter("@IdPersona", SqlDbType.Int, 10));

MyCommand.Parameters["@IdPersona"].Value = txtId.Text;

if ((elimina.ejecutanonquery()) != 0)

{

Response.Write("Registro Eliminado");

}

else

Response.Write("No se afectaron registros");

MyCommand.Connection.Close();

elimina.desconectar();

}

}

# Funcionamiento de la pre visualización

Al momento de dar en examinar nos desplegará una ventana de Windows en la cual seleccionaremos el diseño del diploma en pdf lo seleccionaremos y aparecerá en un embed el archivo deseado

protected void Button1\_Click(object sender, EventArgs e)

{

if (FileUpload1.HasFile)

{

String ext = System.IO.Path.GetExtension(FileUpload1.FileName);

if (ext == ".pdf")

{

String path = Server.MapPath("../pdf//");

FileUpload1.SaveAs(path + "diploma.pdf");

}

else

{

Response.Write("<h3> porfavor selecciona un PDF");

}

}

else

{

Response.Write("<h3> porfavor selecciona un archivo");

}

}

# Funcionamiento de la página Enviar Diplomas

Al seleccionar el botón de enviar diploma manda a llamar dos arreglos los cuales contienen los datos de la base de datos de la persona que es correo a donde se enviara y nombre que será el nombre que se plasme en el pdf después en el código crea variables con las que se llenara el correo que se le enviara a la persona después se le da la ubicación del archivo que se utilizara y posteriormente la dirección del archivo modificado, se le da loa parámetros de tipo de letra y tamaño coordenadas donde se plasmara el nombre, al final se llama al método para enviar los correos

protected void Enviar\_Click(object sender, EventArgs e)

{

Diplomas.Models.Conexion con = new Diplomas.Models.Conexion();

int pos = con.repos();

string[] nombres = new string[20];

string[] correo = new string[20];

nombres = con.devolvernombre();

correo = con.devolvercorreo();

for(int i = 0; i < pos; i++) {

string email, asunto, mensaje;

email = correo[i];

asunto = "Diploma";

mensaje = "Felicidades";

string ruta=Server.MapPath("~/pdf/diplomas.pdf");

String pathin = MapPath("~/pdf/diploma.pdf");

String pathout = MapPath("~/pdf/diplomas.pdf");

var doc = new Document(PageSize.A4);

PdfReader reader = new PdfReader(pathin);

reader.SelectPages("1");

PdfStamper stamper = new PdfStamper(reader, new FileStream(pathout, FileMode.Create));

PdfContentByte pbover = stamper.GetOverContent(1);

Font letra = new Font(Font.FontFamily.TIMES\_ROMAN,Convert.ToInt16(TextBox3.Text));

ColumnText.ShowTextAligned(pbover, Element.ALIGN\_JUSTIFIED, new Phrase(Convert.ToInt16(TextBox3.Text), nombres[i], letra), Convert.ToInt16(TextBox1.Text), Convert.ToInt16(TextBox2.Text), 0);

stamper.Close();

reader.Dispose();

Diplomas.Models.Citas diplomaenvio = new Diplomas.Models.Citas(email, asunto, mensaje, ruta);

}

}

# Funcionamiento Método Enviar correos

El método para enviar los correos se utiliza la librería de .net email la cual nos proporciona todo para enviar los correos se crea el objeto mail para así ir llenando los campos se utiliza el host de google el puerto 587

public Citas(String destinatario, String asunto, String mensaje, string ruta)

{

MailMessage Correo = new MailMessage();

SmtpClient protocolo = new SmtpClient();

Correo.To.Add(destinatario);

Correo.From = new MailAddress("cabresto105@gmail.com", "Abraham Loranca Rosas", System.Text.Encoding.UTF8);

Correo.Subject = asunto;

Correo.SubjectEncoding = System.Text.Encoding.UTF8;

Correo.Body = mensaje;

Correo.BodyEncoding = System.Text.Encoding.UTF8;

Correo.IsBodyHtml = false;

if (ruta.Equals("") == false)

{

System.Net.Mail.Attachment archivo = new System.Net.Mail.Attachment(ruta);

Correo.Attachments.Add(archivo);

}

protocolo.Credentials = new System.Net.NetworkCredential("cabresto105@gmail.com", "prueba123456");

protocolo.Port = 587;

protocolo.Host = "smtp.gmail.com";

protocolo.EnableSsl = true;

try

{

protocolo.Send(Correo);

protocolo.Dispose();

Correo.Dispose();

}

catch (SmtpException error)

{

estado = false;

merror = error.Message.ToString();

}

} }

# Funcionamiento página de Registro

en la página de registro tenemos lo que son varios textbox con lo que el usuario podrá registrarse y enviar la información a la base de datos

protected void Button1\_Click(object sender, EventArgs e)

{

string nombres = nombre.Text+" "+apellidop.Text+" "+ apellidom.Text;

Diplomas.Models.Conexion registro = new Diplomas.Models.Conexion();

DateTime fecha = Convert.ToDateTime(date.Text);

registro.Altas(Convert.ToInt32( curp.Text),nombre.Text, apellidop.Text, apellidom.Text,fecha,ocupacion.Text,Convert.ToInt32(idevento.Text),email.Text);

registro.insertnombrecorreo(nombres, email.Text);

}

} }

public void Altas(int id, string Nombre, string Apellidop, string ApellidoM, DateTime Fecha, string ocupacion, int eve, string correo)

{

SqlConnection dbconexion = new SqlConnection(cadena);

string comando = "insert into Persona values('"+ id +"','" + Nombre + "','" + Apellidop + "','" + ApellidoM +"','"+ Fecha +"','" + ocupacion + "','" + eve + "','"+correo+"')";

SqlCommand comand = new SqlCommand(comando, dbconexion);

dbconexion.Open();

comand.ExecuteNonQuery();

dbconexion.Close();

}

# Funcionamiento página Gafete

El método nos ayudara a crear el pdf alinear las imágenes y guardar el pdf después se le cargaran los nombres del cual pueda llenar los gafetes con los respectivos nombres

public void metodo2()

{

Document pdfDoc = new Document(PageSize.LETTER);

PdfWriter.GetInstance(pdfDoc, Response.OutputStream);

pdfDoc.AddTitle("Gafets");

pdfDoc.AddCreator("Ale Larregui");

pdfDoc.Open();

