

PROYECTO INTEGRADO

Realizado por: Marlen Noemi Sajché López



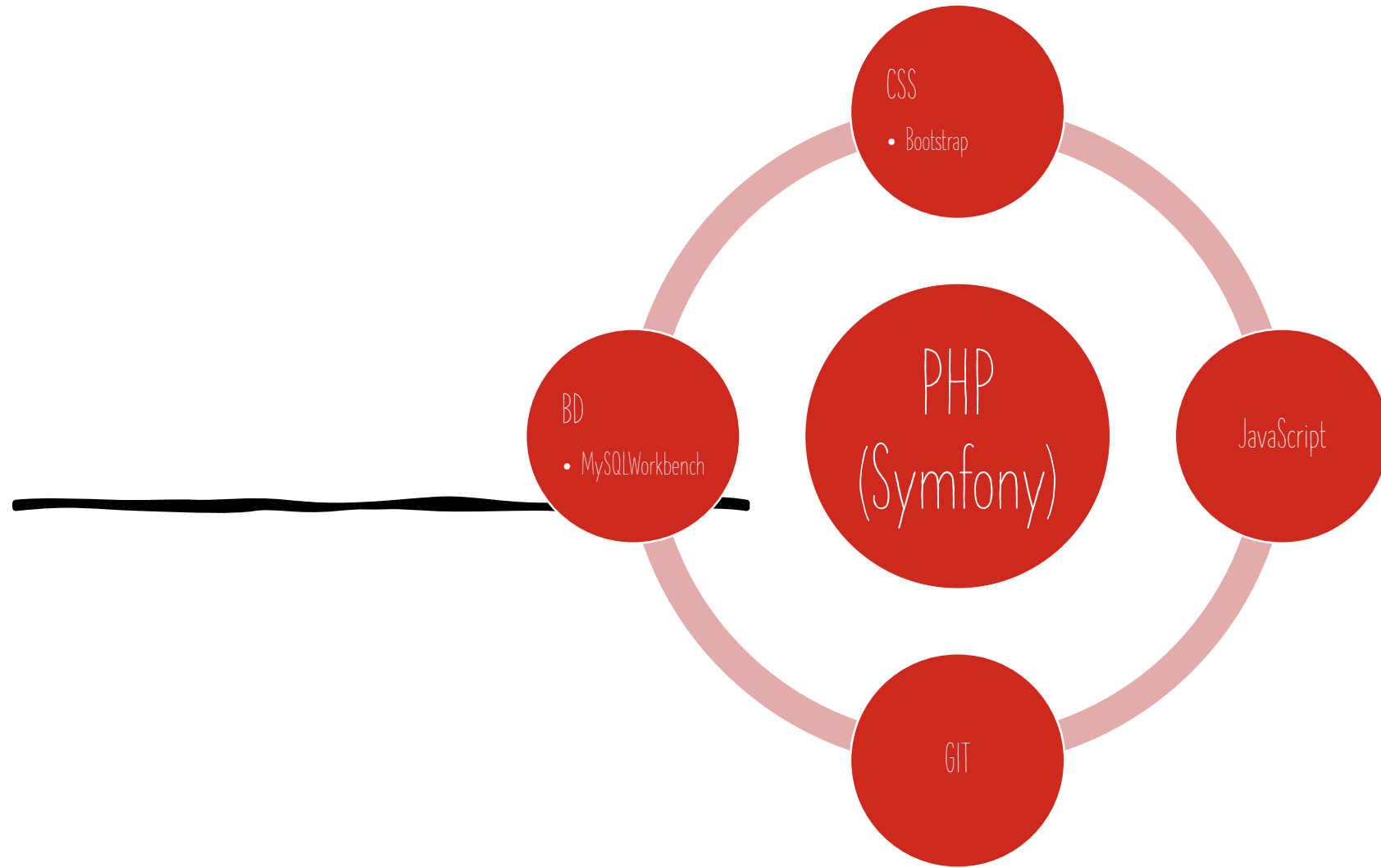
Gestión de empresa de eventos



JUTIFICACIÓN

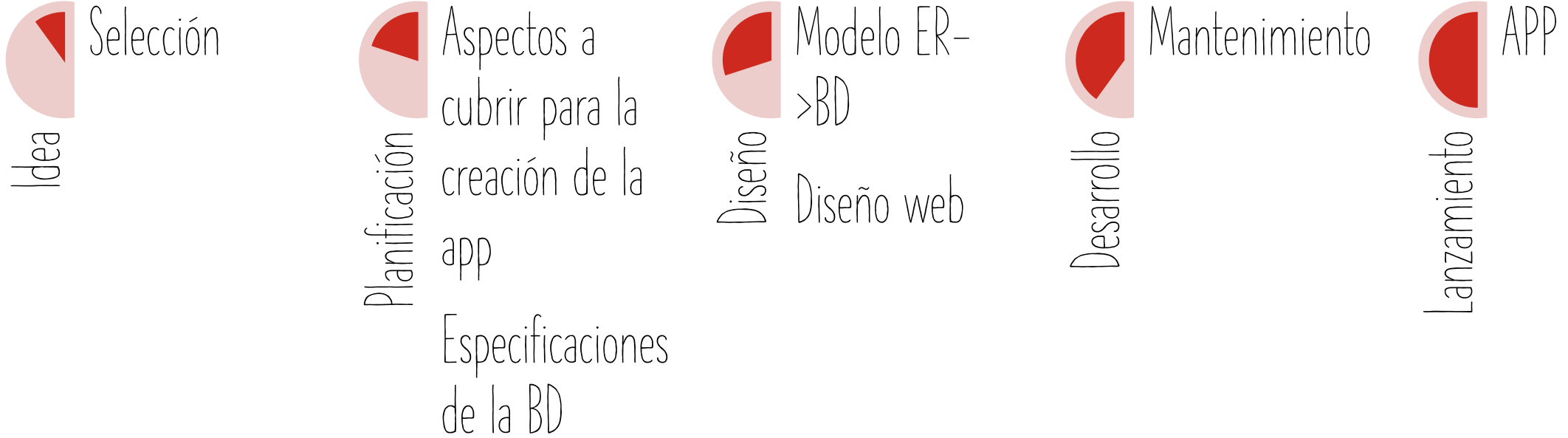
El interés por desarrollar el proyecto es dar al cliente toda la información necesaria para que se cree un contexto real de los productos y/o servicios que la empresa ofrece, con la simplicidad de interactuar con una aplicación web, optimizar el tiempo del cliente y la empresa.

