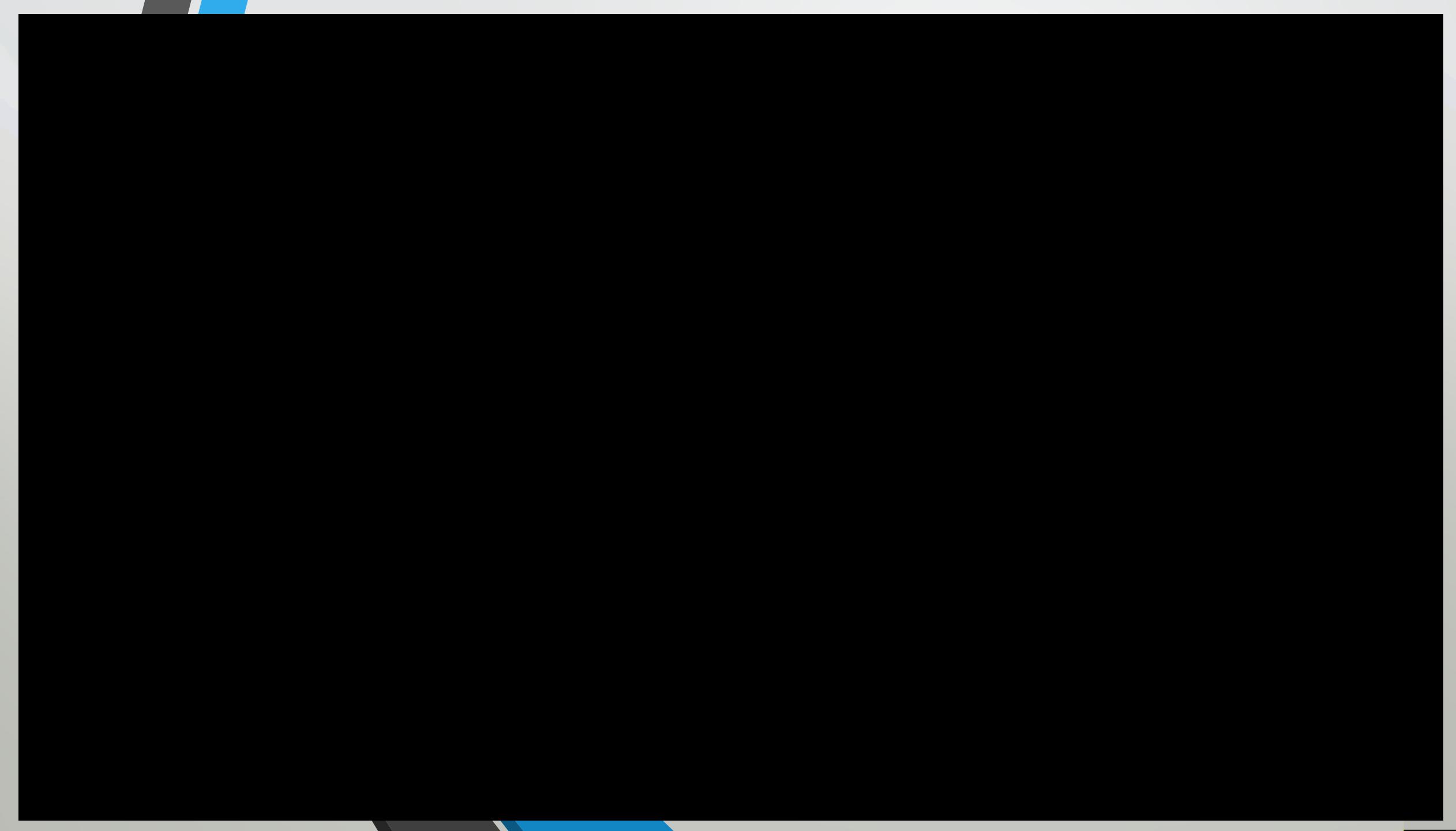


Software de Búsqueda de Aulas

Juna





Problemática:



Falta de información (o no tan a la vista) de ubicación de aulas.



Largas filas de consultas en diferentes oficinas.



Objetivo:

- ★ Acceso rápido a la ubicación de un aula a una materia asignada.
- ★ Fácil aprendizaje de la aplicación.
- Dirigido a compañeros de todas las edades.
- Con o sin experiencia en la rama informática.
- ★ Utilidades extras.

Objetivo Final:

- ★ Utilizar oficialmente como complemento para la página de la universidad.
- ★ Acceso desde puntos estratégicos mediante totems.
- ★ Posible utilización, a futuro, en otras universidades.



Logo y su significado:

Figura			
Descripción	Mano con el dedo índice apuntando hacia arriba	Círculo con relleno radial	El nombre del software
Significado de la figura	Símbolo de interactividad en los GUIs de escritorio También representa a los dispositivos táctiles y el gesto de indicar.	Simboliza el hotspot de un cursor Representa la precisión y exactitud del resultado que busca mostrar este software	La unión entre las letras representa la sinergia entre los distintos institutos
Significado de los colores	El color elegido es el de la universidad	El color negro en forma difuminada representa a la sombra del dedo índice y esta contenido por un contorno de el color de la UNAJ	Los colores corresponden a los institutos de la UNAJ La variedad representa la diversidad



Utilidades Extras:



Utilidades Extras :



Oficinas



Libros



Mapas

Proporciona información y ubicación de diferentes Departamentos, Oficinas e Institutos.

Possible filtro de búsqueda de libros proporcionados Por la facultad. (En Discusión)

Ubicación de las diferentes Sedes y Edificios.

Utilidades Extras:



Ubicación e Información:

- ★ De todos los institutos.
- ★ De todos los departamentos existentes.
Ejemplo: Departamento de Alumnos.
- ★ De todas las oficinas.
- ★ De todas las bedelías.
- ★ De todos los baños.
- ★ Ubicaciones extras: Biblioteca, Fotocopiadoras, Etc.

Utilidades Extras :



Ubicación todas las Sedes.



Ubicación todos los Edificios.

