

PLAN DE DESPLIEGUE

[Document subtitle]



**Índice**

1. [Introducción](#bookmark=id.gjdgxs)
2. [Objetivos del Plan de Despliegue](#bookmark=id.30j0zll)
3. [Requisitos Previos](#bookmark=id.1fob9te)
4. [Flujo General del Despliegue](#bookmark=id.3znysh7)
5. [Pasos Detallados del Despliegue](#bookmark=id.2et92p0)
6. [Cronograma de Actividades](#bookmark=id.tyjcwt)
7. [Diagrama Navegacional del Sistema](#bookmark=id.3dy6vkm)
8. [Conclusión](#bookmark=id.1t3h5sf)

**1. Introducción**

El sistema web implementado permite la gestión integral de empleados, clientes, citas y ventas, asegurando una experiencia optimizada tanto para trabajadores como para administradores. Este documento describe el proceso completo de despliegue del sistema, utilizando GitHub como repositorio de control de versiones y Vercel como plataforma de hosting.

**2. Objetivos del Plan de Despliegue**

* Garantizar un despliegue eficiente, seguro y escalable del sistema web.
* Facilitar actualizaciones continuas mediante la integración automatizada de cambios desde GitHub.
* Asegurar la disponibilidad del sistema para usuarios finales en un entorno de producción confiable.

**3. Requisitos Previos**

**3.1. Requisitos técnicos**

| **Requisito** | **Descripción** |
| --- | --- |
| Repositorio GitHub | Código fuente alojado en repositorio accesible |
| Cuenta en Vercel | Configurada y vinculada al repositorio |
| Framework | Next.js |
| Base de datos | Supabase |
| Lenguaje | TypeScript |

**Configuración de variables de entorno:**

* Claves para la conexión a la base de datos (Supabase URL y clave secreta)
* Cualquier otra API o clave utilizada por el sistema

**3.2. Requisitos funcionales del sistema**

* Gestión de empleados, clientes, citas, ventas y comisiones
* Registro de servicios y generación de reportes detallados
* Autenticación segura basada en correo electrónico y contraseña

**4. Flujo General del Despliegue**

**4.1. Proceso inicial**

* Configuración del repositorio en GitHub
* Configuración de Vercel como plataforma de despliegue

**4.2. Despliegue continuo**

Cada vez que se realiza un push o se aprueba un Pull Request en la rama principal, Vercel ejecuta un despliegue automático.

**5. Pasos Detallados del Despliegue**

**5.1. Configuración del Repositorio GitHub**

**Verificación del código fuente:**

* Confirma que la aplicación funciona correctamente en un entorno local
* Asegúrate de incluir un archivo .env.example con las variables de entorno necesarias

**Configuración de ramas:**

* Establece main como la rama principal para despliegues en producción
* Crea ramas separadas para desarrollo y nuevas características

**5.2. Configuración en Vercel**

**Vinculación del Repositorio:**

* Inicia sesión en Vercel y selecciona "New Project"
* Conecta el repositorio correspondiente desde GitHub

**Configuración del Proyecto:**

* Framework Preset: Next.js
* Variables de Entorno: Configuración del archivo .env.local
* Dominio Personalizado (opcional)

**5.3. Validación Previa al Despliegue**

* Pruebas de Previsualización
* Verificación de funcionalidades clave
  + Acceso al sistema
  + Gestión de empleados, clientes, citas y ventas
  + Generación de reportes

**5.4. Despliegue en Producción**

* Push a la Rama Principal
* Monitoreo del Despliegue

**5.5. Post-Despliegue**

* Pruebas finales en producción
* Configuración de alertas en Vercel

**6. Conclusión**

El despliegue del sistema en Vercel utilizando GitHub como repositorio garantiza un flujo de trabajo eficiente y automatizado. La integración continua asegura que los cambios realizados por los desarrolladores sean revisados y aplicados rápidamente al entorno de producción, manteniendo la calidad del sistema y la satisfacción del usuario final.

*Fecha de última actualización: [FECHA]*