//alineacion y creacion de elemento paragraph

Paragraph nombre = new Paragraph(TextBox1.Text);

nombre.Alignment = Element.ALIGN\_CENTER;

nombre.Font.Size = Convert.ToInt16(tamanioletra.Text);

nombre.Font = FontFactory.GetFont("Arial", Convert.ToInt16(posiciontext.Text));

nombre.Font.SetStyle(Font.BOLD);

//agregar imagen

string imagen = Server.MapPath(".") + Label1.Text;

//string imageFilePath = Server.MapPath(".") + "/Recursos/gafets.png";

iTextSharp.text.Image img = iTextSharp.text.Image.GetInstance(imagen);

img.Alignment = iTextSharp.text.Image.UNDERLYING;

img.SetAbsolutePosition(Convert.ToInt16(TextBox2.Text), Convert.ToInt16(TextBox3.Text));

//añade al documento

pdfDoc.Add(img);

pdfDoc.Add(nombre);

//cierre y abrir pdf

pdfDoc.Close();

Response.ContentType = "application/pdf";

Response.AddHeader("content-disposition", "attachment;" + "filename=gafets.pdf");

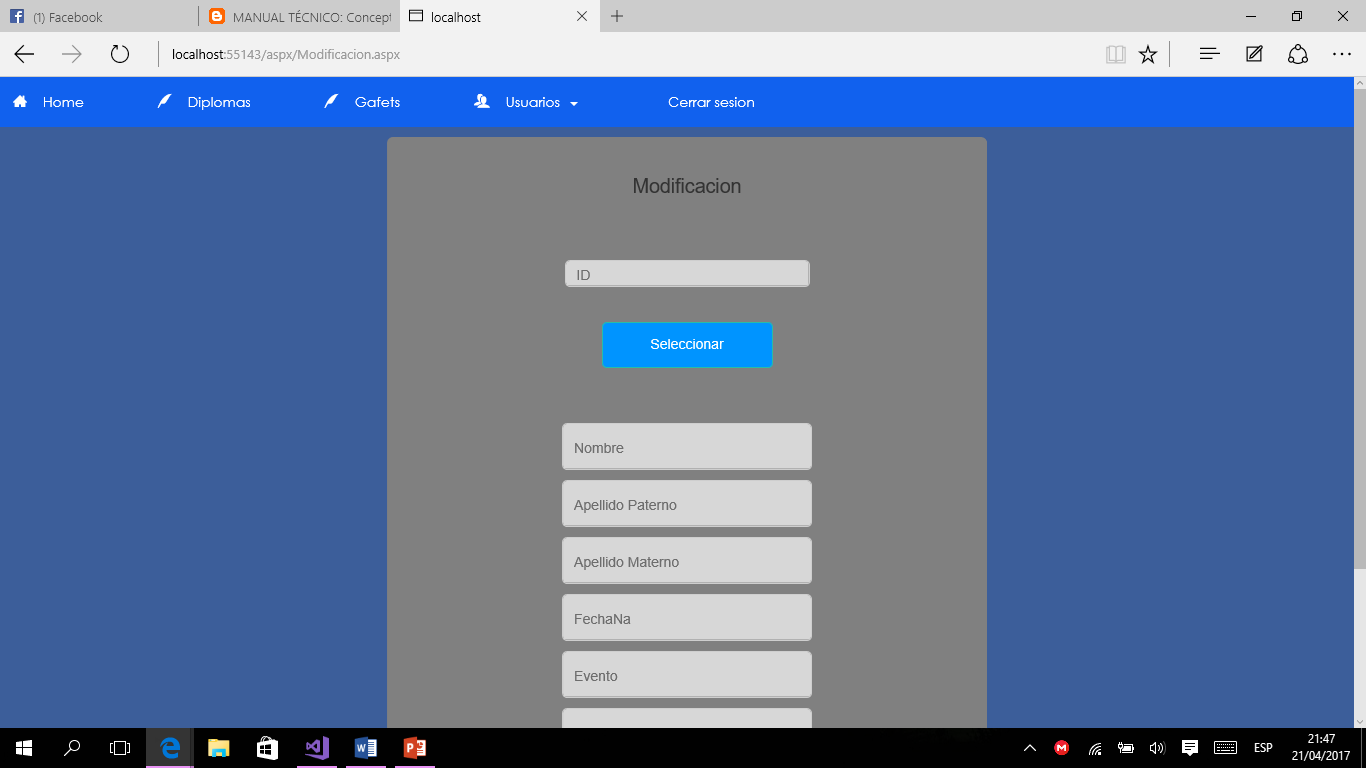
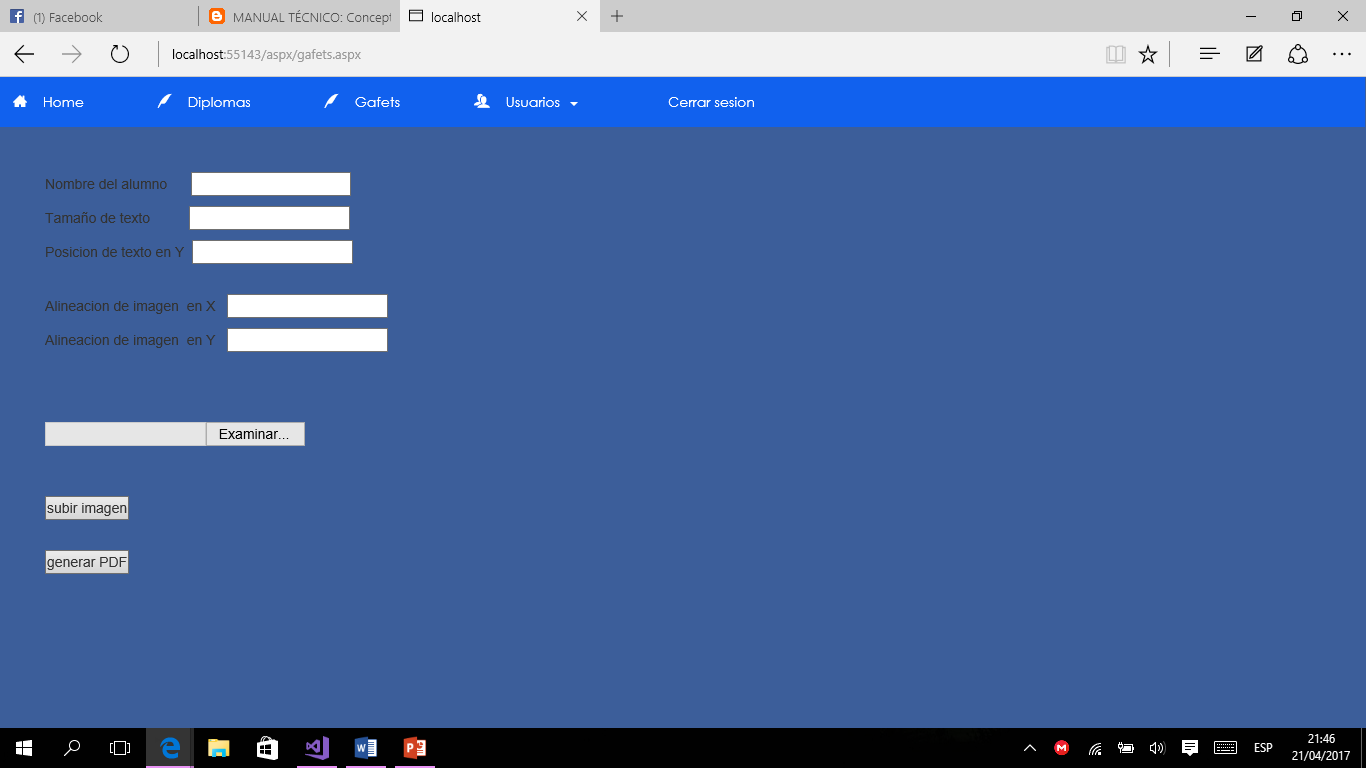
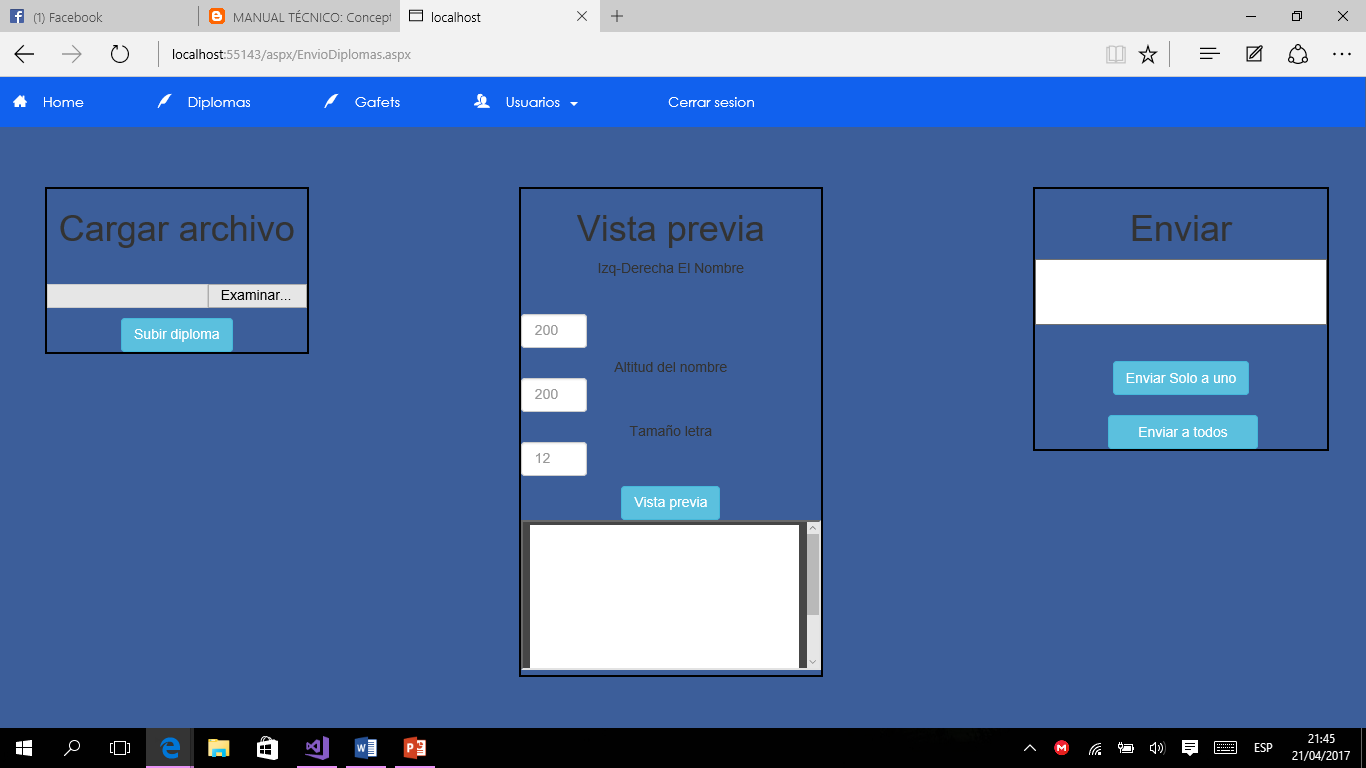
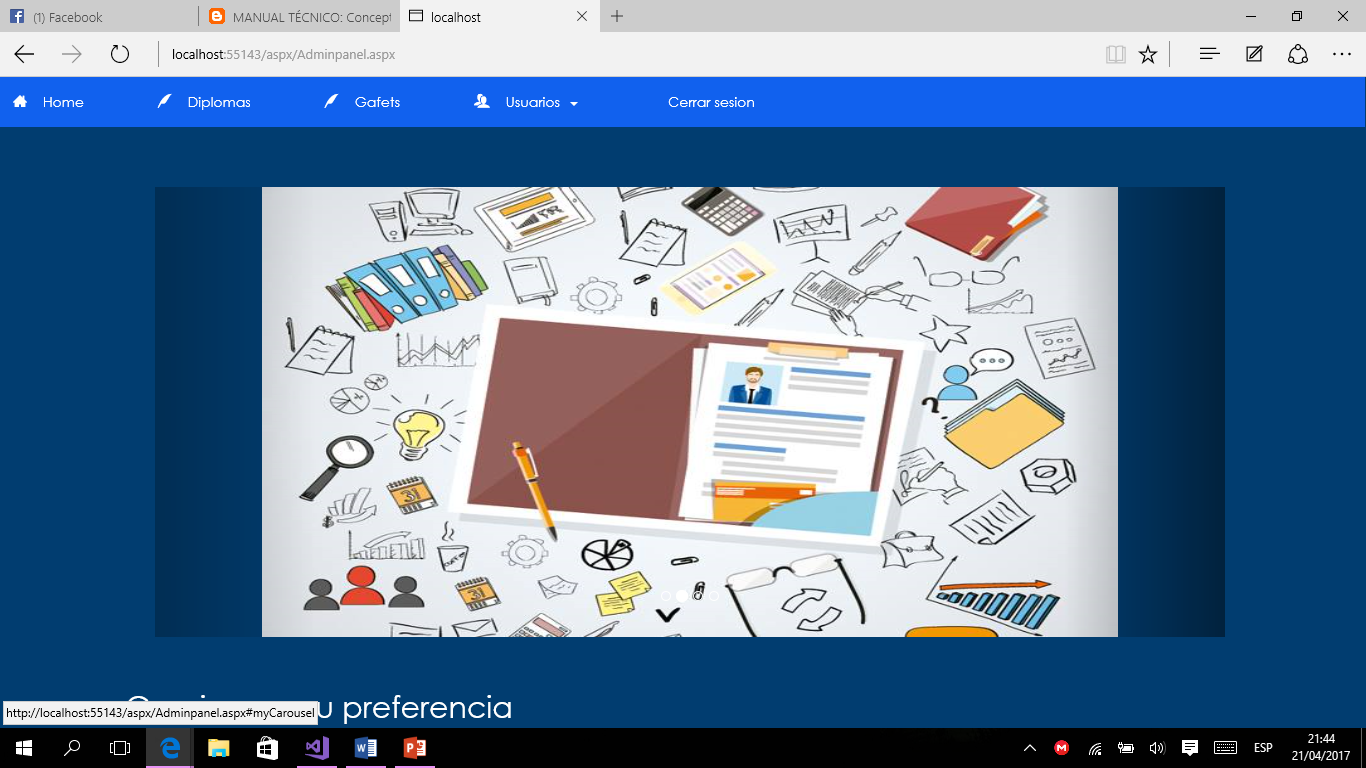
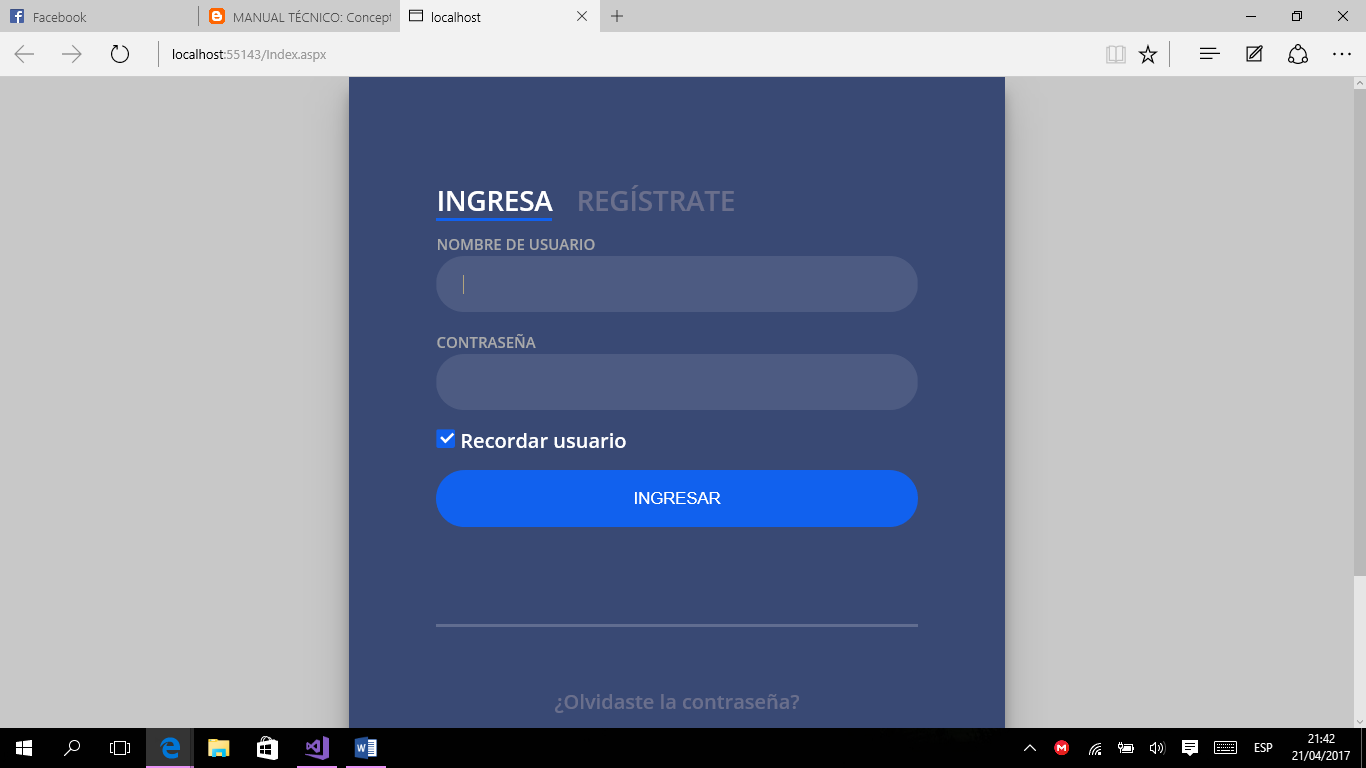
Response.Cache.SetCacheability(HttpCacheability.NoCache);

