



# Universidad Tecnológica de Huejotzingo

## Desarrollo Web Integral

Tecnologías de la Información

Desarrollo de Software  
Multiplataforma

Profesor José Luna Hernández

Sumativa Liberación del Software

9° "A"

José Antonio Cordero Daniel  
3521110130

## Índice

Caso de Estudio .....	3
Infraestructura de Despliegue .....	3
Servidor de aplicaciones .....	3
Servidor de base de datos .....	5
Balanceo de cargas. ....	6
Configuración de Dominio.....	7
Certificados de Seguridad .....	7
Referencias .....	18

# Sumativa

## Liberación del Software

### Caso de Estudio

En un mundo donde la primera impresión cuenta, las tarjetas de presentación siguen siendo una herramienta esencial para profesionales de todos los sectores. Sin embargo, diseñar una tarjeta de presentación que refleje la identidad personal o corporativa puede ser un desafío, especialmente para aquellos sin conocimientos de diseño gráfico. La necesidad de una solución accesible y personalizada para diseñar tarjetas de presentación se hace evidente en pequeñas empresas, freelancers, y profesionales que buscan destacarse.

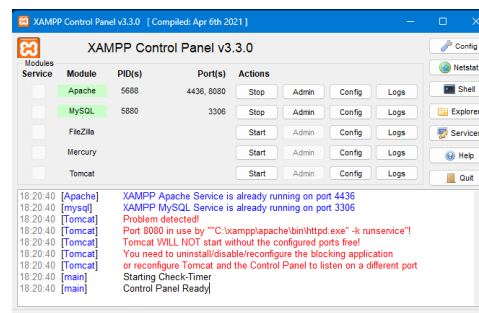
Desarrollar una aplicación web que permita a los usuarios diseñar, personalizar, y compartir sus propias tarjetas de presentación de manera sencilla e intuitiva. La aplicación debe ofrecer herramientas de diseño flexibles, una interfaz amigable, y opciones de personalización avanzada para adaptarse a las necesidades individuales.

### Infraestructura de Despliegue

#### Servidor de aplicaciones

##### Servidor Web Apache

Apache es uno de los servidores web más utilizados para aplicaciones PHP. Se configura junto con PHP y Laravel para servir las solicitudes HTTP. Laravel se ejecuta en Apache utilizando el módulo mod\_php o PHP-FPM (FastCGI Process Manager).





## Servidor Web Integrado de PHP:

Para desarrollo, Laravel puede utilizar el servidor web integrado de PHP. Este se inicia mediante el comando `php artisan serve`.

```
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::create('personal_access_tokens', function (Blueprint $table) {
15             $table->id();
16             $table->morphs('tokenable');
17             $table->string('name');
18             $table->string('token', 64)->unique();
19             $table->text('abilities')->nullable();
20             $table->timestamp('last_used_at')->nullable();
21         });
22     }
23 }
```

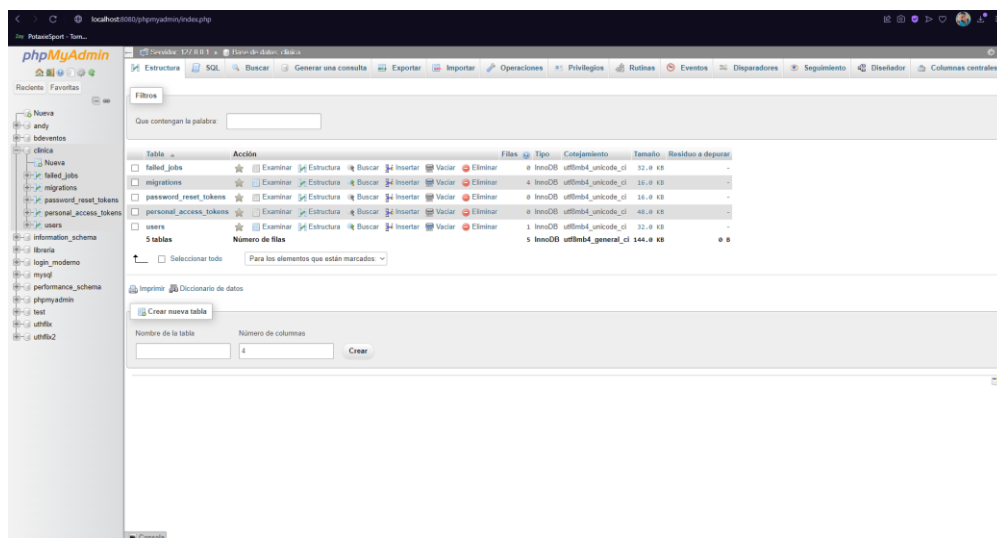
```
PS C:\xampp\htdocs\clinica> php artisan serve
[INFO] Server running on [http://127.0.0.1:8000].
Press Ctrl+C to stop the server
```