TECNOLOGÍAS UTILIZADAS

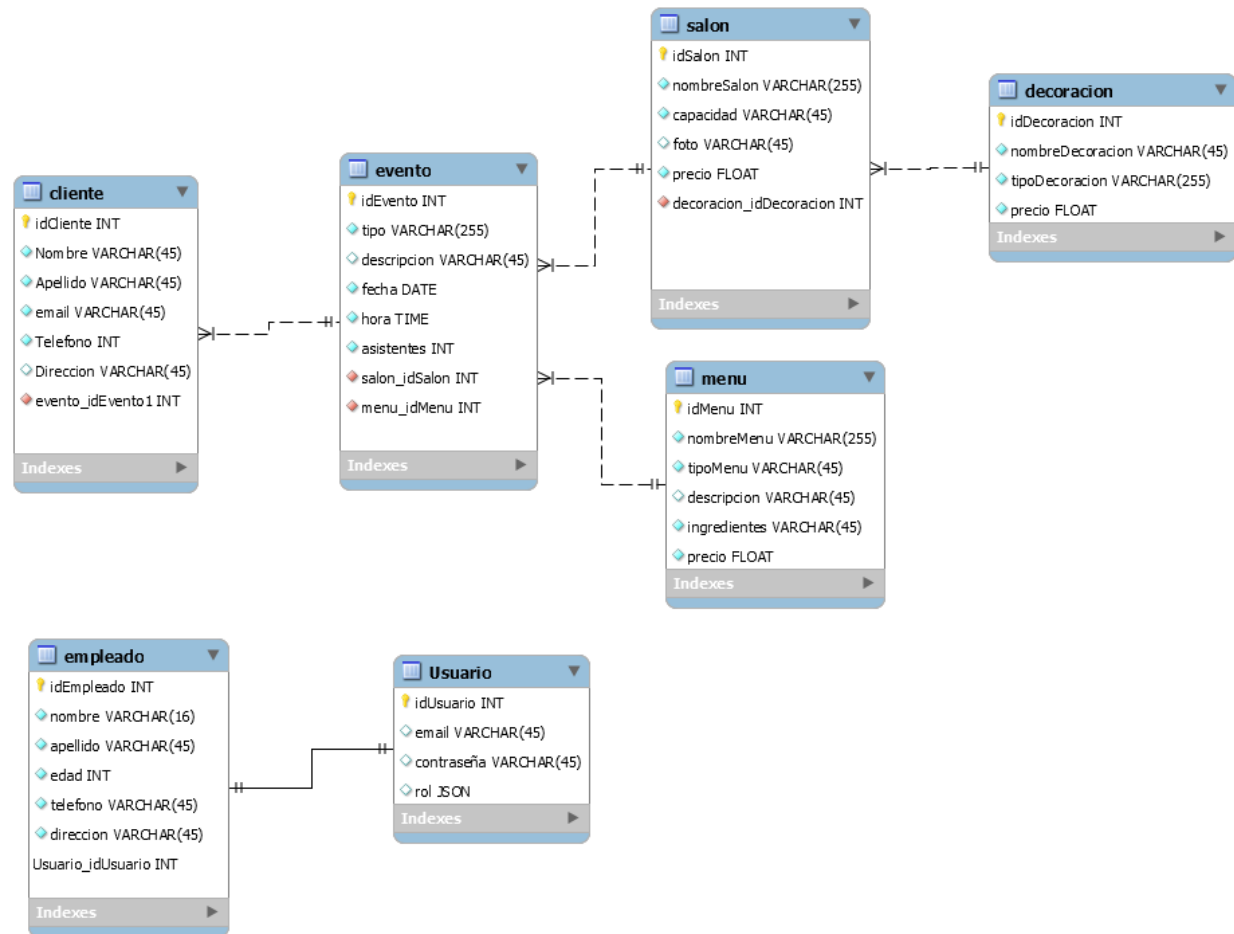