Response.Write(pdfDoc);

Response.End();

}

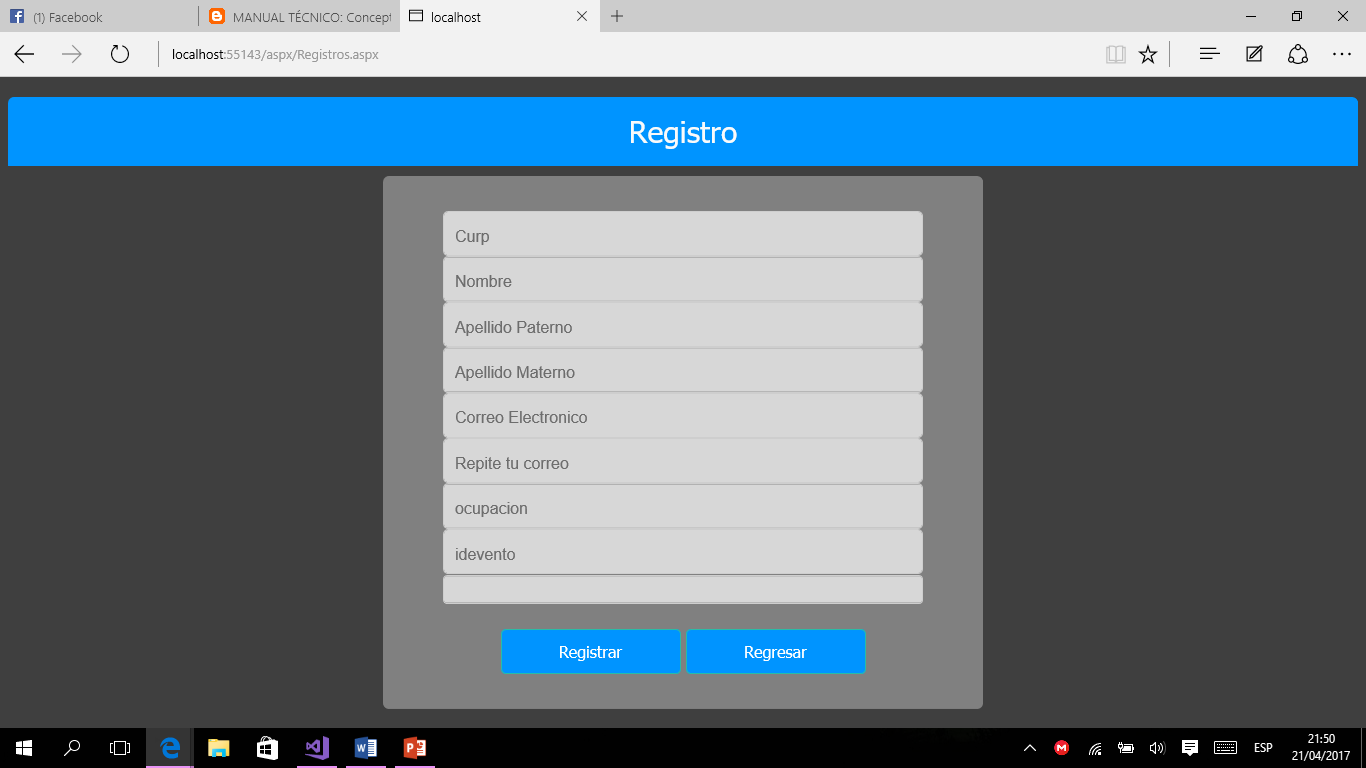
# Mapa de navegación



Modificación

Gafetes

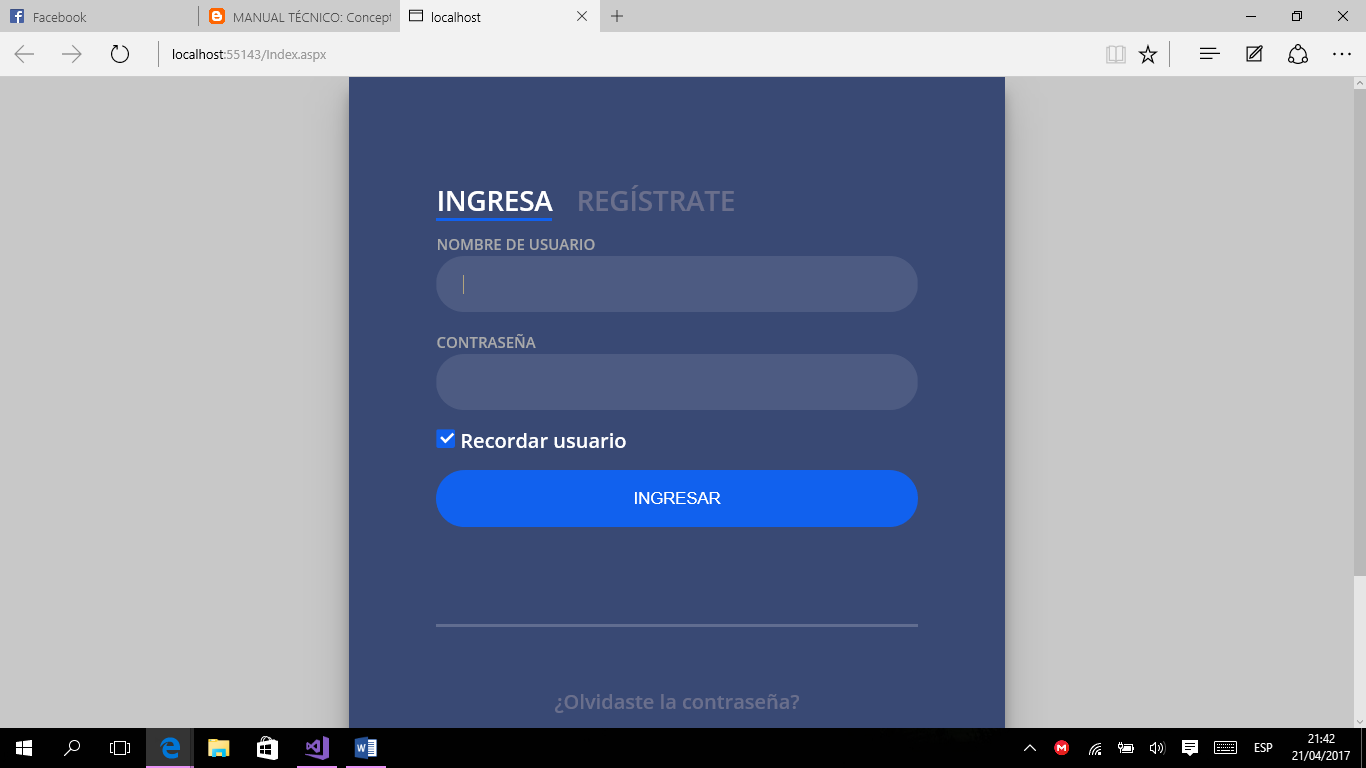
Envió Diploma



Registro

Index

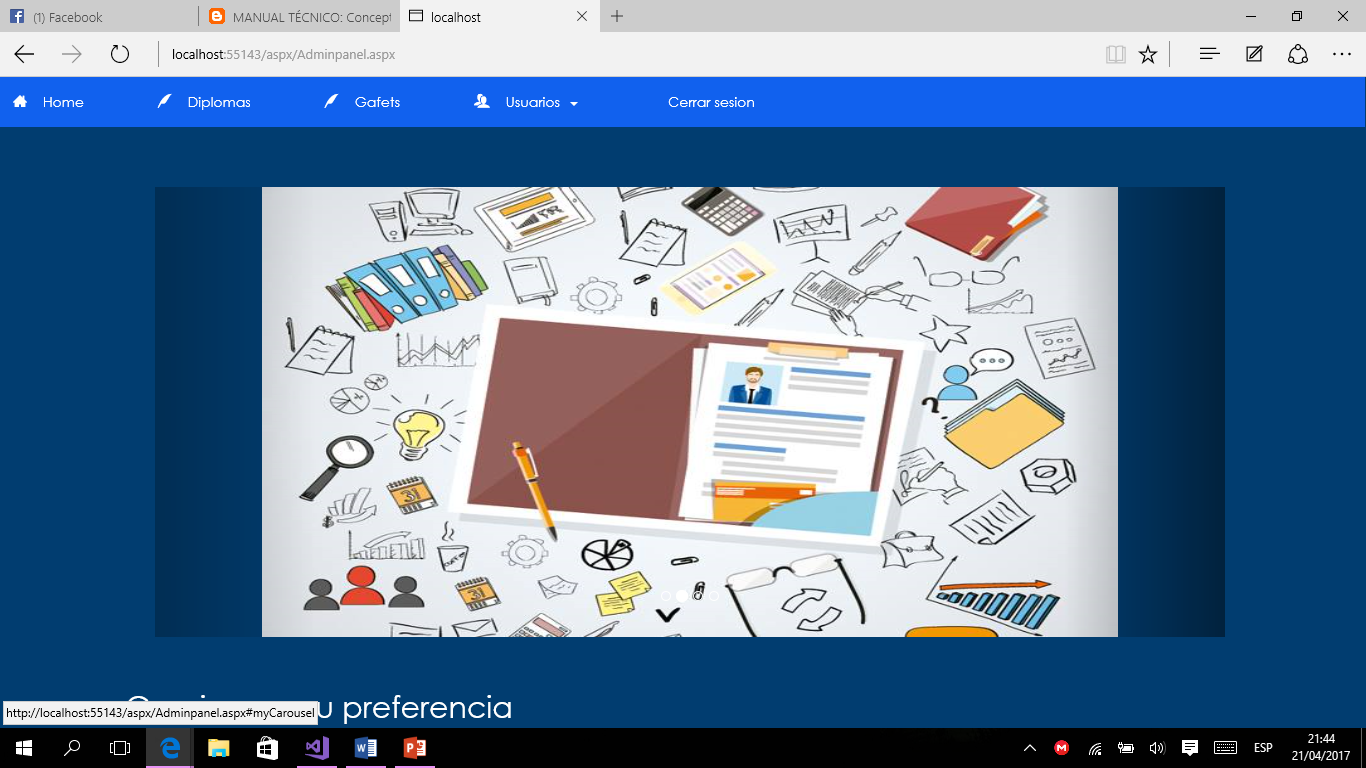
Admón. panel