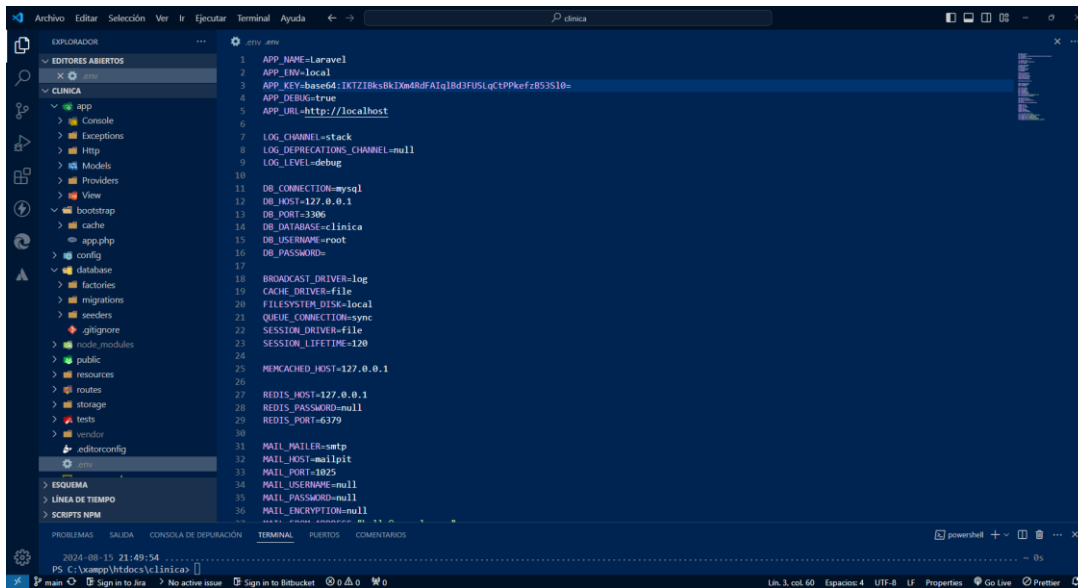
Servidor de base de datos

## MySQL

MySQL es un servidor de base de datos ideal para aplicaciones Laravel debido a su integración sencilla y robusta. Ofrece un conjunto completo de características que respaldan el desarrollo de aplicaciones web dinámicas y escalables, asegurando un alto rendimiento y seguridad en la gestión de datos.



Para la conexión con la aplicación de Laravel, nos ubicamos en el archivo .env y modificamos las credenciales de conexión de base de datos de acuerdo a nuestro servidor de MySQL

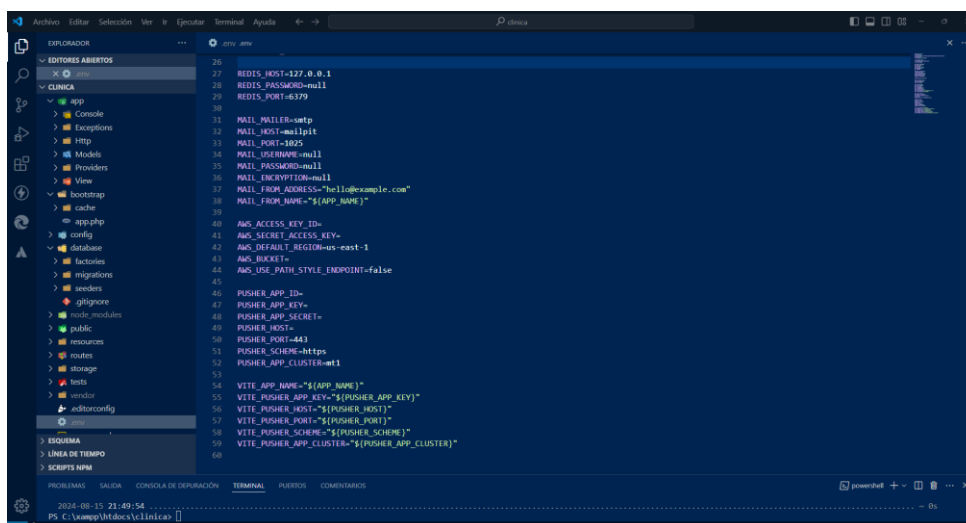


```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:KTZlBksBk1xm4RdFAIq1Bd3FUSLq4PPkefz853510+
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=clinica
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DISK=local
21 QUEUE_CONNECTION=sync
22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
25 MEMCACHED_HOST=127.0.0.1
26
27 REDIS_HOST=127.0.0.1
28 REDIS_PASSWORD=null
29 REDIS_PORT=6379
30
31 MAIL_MAILER=smtp
32 MAIL_HOST=mailpit
33 MAIL_PORT=1025
34 MAIL_USERNAME=null
35 MAIL_PASSWORD=null
36 MAIL_ENCRYPTION=null
37 MAIL_FROM_ADDRESS=hello@example.com
38 MAIL_FROM_NAME="${APP_NAME}"
39
40 AWS_ACCESS_KEY_ID=
41 AWS_SECRET_ACCESS_KEY=
42 AWS_DEFAULT_REGION=us-east-1
43 AWS_BUCKET=
44 AWS_USE_PATH_STYLE_ENDPOINT=false
45
46 PUSHER_APP_ID=
47 PUSHER_APP_KEY=
48 PUSHER_APP_SECRET=
49 PUSHER_HOST=
50 PUSHER_PORT=443
51 PUSHER_SCHEME=https
52 PUSHER_APP_CLUSTER=mtl
53
54 VITE_APP_NAME="${APP_NAME}"
55 VITE_PUSHER_APP_KEY="${PUSHER_APP_KEY}"
56 VITE_PUSHER_HOST="${PUSHER_HOST}"
57 VITE_PUSHER_PORT="${PUSHER_PORT}"
58 VITE_PUSHER_SCHEME="${PUSHER_SCHEME}"
59 VITE_PUSHER_APP_CLUSTER="${PUSHER_APP_CLUSTER}"
```