- Subsuelo: Aulas 01 a 07
- Planta Baja: Aulas 16-17 / 52 a 55
- Segundo Piso: Aulas 101 a 140

- Segundo Piso: Aulas 205 a 218
- Tercer Piso: Aulas 302 a 318
- Cuarto Piso: Aulas 407 a 420

Windows
en paré



Edificio Enrique Mosconi

Edificio Manuel Savio

Edificio Juan Pistorini

Edificio Silvio Dessi

Edificio Ing. Héctor Abrales

Edificio Guillermo Hudson

Edificio Julieta Lanteri

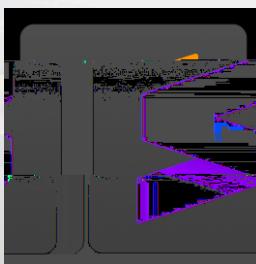
Edificio Manuel Ugarte

Edificio Homero Manzi

Edificio Ramón Carrillo

Activar
Ve a Cont

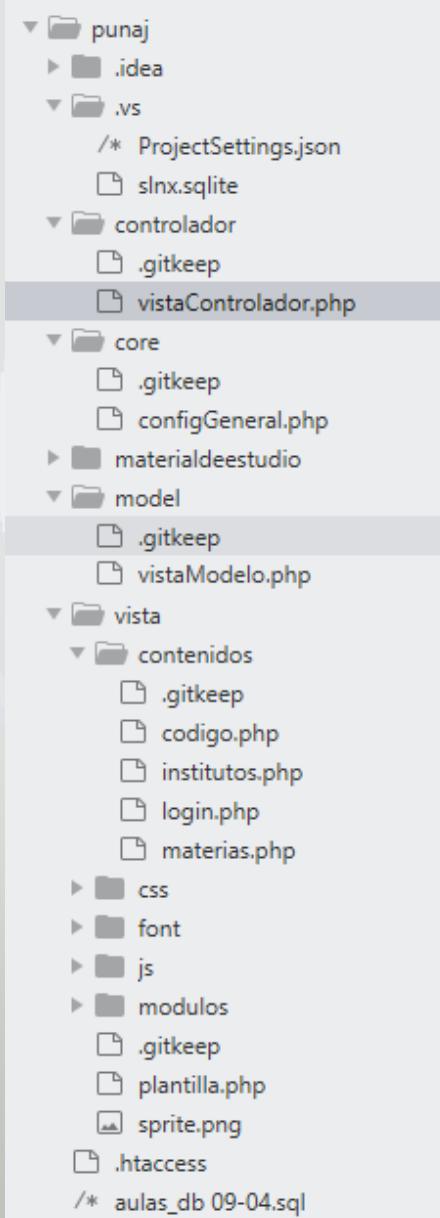
Herramientas de Desarrollo



 *Trello*
DISCORD



Juna con MVC



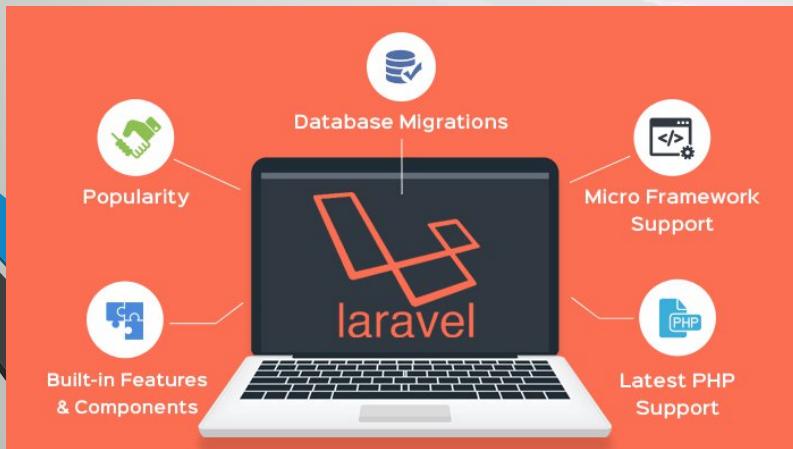
```
plantilla.php      vistaControlador.php      vistaModelo.php
1 <?php
2
3     class vistaModelo
4     {
5         /*
6             Funcion que busca el archivo la vista actual dependiendo de
7             lo que haya en la barra de tareas, o lo que seleccione
8             el usuario con los botones de la web.
9             Está conectado con el vistaControlador.
10            */
11
12        protected function obtener_vista_modelo($vista)
13        {
14            /*
15                En esta lista deben ir los nombres de todos los
16                ficheros que haya en la carpeta contenidos, donde
17                estan todo lo que se muestra al usuario y varía.
18            */
19            $listaBlanca = ["home", "institutos", "materias", "carreras", "años",
20
21
22            /*
23                Se testea si el parametro $vista es un archivo válido dentro
24                de la carpeta contenidos, y lo llama.
25                Si no es un archivo valido, o si es login/index entonces llama
26                al "login"
27            */
28            if (in_array($vista, $listaBlanca))
29            {

```

FrameWorks

Juna con Laravel:

- ★ Usa MVC.
- ★ Simplicidad de creación de vistas y sus respectivos controladores
- ★ Fácil implementación y manejo de base de datos mediante Eloquent.



Juna con Laravel:



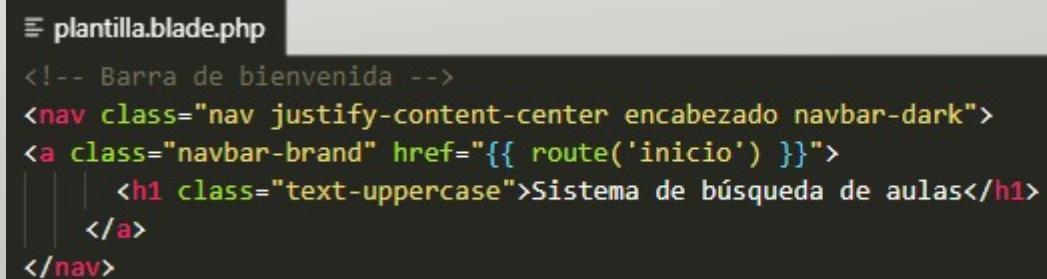
```
web.php
Route::get('/', 'PagesController@inicio')->name('inicio');

PagesController.php
public function inicio()
{
    return view('inicio');
```