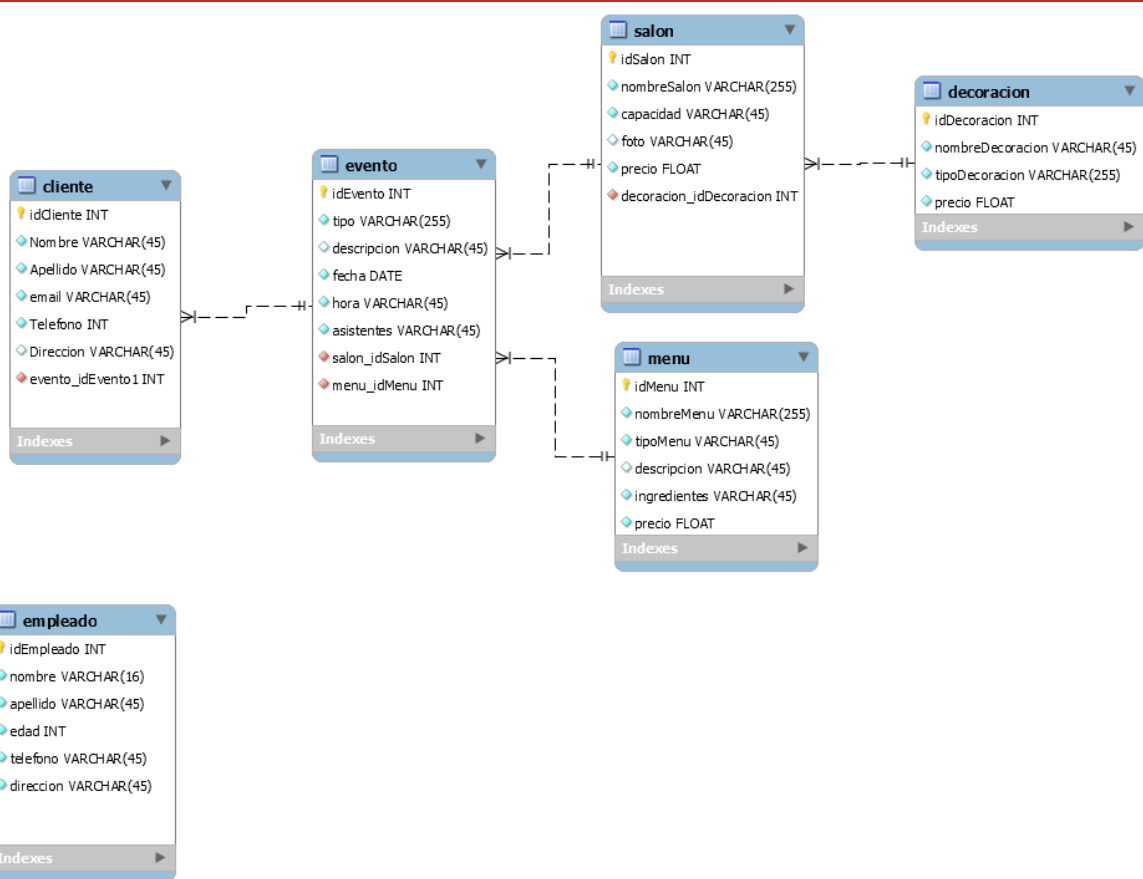