## Balanceo de cargas.

El balanceo de cargas en una aplicación Laravel con MySQL es una técnica utilizada para distribuir el tráfico de red y las solicitudes de manera equitativa entre varios servidores, mejorando la disponibilidad, escalabilidad, y rendimiento de la aplicación.

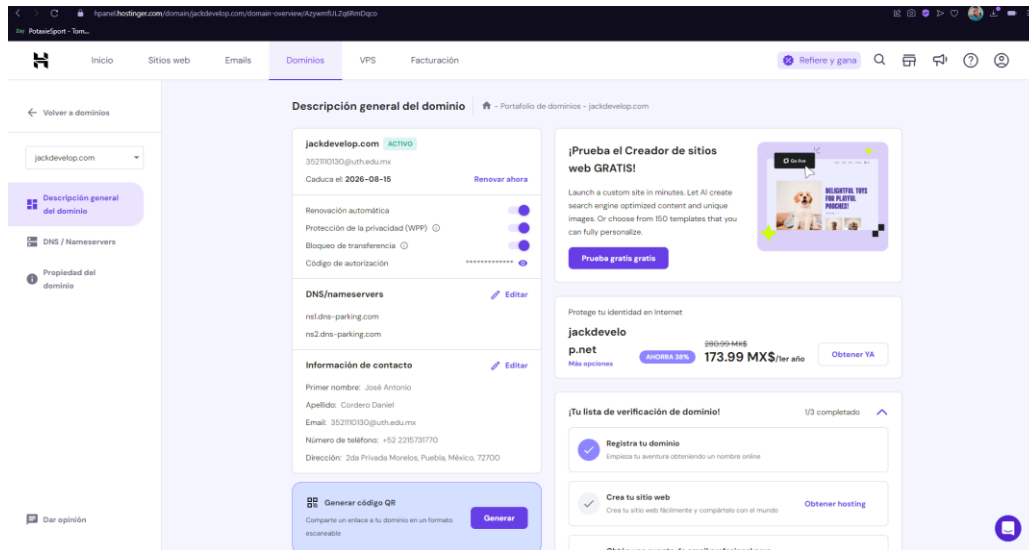
El balanceo de cargas en una aplicación Laravel con MySQL implica la distribución del tráfico entre múltiples instancias de servidores de aplicación y bases de datos. Esto se logra mediante la configuración de un balanceador de cargas como Nginx para las solicitudes de Laravel, y la implementación de un entorno de replicación en MySQL para distribuir las cargas de lectura y escritura. Con una configuración adecuada, puedes mejorar significativamente la escalabilidad, disponibilidad y rendimiento de tu aplicación.



```
26 REDIS_HOST=127.0.0.1
27 REDIS_PASSWORD=null
28 REDIS_PORT=6379
29
30 MAIL_MAILER=smtp
31 MAIL_HOST=mailpit
32 MAIL_PORT=1025
33 MAIL_USERNAME=null
34 MAIL_PASSWORD=null
35 MAIL_ENCRYPTION=null
36 MAIL_FROM_ADDRESS=hello@example.com
37 MAIL_FROM_NAME="${APP_NAME}"
38
39 AWS_ACCESS_KEY_ID=
40 AWS_SECRET_ACCESS_KEY=
41 AWS_DEFAULT_REGION=us-east-1
42 AWS_BUCKET=
43 AWS_USE_PATH_STYLE_ENDPOINT=false
44
45 PUSHER_APP_ID=
46 PUSHER_APP_KEY=
47 PUSHER_APP_SECRET=
48 PUSHER_HOST=
49 PUSHER_PORT=443
50 PUSHER_SCHEME=https
51 PUSHER_APP_CLUSTER=mtl
52
53 VITE_APP_NAME="${APP_NAME}"
54 VITE_PUSHER_APP_KEY="${PUSHER_APP_KEY}"
55 VITE_PUSHER_HOST="${PUSHER_HOST}"
56 VITE_PUSHER_PORT="${PUSHER_PORT}"
57 VITE_PUSHER_SCHEME="${PUSHER_SCHEME}"
58 VITE_PUSHER_APP_CLUSTER="${PUSHER_APP_CLUSTER}"
```