Descripción del mapa de navegación.

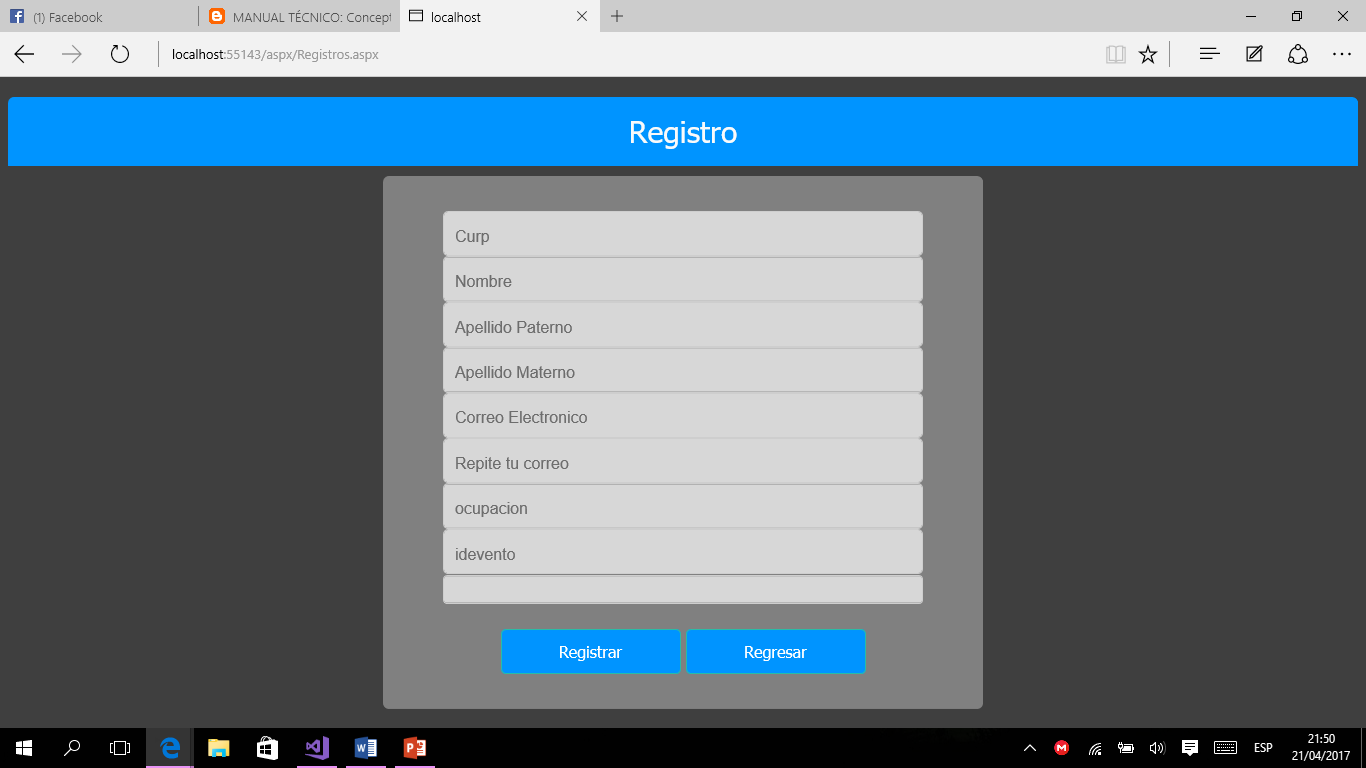
INGRESAR

El usuario, accederá con un usuario y contraseña antes registrado en ella. Tendrá la opción de recordar contraseña, en caso de no estar registrado podrá hacerlo en la siguiente página de registro.