SISTEMA DE BÚSQUEDA DE AULAS



The screenshot shows a user interface for a classroom search system. At the top, there's a blue header bar with the title "SISTEMA DE BÚSQUEDA DE AULAS". Below the header are four circular icons with corresponding labels and icons: "Aulas" (green circle with a person icon), "Libros" (yellow circle with a book icon), "Oficinas" (teal circle with a building icon), and "Mapas" (red circle with a location pin icon). At the bottom, a blue footer bar displays the text "UNIVERSIDAD NACIONAL ARUTO JAURETCHE".



```
plantilla.blade.php

<nav class="nav justify-content-center encabezado navbar-dark">
<a class="navbar-brand" href="{{ route('inicio') }}>
    <h1 class="text-uppercase">Sistema de búsqueda de aulas</h1>
</a>
</nav>
```

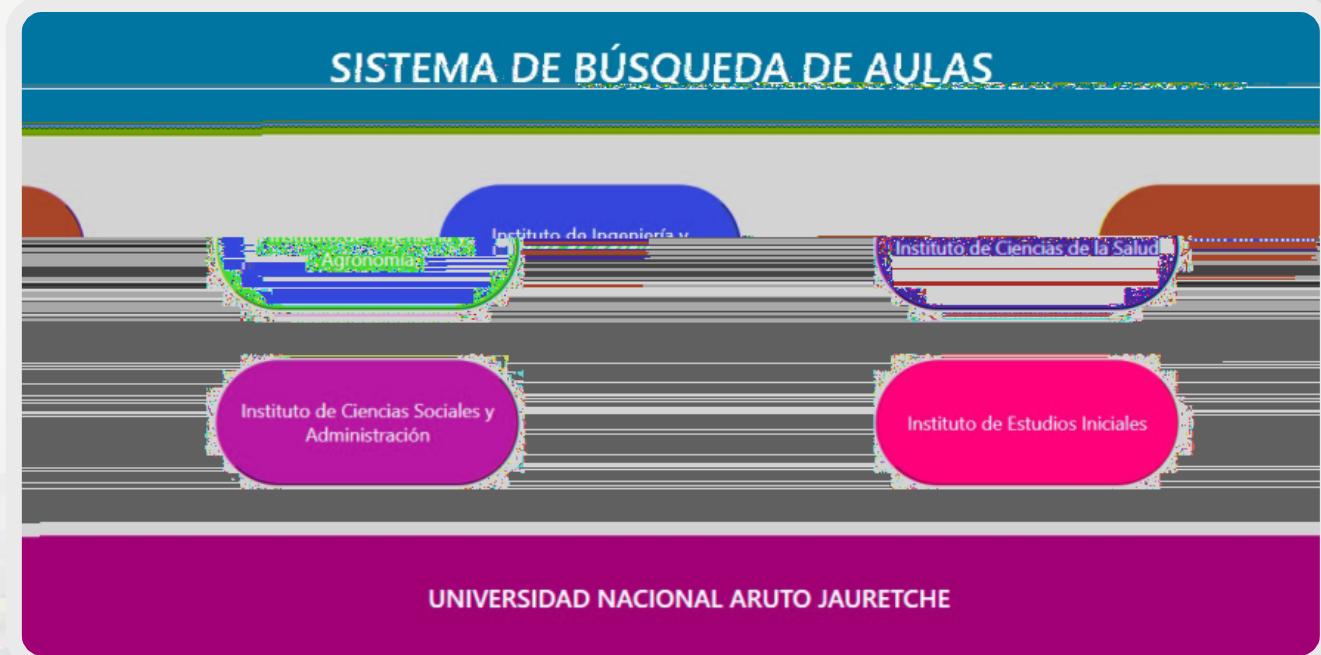
Juna con Laravel:

web.php

```
Route::get('institutos', 'PagesController@institutos')->name('institutos');
```

PagesController.php

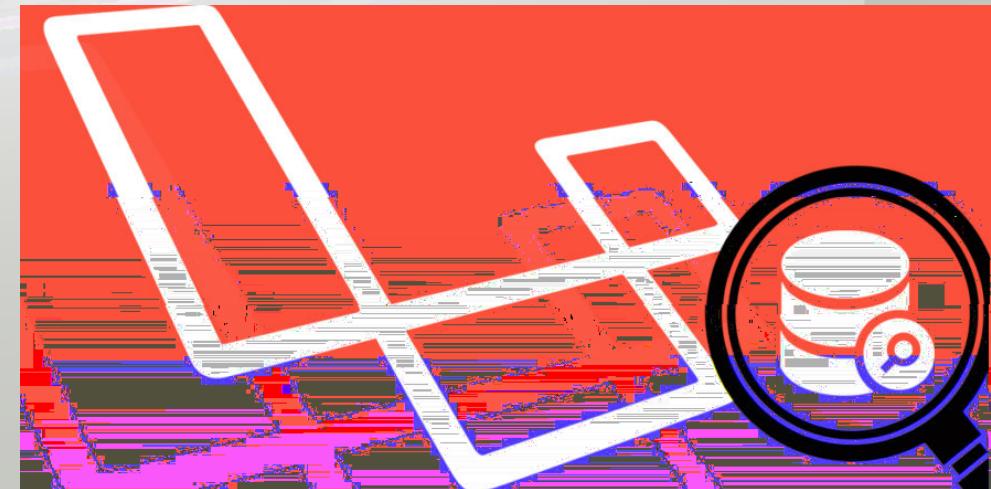
```
public function institutos()
{
    $nom_inst = ['Ingeniería', 'Sociales', 'Salud', 'Iniciales'];
    return view('institutos', compact('nom_inst'));
}
```



UNIVERSIDAD NACIONAL ARUTO JAURETCHE

Base de datos con Laravel:

- ★ Laravel proporciona ORM Eloquent.
 - Uso de Modelos.
 - Migraciones de tablas.
 - Factories y Seeders.