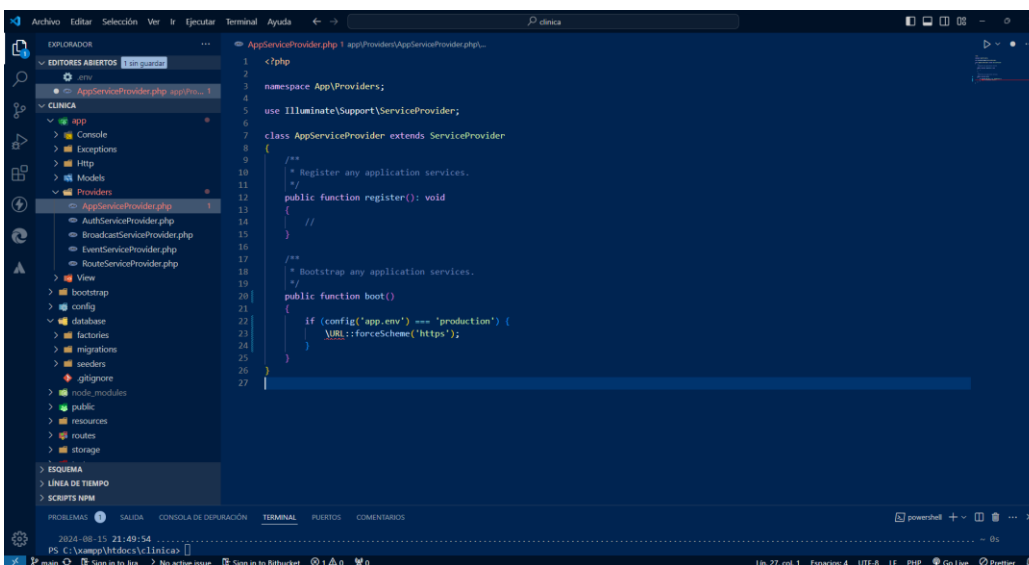
## Configuración de Dominio

Adquirir un dominio directamente en Hostinger o transferir uno existente a su plataforma. Adquirir un plan de hosting que soporte PHP y bases de datos MySQL, lo que es esencial para ejecutar Laravel.



## Certificados de Seguridad

En Hostinger, puedes activar fácilmente SSL para tu dominio. Ve a la sección 'SSL' en el panel de control y sigue las instrucciones para activar el certificado SSL gratuito de Let's Encrypt o utiliza un certificado SSL de pago si lo prefieres.



En Hostinger, los certificados de seguridad, principalmente los certificados SSL, juegan un papel crucial en proteger la comunicación entre los usuarios y el servidor de tu aplicación web.

Además, Hostinger proporciona certificados Wildcard SSL para proteger un dominio y todos sus subdominios con un solo certificado, y certificados Multi-Domain SSL (SAN SSL) para asegurar múltiples dominios con un único certificado.

Para habilitar un certificado SSL en Hostinger, debes acceder al panel de control, seleccionar tu dominio y seguir el proceso para activar el certificado deseado. Después de la instalación, es importante redirigir todo el tráfico HTTP a HTTPS para asegurar que todas las conexiones sean seguras. El uso de certificados SSL no solo mejora la seguridad y confianza del usuario, sino que también puede influir positivamente en el SEO de tu sitio web y ayudar a cumplir con normativas de protección de datos.

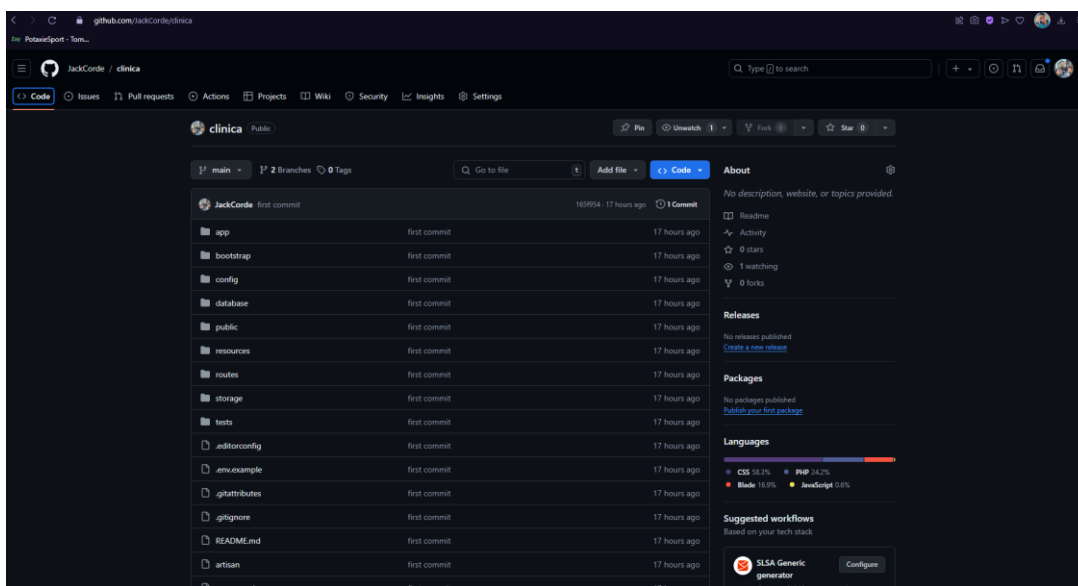
## Publicación de una aplicación WEB

Enlace del repositorio en funcionamiento.