Menú principal

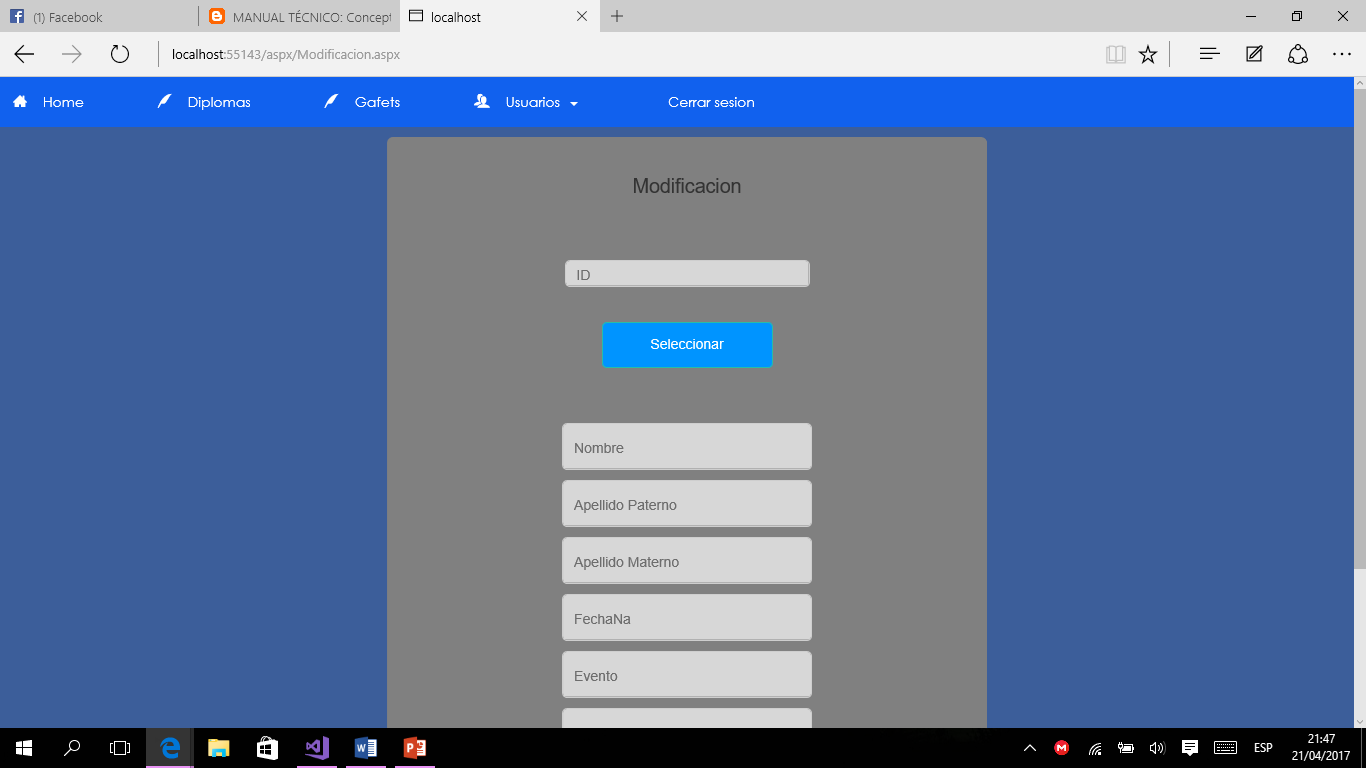
Al ingresar, el administrador podrá visualizar un menú con las siguientes actividades a realizar.

* Home
* Diplomas
  + - Pagina para enviar diplomas y agregar nombres.
* Gafets
* Crear gafets, imprimir.
* Usuarios
* Cerrar sesión