Base de datos con Laravel:

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Edificio extends Model
{
    public function sede()
    {
        return $this->belongsTo(Sede::class);
    }
}
```



```
public function up()
{
    Schema::create('edificios', function (Blueprint $table) {
        $table->increments('id');
        $table->string('nombre');

    });

    Schema::table('edificios', function (Blueprint $table) {
        $table->integer('sede_id')->unsigned();
        $table->timestamps();
        $table->foreign('sede_id')->references('id')->on('sedes');

    });
}
```



Base de datos con Laravel:

```
class EdificiosTableSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        factory(App\Edificio::class, 1)->create();
    }
}
```

↑
Edificio Seeder

```
$factory->define(App\Edificio::class, function (Faker $faker) {
    return [
        'nombre' => $faker->name,
    ];
});
```

↑
Edificio Factory

Ajax en Juna:

Pop-up Materias

```
contenidoMate = document.getElementById('listarMaterias');//ID DONDE VOY A MOSTRAR EL RESULTADO
function materiasAjax(url){
    console.log(url);
    const miajax= new XMLHttpRequest();
    // INSTANSIAMOS OBJETO XMLHttpRequest ES EL QUE NO SE PUEDE CREAR DE MANERA SENCILLA
    miajax.open( GET ,url,true);
    miajax.send();
    miajax.onreadystatechange = function(){
        if (this.readyState == 4 && this.status == 200) {
            let datos=JSON.parse(this.responseText);
            console.log(datos);
            proyeccion(contenidoMate,datos);
            return datos;
        }
    }
}
```

Proyectando materias:

```
// FUNCION PARA PROYECTAR
function proyeccion (contenido, listaCarreras){
    contenido.innerHTML='';
    for(let item of listaCarreras){
        contenido.innerHTML +=`|
            <td>
                ${item.nombre}
            </td>
            <td>
                <h4 class="boton-carreras">${item.anio}</h4>
            </td>
            <td>
                <a href="/listarComisiones/${item.id}" class="btn btn-success" role="button">Buscar</a>
            </td>
        </tr>
        `
    }
}
|  |

```

Javascript en Juna:

Filtro de Comisiones

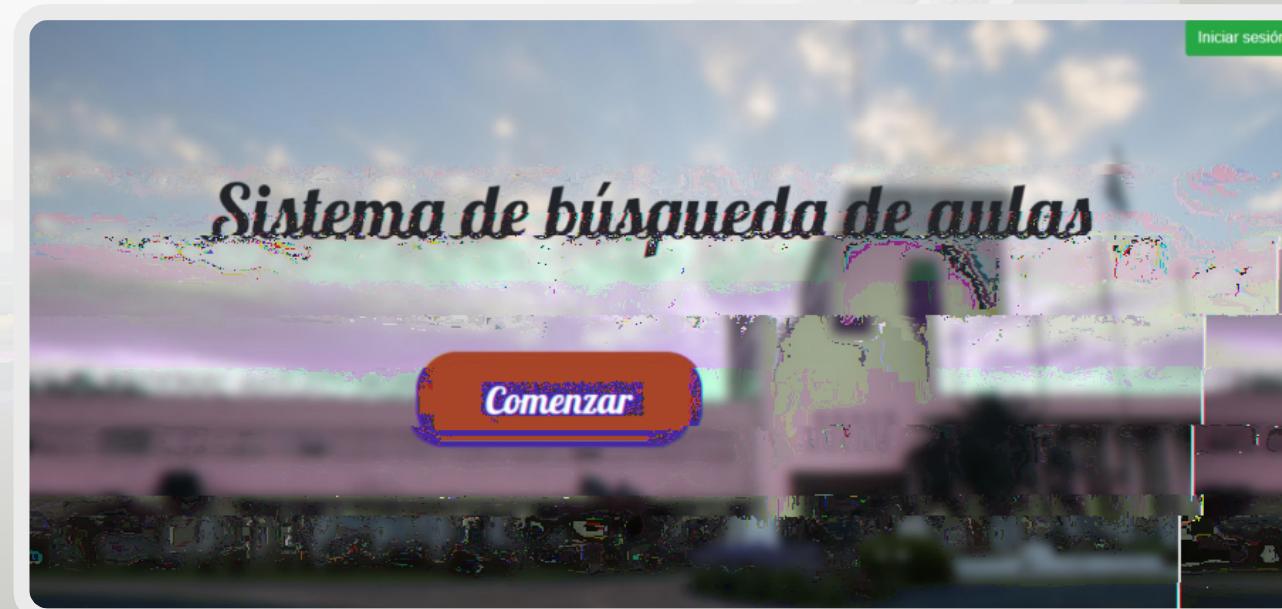
```
function filtroComision() {
    var input, filter, ul, li, a, i, txtValue;
    input = document.getElementById("myInputcomi");//CAPTURO ID INPUT
    filter = input.value.toUpperCase();//CONVIERTO SU CONTENIDO TODO A MAYUSCULA
    ul = document.getElementById("myULcomi");//CAPTURO ID UL
    li = ul.getElementsByTagName("li");//CAPTURO TAG LI
    for (i = 0; i < li.length; i++) {
        a = li[i].getElementsByTagName("p")[0];