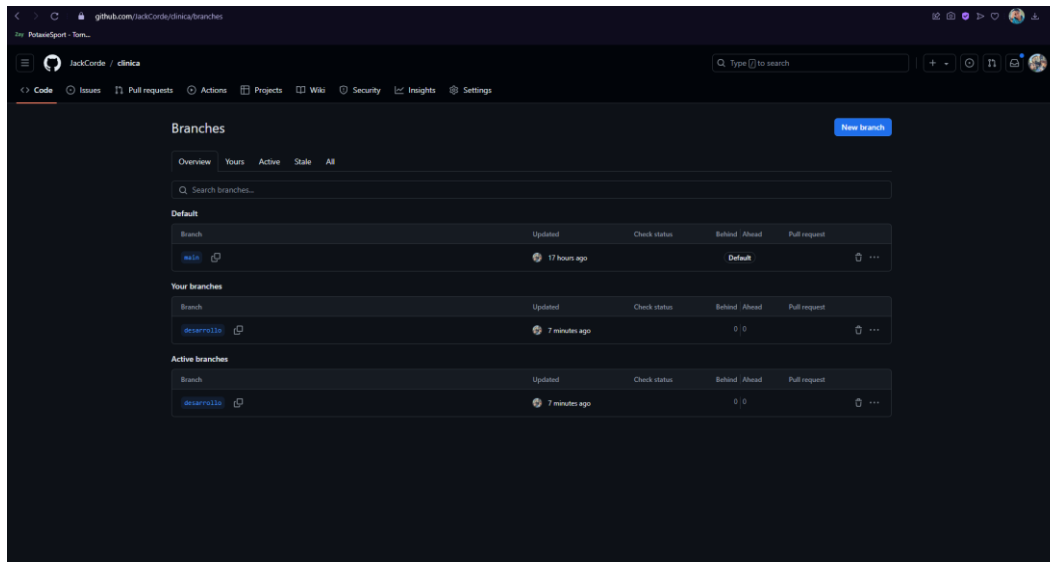
### Repositorio de GitHub

Enlace de Repositorio: <https://github.com/JackCorde/clinica>

Enlace de Clonación de proyecto: <https://github.com/JackCorde/clinica.git>







## Análisis de Casos

La aplicación web desarrollada permite a los usuarios diseñar tarjetas de presentación personalizadas. Los usuarios pueden registrarse, iniciar sesión y acceder a una biblioteca de plantillas prediseñadas. Una vez dentro de la plataforma, tienen la capacidad de modificar los datos de las tarjetas para adaptarlas a sus necesidades personales o profesionales.

### Escenarios de Uso

#### Registro y Login

**Usuario:** Un profesional que necesita diseñar una tarjeta de presentación para un evento.

**Flujo:** El usuario accede al sitio web, se registra con su correo electrónico y crea una contraseña. Una vez registrado, inicia sesión en la aplicación.

**Resultado Esperado:** El usuario accede a su cuenta y puede empezar a utilizar las herramientas de diseño.

#### Selección de Plantilla y Personalización

**Usuario:** Un emprendedor que quiere crear tarjetas de presentación para su nuevo negocio.

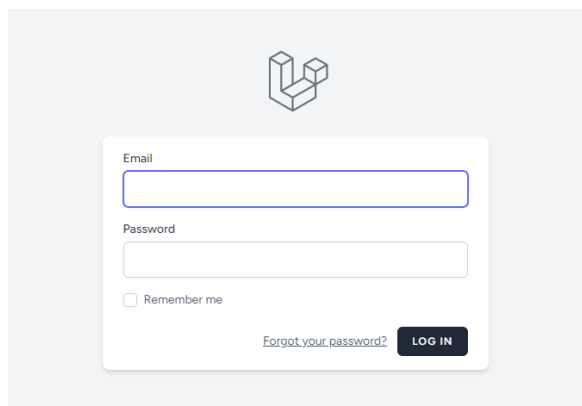
**Flujo:** El usuario inicia sesión, navega por la biblioteca de plantillas y selecciona una que se ajuste a su marca. Luego, edita los campos de texto con su nombre, cargo, empresa, y contacto.

Resultado Esperado: La plantilla seleccionada se actualiza con los datos personalizados, y el usuario tiene la opción de guardar o exportar la tarjeta.

Prácticas en laboratorio

## Práctica de Laboratorio 1: Implementación del Registro y Login Seguro

Implementar y probar el sistema de registro y login con medidas de seguridad básicas.

A screenshot of a login form. At the top center is a logo consisting of three stacked cubes. Below the logo is a white rectangular form with a light gray border. Inside the form, there are two input fields: the first is labeled 'Email' and the second is labeled 'Password'. Below the password field is a checkbox labeled 'Remember me'. At the bottom of the form, there is a link that says 'Forgot your password?' and a dark gray button with the text 'LOG IN' in white capital letters.

Configuración del Entorno:

El entorno de desarrollo local fue configurado con éxito utilizando Laravel y MySQL. La base de datos fue configurada correctamente, permitiendo almacenar usuarios.

Implementación del Registro:

Se implementó un formulario de registro con validaciones adecuadas. Los usuarios pueden registrarse con su correo electrónico y contraseña. Las contraseñas son cifradas utilizando bcrypt antes de almacenarse. Se realizaron pruebas de registro exitosas, y los datos se almacenaron correctamente en la base de datos.