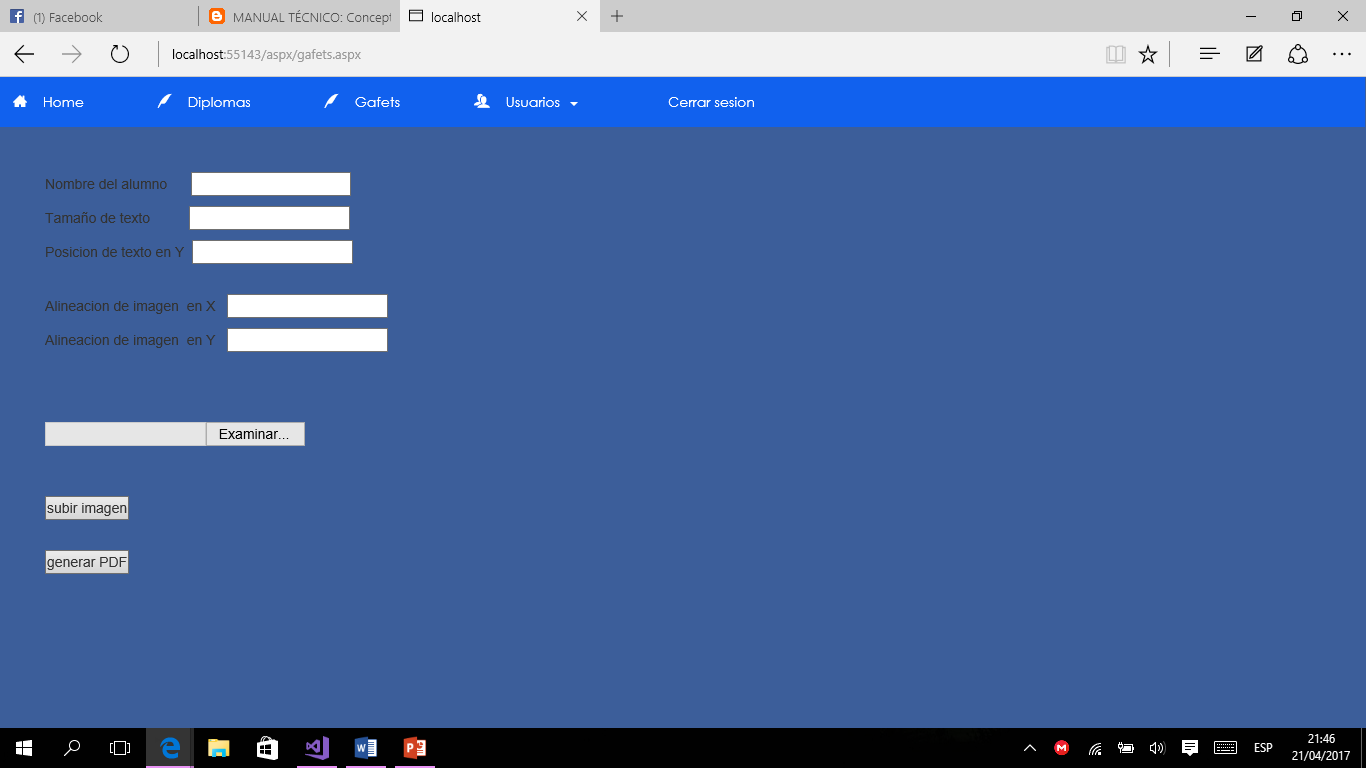
 Registrar

El usuario podrá registrar los siguientes de todos:

* CURP
* Nombre
* Apellido paterno y materno
* Correo electrónico
  + - * + En este campo se pedirá la repetición del correo para evitar errores.
* Ocupación
* Id evento
  + Idevento muestra el nombre de los cursos disponibles dentro de la universidad.

 Modificación

El administrador tendrá acceso a los datos de los alumnos, en donde podrá actualizar datos con solo ingresar el id del alumno y dar clic en seleccionar. Al encontrar el id, mostrara todos los datos y modificara los datos correspondientes.

Diplomas y gafets

En la página de diplomas el administrador seleccionara el nombre del alumno, agregara una plantilla de diploma y se asignaran los tamaños de texto, las posiciones e alineaciones de cada imagen.

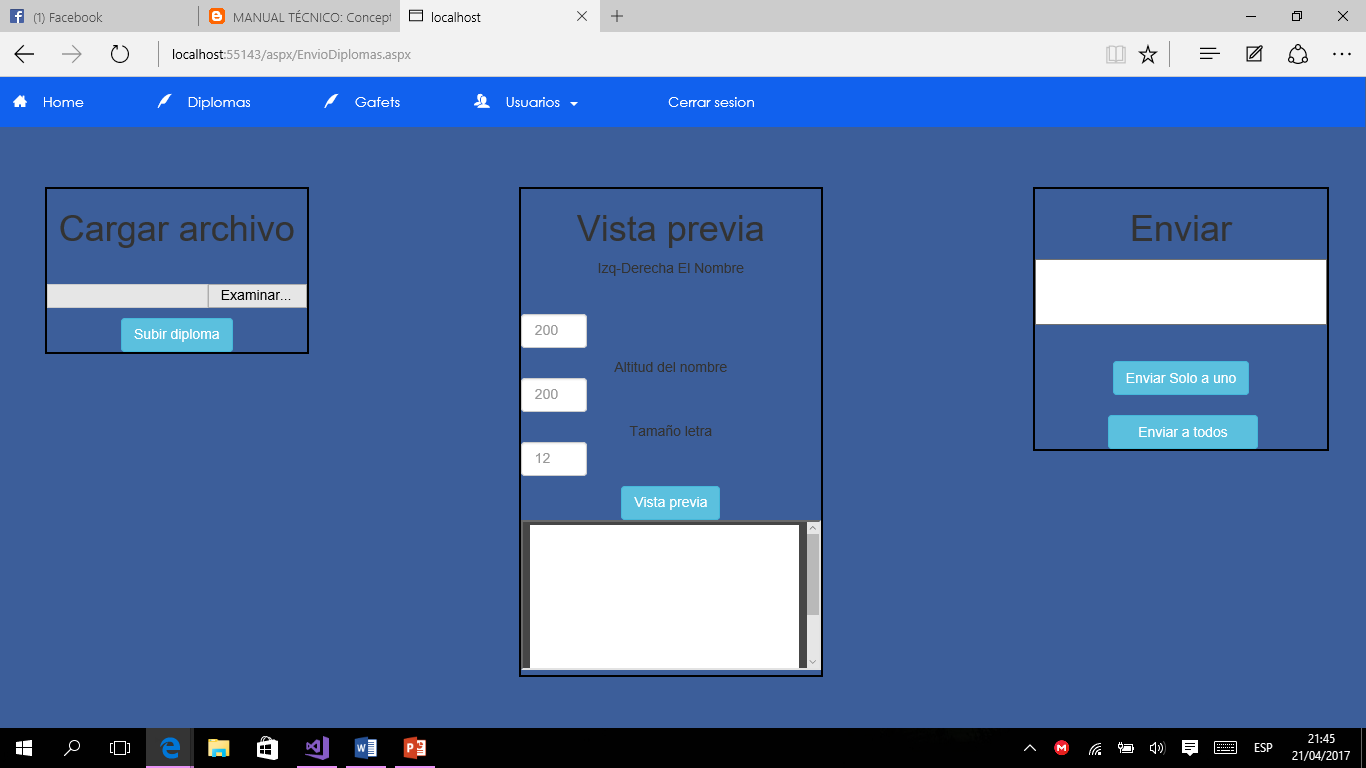
Al terminar de editar el diploma seleccionado el administrador tendrá dos opciones de visualización.

* Subir imagen
* Generar PDF

Ambas opciones son la manera más segura de no ser editadas por terceros, de esta manera el administrador enviara el diploma al alumno.

Envió

El envió de los diplomas al cargar el diploma modificado anteriormente. Al insertar el archivo el usuario tendrá una vista previa con las siguientes características.

* Posición del nombre
* Altitud del nombre
* Tamaño de letra

En donde se visualizara el diploma para ser enviado.

El administrador podrá enviar varios diplomas de diversos alumnos seleccionados o la opción de enviar un solo diploma.

# Base de datos