        //CAPTURO EL PRIMER ELEMENTO P DEL TAG LI Y LO GUARDO EN VARIABLE A
        txtValue = a.textContent || a.innerText;//OBTENGO EL CONTENIDO DE TEXTO DE A
        if (txtValue.toUpperCase().indexOf(filter) > -1) {// BUSCO EN LA CADENA EL CONTENIDO INGRESADO
            li[i].style.display = "";
        } else {
            // SI NO TIENE CONTENIDO NO MUESTRO EL TAG LI
            li[i].style.display = "none";
        }
    }
}
```

Diferentes Modelos:

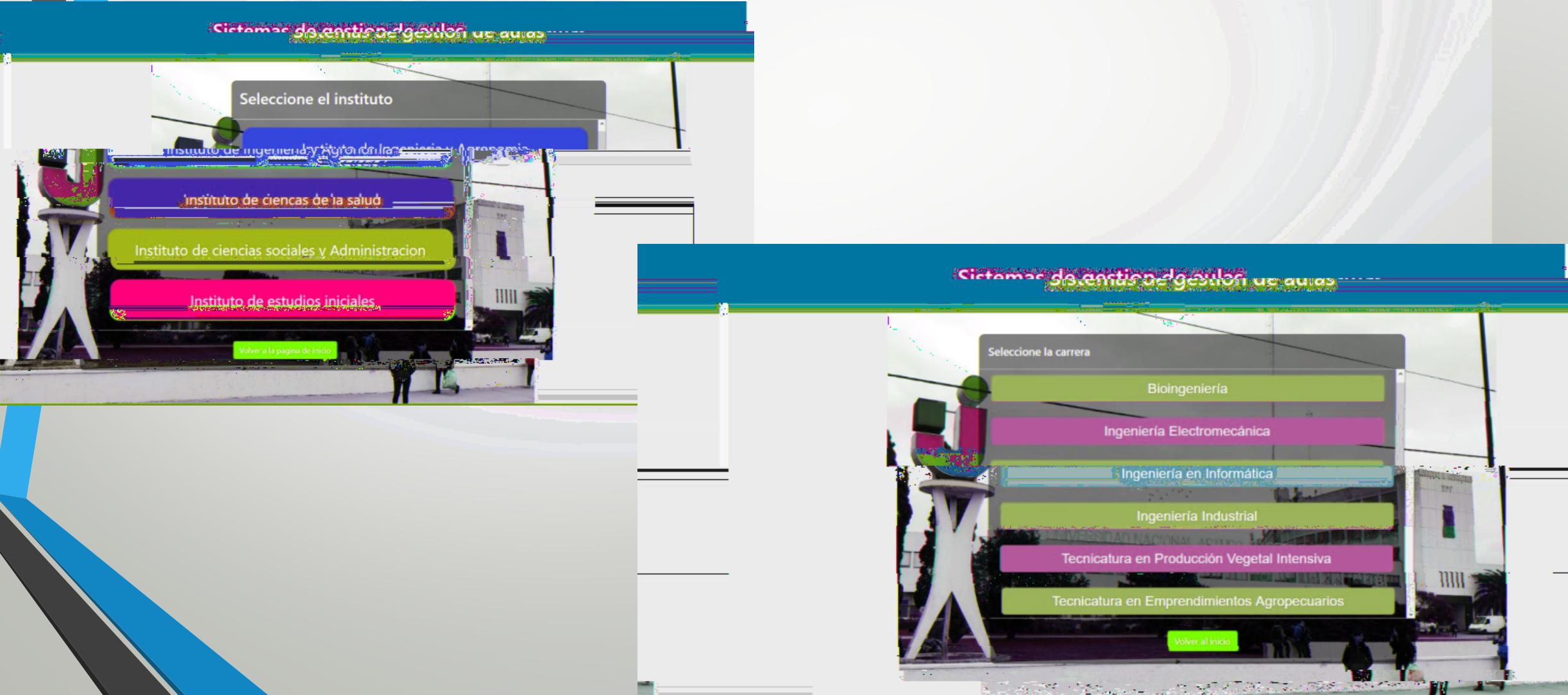
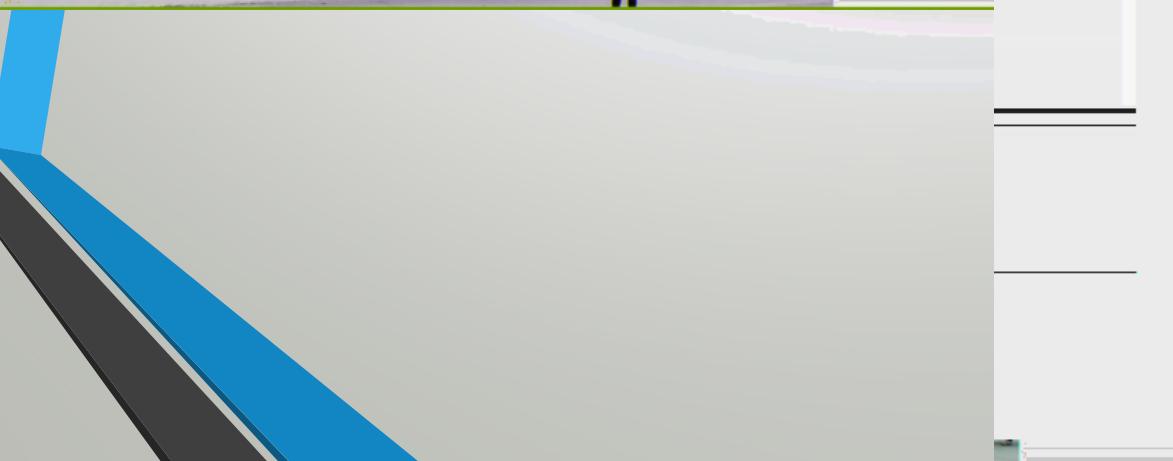
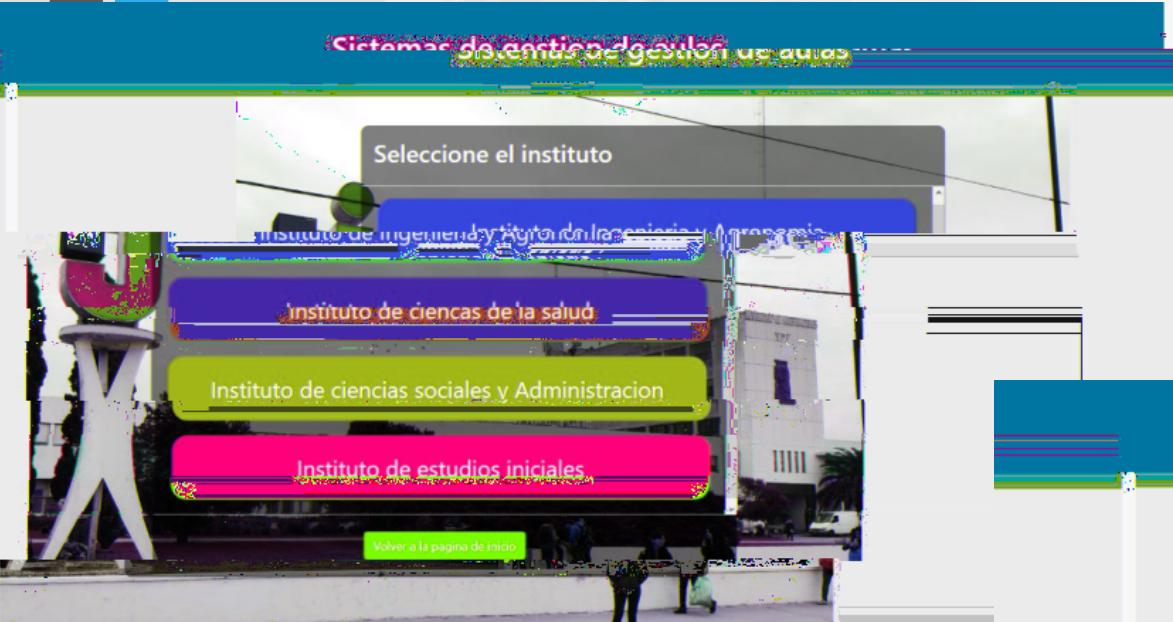
Diseño Principal



Diseño Administrativo



Diferentes Modelos: Diseño Principal Alternativo



Diferentes Modelos: Diseño Principal Alternativo

Sistema de Gestión de Aulas



Universidad Arturo Jauretche

Sistema de Gestión de Aulas

Seleccione el Instituto al que pertenece



Instituto de Ingeniería y
Agronomía

Instituto de Ciencias Sociales y
Administración



Sistema de Gestión de Aulas

Instituto de Ingeniería y Agronomía



Ingenería en Petróleo

Ingenería en Informática