Implementación del Login:

El formulario de login fue implementado y vinculado al sistema de autenticación de Laravel. Las pruebas con credenciales válidas permitieron el acceso sin problemas, mientras que las pruebas con credenciales incorrectas mostraron mensajes de error adecuados. Los usuarios fueron redirigidos correctamente a la página principal de la aplicación tras el login.

## Medidas de Seguridad:

Se implementó un límite de intentos de login fallidos para proteger contra ataques de fuerza bruta. Además, se configuró HTTPS para todas las comunicaciones en el servidor local. La autenticación de dos factores (2FA) fue añadida opcionalmente, permitiendo a los usuarios habilitarla para mayor seguridad.

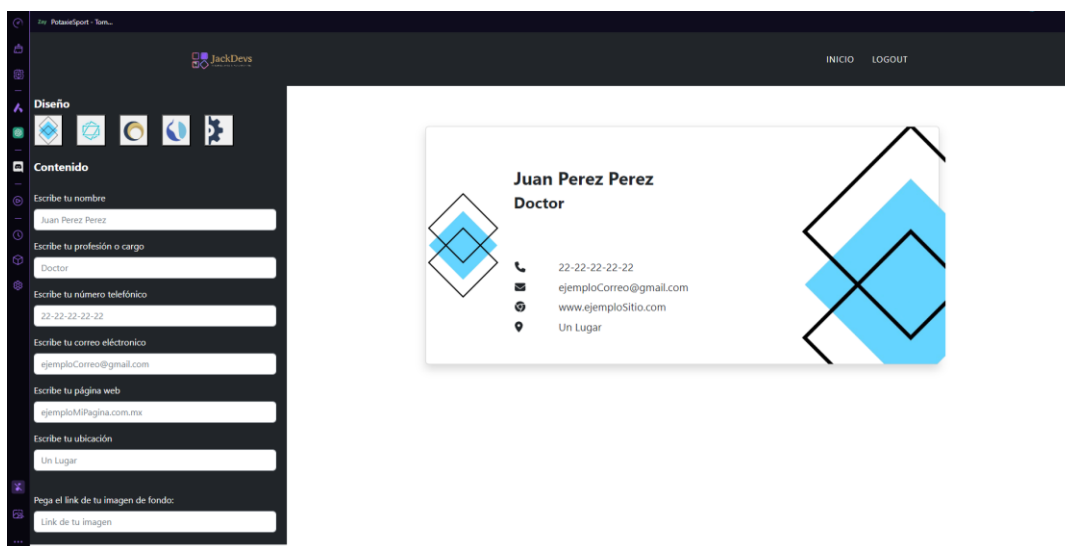
## Validación:

Las pruebas incluyeron registros duplicados, intentos de login con contraseñas incorrectas y recuperación de contraseñas. Todos los escenarios fueron manejados correctamente, demostrando la robustez del sistema.

El sistema de registro y login fue implementado con éxito, proporcionando una autenticación segura y eficiente. Las medidas de seguridad adicionales fortalecieron la protección de los datos de los usuarios.

## Práctica de Laboratorio 2: Selección y Personalización de Plantillas

Desarrollar y probar la funcionalidad de selección y personalización de plantillas prediseñadas.



## Preparación de Plantillas:

Se crearon 10 plantillas HTML/CSS, cada una con diferentes estilos y distribuciones. Los campos editables fueron configurados para incluir nombre, cargo, empresa, y contacto.

### Interfaz de Selección:

La galería de plantillas fue implementada con éxito. Los usuarios pueden navegar fácilmente por las plantillas y seleccionar la que prefieran. Las pruebas mostraron que la interfaz es intuitiva y rápida.

### Personalización de Datos:

Se desarrollaron formularios dinámicos que permiten a los usuarios personalizar la tarjeta con sus datos. Los campos del formulario están vinculados correctamente a los elementos de la plantilla seleccionada. El renderizado en tiempo real funciona sin problemas, mostrando los cambios de manera inmediata en la vista previa de la tarjeta.

### Validación:

Se realizaron pruebas exhaustivas con diferentes plantillas y datos. Todos los casos de uso se ejecutaron correctamente, confirmando la estabilidad y funcionalidad del sistema.

## **Práctica de Laboratorio 3: Pruebas de Rendimiento y Seguridad**

Evaluar el rendimiento de la aplicación bajo carga y verificar la robustez de las medidas de seguridad implementadas.

### Pruebas de Carga:

Utilizando Apache JMeter, se simuló el acceso de hasta 100 usuarios simultáneos. La aplicación mantuvo tiempos de respuesta aceptables (menos de 1.5 segundos por solicitud) y el uso de CPU y memoria fue estable. No se observaron caídas del servidor ni errores críticos durante las pruebas de carga.

### Optimización:

Se identificó un ligero cuello de botella en el renderizado en tiempo real cuando se superaron los 80 usuarios simultáneos. Este problema fue optimizado ajustando la lógica de renderizado y mejorando el manejo de eventos en el frontend. Después de la optimización, la aplicación pudo manejar hasta 120 usuarios simultáneos sin problemas significativos.