ETAPAS DEL PROYECTO

ETAPAS



DISEÑO



DESARROLLO

```
$user = new User();  
$form = $this->createForm(UserType::class, $user);  
$form->handleRequest($request);
```

```
if ($form->isSubmitted() && $form->isValid()) {  
    $entityManager = $this->getDoctrine()->getManager();  
    /*  
    El tipo de usuario, no lo define el nuevo usuario al registrarse, se asigna automaticamente  
    con el rol user lo cual se esta declando desde la entidad por medio del constructor (Buenas practicas)  
    ya que solo puede haber un usuario con el rol administrador (ROLE_ADMIN).  
    */  
    //comentamos esta línea ya que ahora la controla el constructor en la entity  
    //$user->setRoles(['ROLE_USER']);  
    //Encriptado de contraseña  
    $user->setPassword($passwordEncoder->encodePassword($user, $form['password']->getData()));  
    //Guardar los datos  
    $entityManager->persist($user);  
    $entityManager->flush();  
    //Mensaje de guardado con exito se muestra en el index de usuario se empleo una constante declarada en la entity  
    $this->addFlash('realizado', User::REGISTRO_EXITOSO);  
    return $this->redirectToRoute('user_index');  
}  
  
return $this->render('user/new.html.twig', [  
    'user' => $user,  
    'form' => $form->createView(),  
]);
```

```
public function __construct()  
{  
    $this->roles = ['ROLE_USER'];  
}
```

```
13 class User implements UserInterface  
14 {  
15     const REGISTRO_EXITOSO = "Se ha registrado Exitosamente";  
16 }
```

```

if ($fotoFile) {
    $originalFilename = pathinfo($fotoFile->getClientOriginalName(), PATHINFO_FILENAME);
    // esto es necesario para incluir de manera segura el nombre del archivo como parte de la URL
    $safeFilename = $slugger->slug($originalFilename);
    $nuevaFoto = $safeFilename . '-' . uniqid() . '.' . $fotoFile->guessExtension();
}