Bioingenería

Ingenería Industrial

Ingenería Electromecánica

Ingenería en Administración Agraria

Ingenería en Ciencias Agrarias

Ingenería en Transporte

Tec. Universitaria en Emprendimientos Agropecuarios

Tec. Universitaria en Producción Vegetal Intensiva

Framework: Bootstrap

- Sistema de cuadrícula (GRID)
- Contenedores (12 columnas)

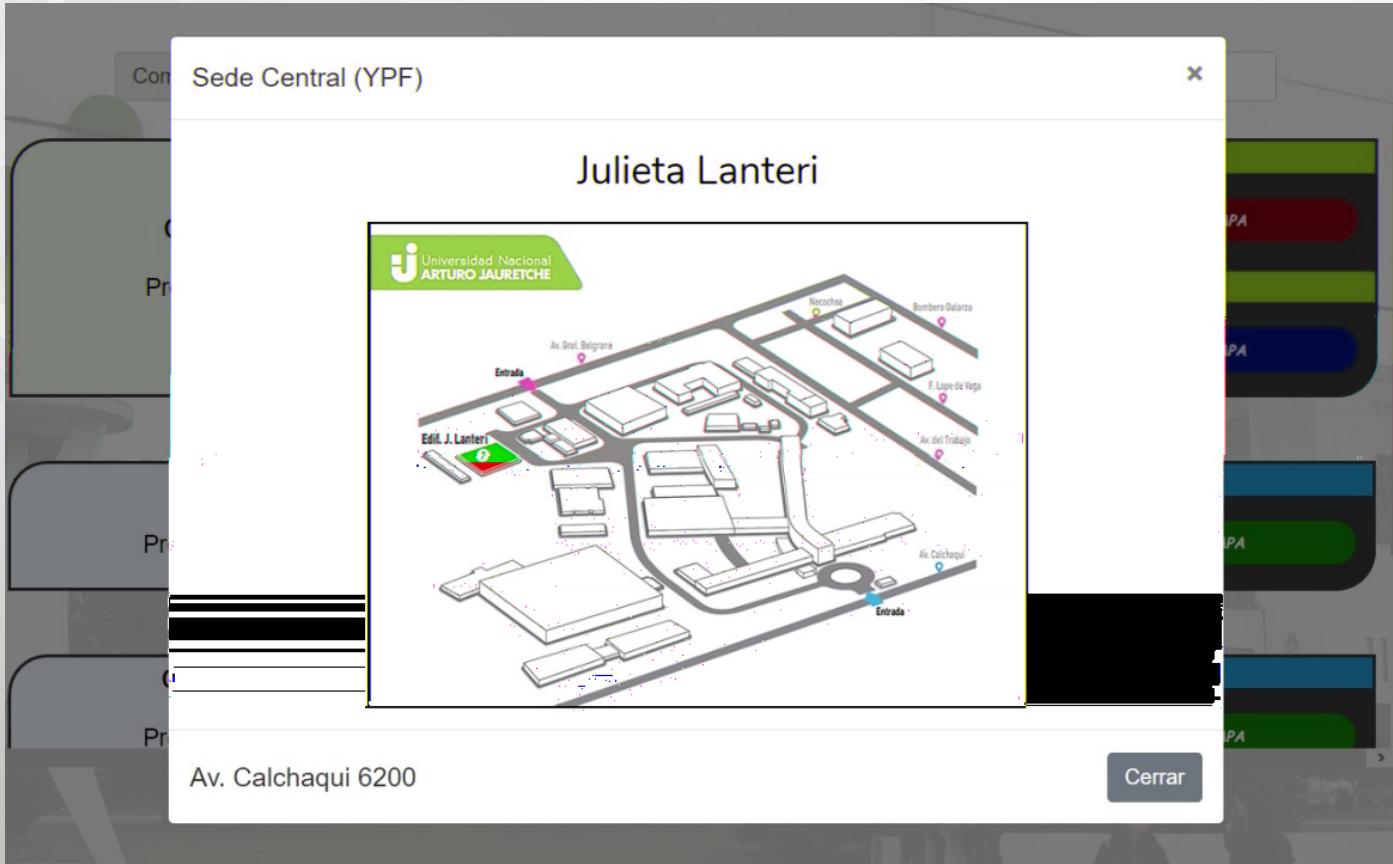


Adaptable: A distintos dispositivos



Framework: Bootstrap

- Diseño de botones, tablas y modal (popup)

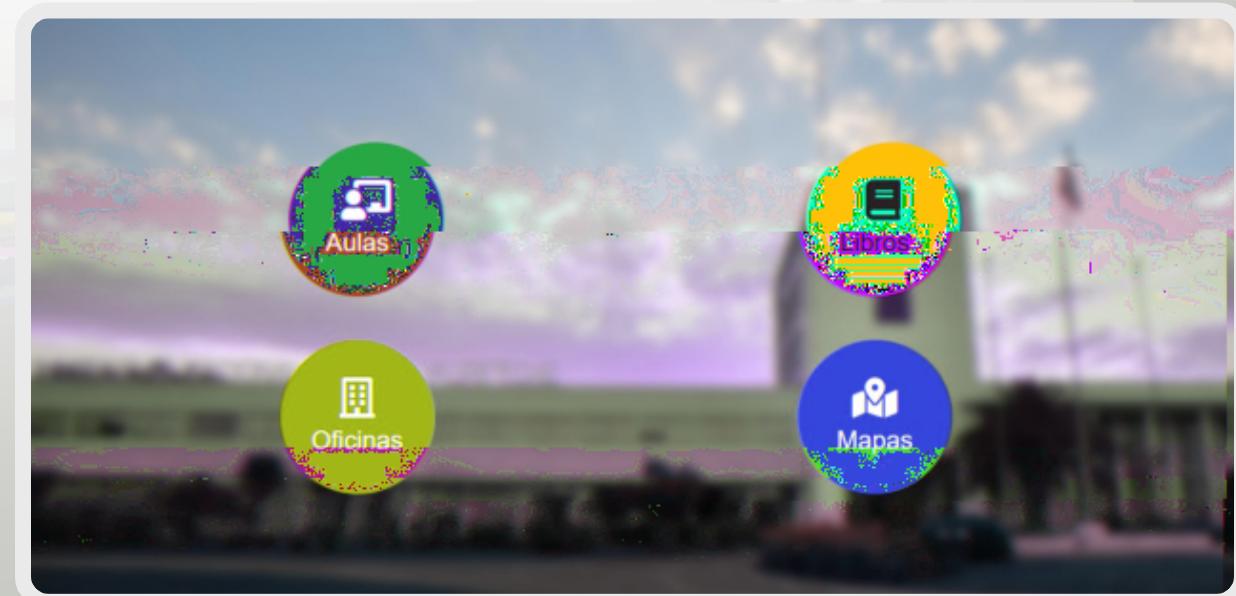
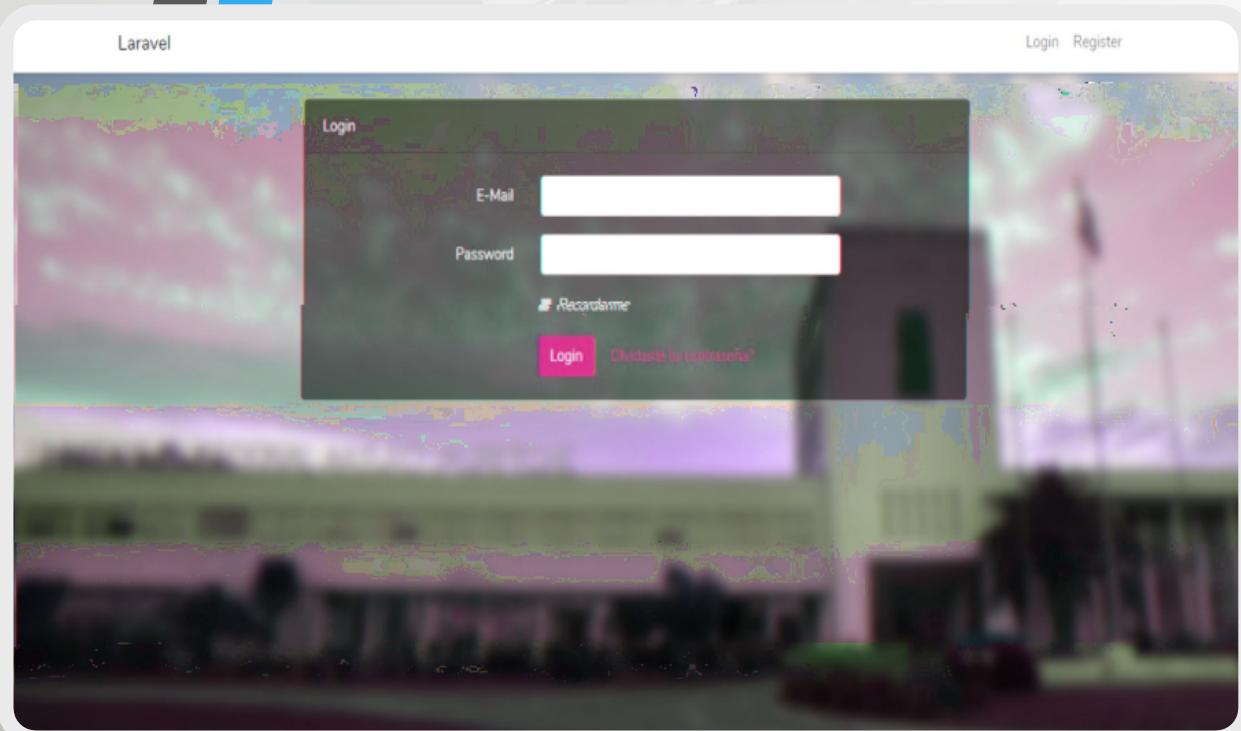


Colores del diseño:

Botones y cabecera de colores de los institutos de la universidad



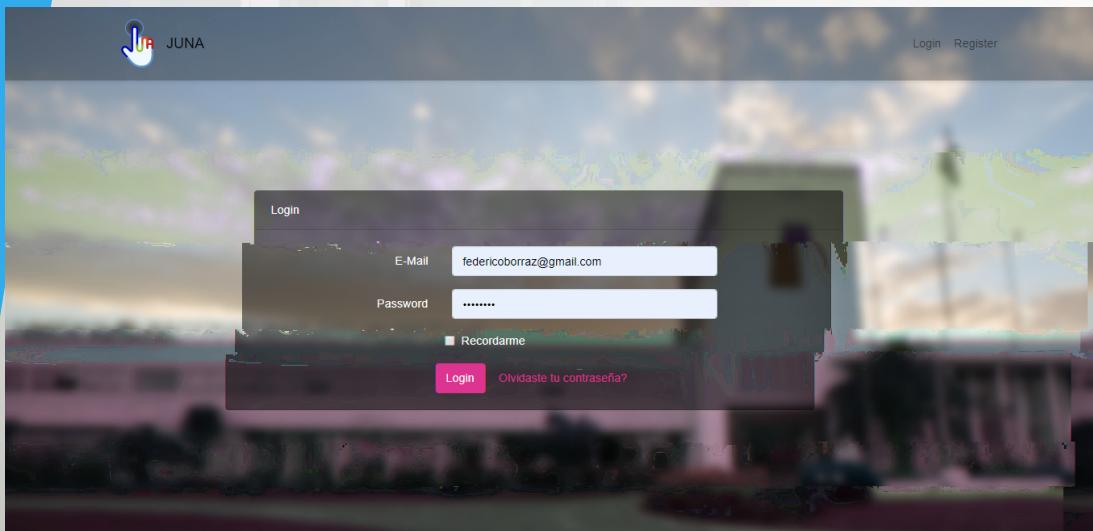
Diferentes Modelos: Diseño Administrativo



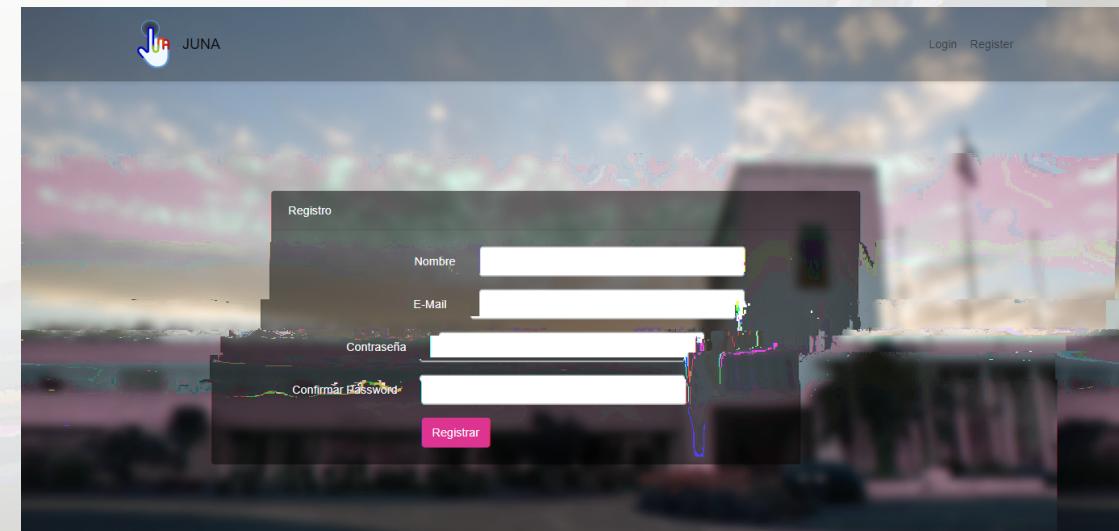
Funcionalidad a incorporar: Autenticación de usuarios

- APP Web pensada no solamente para la UNAJ sino también para todas las universidades nacionales.
- Migrador de base de datos propio.
- Modificación de base de datos.
- Gestión de usuarios con permisos para modificar la base de datos.