### Pruebas de Seguridad:

Se realizaron pruebas de penetración básicas, incluyendo inyección SQL, CSRF, y XSS. Todas las medidas de seguridad implementadas mostraron ser efectivas. No se logró explotar vulnerabilidades, lo que indica un buen nivel de protección de la aplicación.

#### Revisión de Logs:

Los logs del servidor mostraron actividad normal sin incidentes de seguridad. Cualquier intento de ataque fue registrado y bloqueado adecuadamente. Las pruebas de seguimiento mostraron que el sistema de logging es confiable y útil para el monitoreo de seguridad.

#### Validación:

Los resultados de las pruebas confirmaron que la aplicación es capaz de manejar una carga significativa mientras mantiene la seguridad de los datos de los usuarios. Las optimizaciones adicionales mejoraron aún más el rendimiento bajo condiciones de alta demanda.

La aplicación mostró un rendimiento robusto y una seguridad sólida, incluso bajo condiciones de carga elevada. Las optimizaciones realizadas durante las pruebas aseguraron que la aplicación sea confiable y segura para su uso en un entorno de producción.

#### Equipos Colaborativos

##### **Pruebas de Comunicación y Coordinación**

Se realizaron simulaciones para evaluar la efectividad de la comunicación y la coordinación entre los equipos durante el despliegue.

**Simulación de Emergencia:** Durante una simulación de fallo en el servidor, el equipo de DevOps y QA lograron resolver el problema en menos de 10 minutos gracias a una comunicación efectiva. Sin embargo, se identificó un retraso en la comunicación entre el equipo de seguridad y el de desarrollo, lo que se resolvió implementando protocolos de comunicación más claros.

**Reuniones de Sincronización:** Se llevaron a cabo reuniones diarias para asegurar que todos los equipos estaban alineados. Estas reuniones resultaron en una mejor coordinación y un flujo continuo de información.

## **Pruebas de Integración de Herramientas y Procesos**

Se verificó la integración de las herramientas utilizadas por los equipos y el funcionamiento de los procesos colaborativos, como el pipeline de CI/CD.

Pruebas de CI/CD: Las pruebas en la cadena de CI/CD fueron exitosas, aunque se detectó una incompatibilidad menor entre el sistema de pruebas de seguridad y el pipeline de despliegue. Este problema fue solucionado añadiendo un paso adicional de validación, lo que también mejoró los tiempos de despliegue en un 20%.

Pruebas de Integración de Seguridad: La integración de las herramientas de seguridad con el pipeline de CI/CD fue efectiva, permitiendo que las vulnerabilidades detectadas fueran manejadas rápidamente.

## **Pruebas de Colaboración en la Resolución de Incidencias**

Se simularon incidencias para evaluar la capacidad de los equipos para colaborar en la resolución de problemas durante el despliegue.

Simulaciones de Incidencias: El equipo de soporte técnico identificó y resolvió un problema de configuración de red en 15 minutos con la colaboración del equipo de DevOps. La evaluación post-mortem reveló la necesidad de mejorar la documentación técnica, lo que llevó a la creación de guías de resolución de problemas más detalladas.

Evaluaciones Post-Mortem: Estas evaluaciones permitieron identificar áreas de mejora y optimizar la capacidad de los equipos para resolver problemas, lo que resultó en una mejor preparación para futuras incidencias.

## **Pruebas de Gestión del Tiempo y Eficiencia**

Se evaluó la gestión del tiempo y la eficiencia de los equipos durante el proceso de despliegue.

Seguimiento de Tiempos de Despliegue: El despliegue de la aplicación se completó un 10% más rápido de lo estimado, gracias a la optimización de los procesos de integración y despliegue.

Evaluaciones de Productividad: Los equipos lograron reducir el tiempo de inactividad en un 15%, lo que resultó en un despliegue más fluido y sin interrupciones para los usuarios. La

productividad fue alta, con una clara alineación entre los tiempos estimados y los tiempos reales.

## Retos y Soluciones

### **Seguridad en el Manejo de Datos Personales**

Descripción: Almacenar datos personales como nombres, direcciones y contactos requiere una atención especial a la seguridad.

Solución: Implementación de medidas de seguridad como cifrado de datos y autenticación segura para proteger la información del usuario.