```

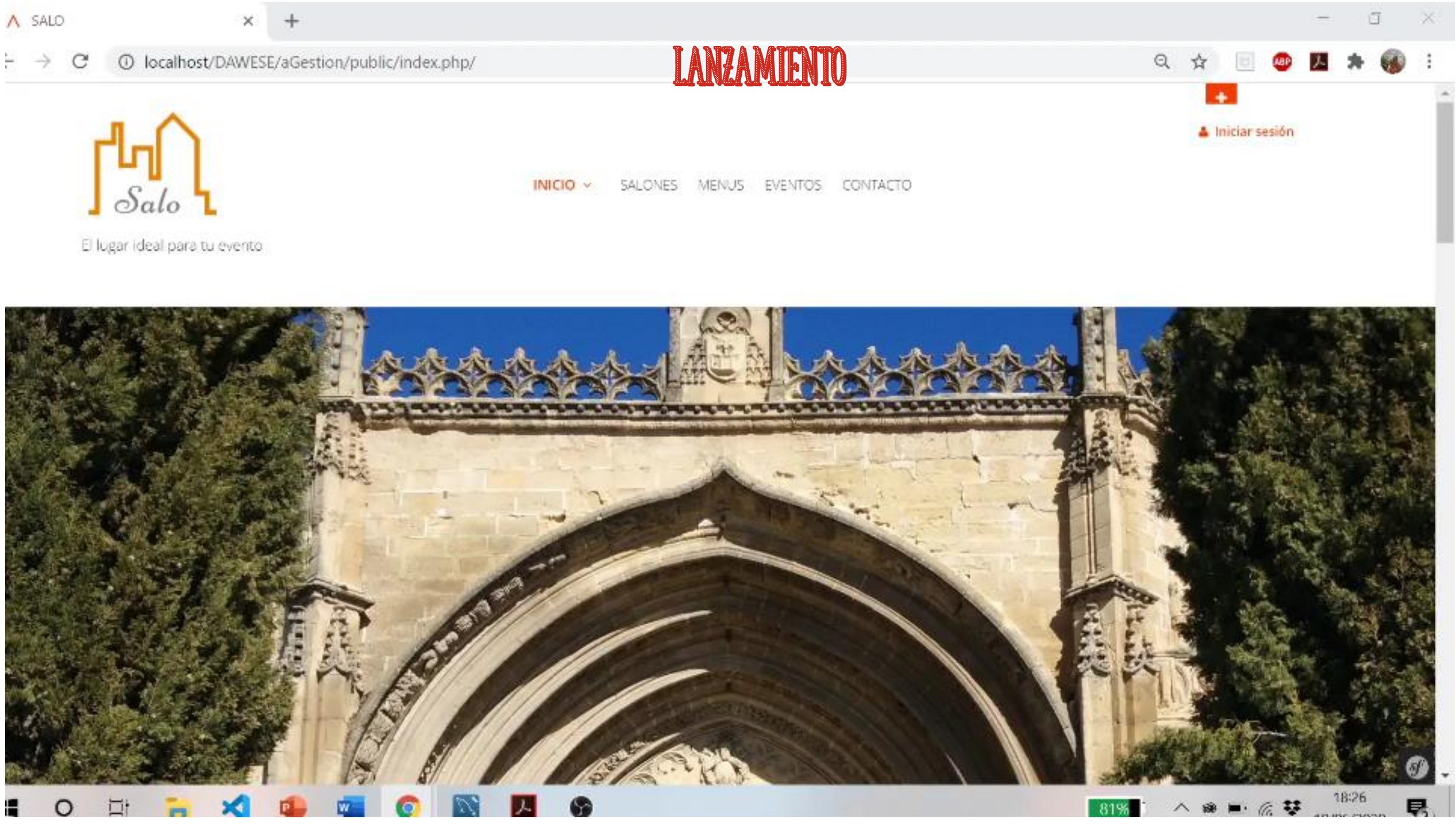
```

1  $(window).ready(function(){
2      $('input[type="file"]').change(function(ev){
3          ev.preventDefault();
4          var file=ev.target.files[0];
5          var imgMini=document.getElementById('imgMini');
6          imgMini.src=window.URL.createObjectURL(file);
7          console.log(file);
8      })
9  })

```

img

- > bodybg
- > dummies
- > slides
- > works
- avatar.png
- cargando.gif
- glyphicons-halflings.png
- glyphicons-halflings-white.png
- icoLogo.png
- IMG-20181027-WA0066-5ea6ec99f12ea.jpeg
- logo.png





FUNCIONALIDADES

- Versatilidad
- Capacidad de uso
- Seguridad



RECURSOS UTILIZADOS

- Recursos humanos
- Recursos materiales
- Recursos software



CRITERIOS DE CALIDAD

- Confiabilidad
- Usabilidad
- Seguridad
- Disponibilidad

PROBLEMAS ENCONTRADOS

Desarrollo y el modelo de la base de datos.

MEJORAS

- Identificar causas que provoquen alguna debilidad en la funcionalidad de la aplicación.
- Al contratar el servicio, que el cliente, pueda ir visualizando los avances de su evento hasta la fecha acordada.
- Crear una encuesta de satisfacción al cliente.



GRACIAS.