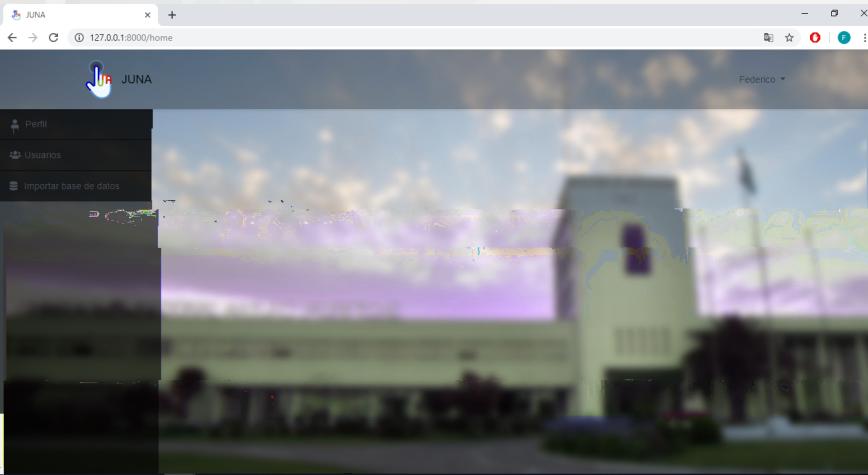
Login



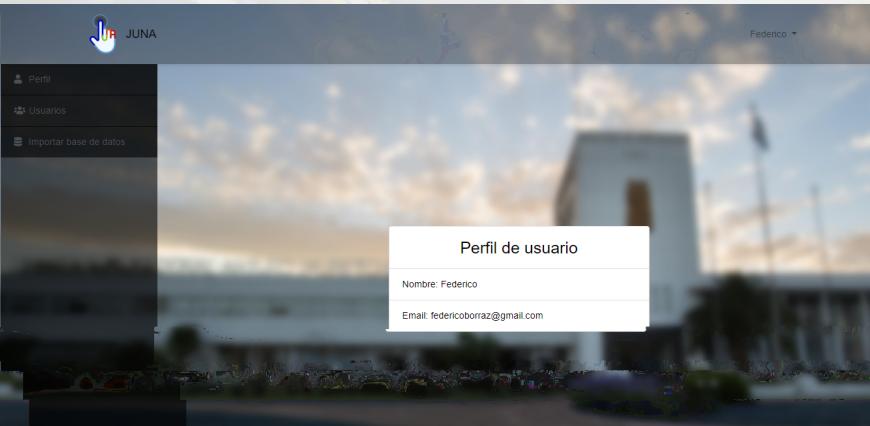
Registro de usuarios



Vista del usuario autenticado



Perfil de usuario autenticado



Usuarios registrados

A screenshot of the JUNA application showing a list of registered users. The sidebar menu shows "Usuarios" is selected. The main content area displays a table with the following data:

ID	Nombre	Email	Fecha de creación	Detalles
1	Federico	federicoborraz@gmail.com	2019-07-17 18:35:49	Ver detalles
2	Roberto	roberto@gmail.com	2019-07-17 18:46:54	Ver detalles
3	David	david@gmail.com	2019-07-17 22:43:18	Ver detalles
4	Alex	alex@gmail.com	2019-07-17 22:43:40	Ver detalles
5	Nahuel	nahuel@gmail.com	2019-07-17 22:44:12	Ver detalles



The background features a series of concentric, slightly curved circles in shades of light gray, white, and yellow. On the left side, there are two prominent diagonal stripes: a thick dark gray one at the top and a thick blue one below it, which extends towards the bottom right corner.

¡Gracias!