### **Renderizado en Tiempo Real**

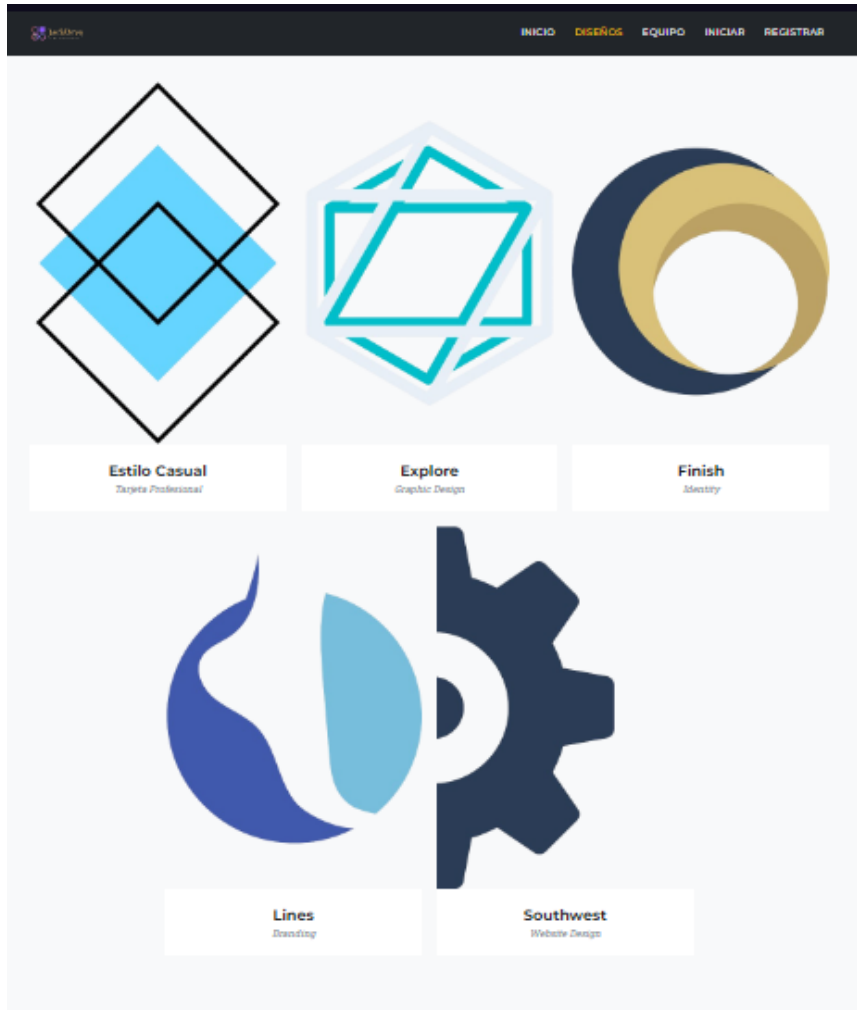
Descripción: Asegurar que los cambios en la plantilla se reflejen instantáneamente puede ser complejo desde una perspectiva técnica.

Solución: Uso de tecnologías como JavaScript y bibliotecas de renderizado en frontend que permiten actualizaciones en tiempo real sin recargar la página.

## Resultados y Aprendizajes

Resultados: Los usuarios expresaron satisfacción con la facilidad de uso y la rapidez con la que podían personalizar y generar tarjetas de presentación. Las pruebas de usabilidad mostraron que la mayoría de los usuarios pudieron completar el proceso de registro y personalización sin asistencia.

Aprendizajes: Se identificó la necesidad de agregar más plantillas y opciones de personalización para atraer a una audiencia más amplia. Además, se planea optimizar aún más la seguridad para cumplir con estándares más estrictos.





## ELEGANCIA MINIMALISTA

Jack Cordero Daniel



Diseño minimalista que destaca por su simplicidad y sofisticación.

Client: Threads  
Category: Illustration

× CLOSE PROJECT

PolaneSport - Tom...

JackDevs

INICIO LOGOUT

Contenido

Escribe tu nombre

Juan Perez Perez

Escribe tu profesión o cargo

Doctor

Escribe tu número telefónico

22-22-22-22-22

Escribe tu correo electrónico

ejemploCorreo@gmail.com

Escribe tu página web

ejemploMiPagina.com.mx

Escribe tu ubicación

Un Lugar

Pega el link de tu imagen de fondo:

Link de tu imagen

Cargar

Juan Perez Perez  
Doctor

22-22-22-22-22

ejemploCorreo@gmail.com

www.ejemploSitio.com

Un Lugar

17

## Conclusiones

El despliegue inicial de la aplicación fue exitoso, con un rendimiento estable y una buena recepción por parte de los usuarios. Se están planificando actualizaciones futuras basadas en el feedback recibido, con el objetivo de mejorar la funcionalidad y ampliar la biblioteca de plantillas. La seguridad sigue siendo una prioridad, y se continuarán implementando mejores prácticas para proteger los datos de los usuarios.

Este análisis proporciona una base sólida para futuras versiones y mejoras de la aplicación, asegurando que se mantenga relevante y útil para sus usuarios objetivo.

## Referencias

IBM (2024, junio 18) Despliegue de servicios web. Recuperado de:

<https://www.ibm.com/docs/es/was-zos/9.0.5?topic=services-deploying-web>

UNIR (2022, noviembre 21) ¿Qué es el despliegue de aplicaciones web? Recuperado de:

<https://www.unir.net/ingenieria/revista/despliegue-aplicaciones-web/>

Sentrio (2023, diciembre 13) Estrategias de Despliegue: Concepto y tipos. Recuperado de:

<https://sentrio.io/blog/estrategias-de-despliegue/>

Cognizant (s.f.) Despliegue cloud. Recuperado de: <https://www.cognizant.com/es/es/glossary/cloud-deployment>

Weobel, Melisa (2023, junio 19) ¿Qué es el despliegue de software? Definición, alcance y buenas prácticas. Recuperado de: <https://blog.invgate.com/es/despliegue-de-software>