



Medellín, Agosto 2024

Proyecto (Solución 1)	Desarrollo de la Aplicación Móvil y Portal Web para la Gestión de órdenes de servicio en campo
Director Líder Técnico	David Maestre
Director de proyecto	Maria Catalina Cardona

1. Descripción de la solución:

Desarrollo de una solución tecnológica integral que abarque tanto una aplicación móvil destinada a los técnicos en campo como un portal web que facilite la gestión administrativa desde la sede central. Esta solución permitirá a Gescomer no solo mejorar la eficiencia operativa, sino también ofrecer un servicio más ágil, seguro y personalizado a sus clientes.

El proyecto consistirá en diseñar, desarrollar e implementar una solución utilizando las tecnologías Flutter y Angular. Estas herramientas se integrarán para lograr un flujo de trabajo eficiente entre el personal de campo y la administración central. La aplicación móvil se construirá con Flutter, un framework que permite compilar aplicaciones nativas tanto para Android como para iOS, garantizando un rendimiento óptimo en ambas plataformas. Por otro lado, el portal web se desarrollará en Angular, una plataforma robusta para construir aplicaciones web modernas. Además, utilizaremos la API de Google Maps para gestionar los mapas. Para asegurar la funcionalidad offline, almacenaremos las coordenadas y rutas localmente en el dispositivo mediante local storage.

Integración entre la App Móvil y el Portal Web:

- **Sincronización de Datos:** La app móvil y el portal web están continuamente sincronizados, asegurando que cualquier actualización en el campo sea reflejada inmediatamente en el sistema administrativo y viceversa. Esto permite una gestión fluida y sin interrupciones de las órdenes de servicio.
- **Trabajo Offline:** Para situaciones en las que el técnico no tiene acceso a internet, la app móvil descargará previamente las coordenadas y rutas necesarias, almacenándolas en local storage. Esto permitirá que el técnico continúe trabajando de manera eficiente y sin interrupciones.
- **Mapas y Navegación:** La integración con Google Maps permite que tanto en el portal web como en la app móvil se visualicen las rutas y ubicaciones en tiempo real, con la posibilidad de actualizar y manejar estas informaciones en un entorno offline para su posterior sincronización.

Seguridad y Control de Acceso: Ambas plataformas implementarán medidas de seguridad robustas, incluyendo autenticación OAuth 2.0, encriptación de datos y control de acceso basado en roles, asegurando que solo el personal autorizado pueda acceder a información sensible.

2. **Objetivo:**

Desarrollar e implementar una aplicación móvil en Flutter y un portal web en Angular para Gescomer, que permitan la gestión integral de órdenes de servicio, desde la recepción y asignación hasta su ejecución y cierre, integrando funcionalidades de autenticación, sincronización de datos, gestión de inventarios, notificaciones, mapas con Google, y almacenamiento local de datos para optimizar la operatividad y mejorar la experiencia del cliente.

3. **Descripción de proceso en la aplicación móvil:**

a. **Autenticación y Acceso (Fase Previa a la Recepción de la Orden de Servicio)**

Antes de que el técnico pueda interactuar con la aplicación, se debe asegurar que esté autenticado de manera segura. La autenticación se realiza utilizando OAuth 2.0, con opciones de autenticación biométrica (como huella digital o Face ID) para dispositivos compatibles. Esto garantiza que solo usuarios autorizados puedan acceder y gestionar las órdenes de servicio.

Sub-pasos:

- ✓ Implementación de OAuth 2.0 para la autenticación segura.
- ✓ Generación de tokens de acceso para la gestión de sesiones.
- ✓ Uso de datos biométricos para una autenticación más rápida y segura.
- ✓ Gestión de sesiones activas, con cierre automático tras inactividad.
- ✓ Sistema de recuperación de contraseña.

b. **Recepción de la Orden de Servicio**

Una vez autenticado, el técnico recibe notificaciones a través de la app móvil sobre las órdenes de servicio asignadas. Estas notificaciones incluyen detalles como la tarea a realizar, la ubicación del cliente y los materiales necesarios. La app permite al técnico visualizar todos los detalles relevantes de la orden, incluyendo el historial de servicios previos y las notas asociadas.

Sub-pasos:

- ✓ Recepción de notificaciones push sobre nuevas órdenes de servicio.

- ✓ Visualización de detalles completos de la orden, como tareas, ubicación y materiales.
- ✓ Acceso al historial y notas previas relacionadas con la orden.

c. Ejecución de la Orden en Campo

Durante la ejecución de la orden, la app móvil ofrece funcionalidades críticas para que el técnico realice su trabajo de manera eficiente. La navegación en tiempo real, integrada con Google Maps, guía al técnico hacia la ubicación del cliente. Una vez en el sitio, el técnico puede registrar el inicio y fin del trabajo, tomar notas, capturar imágenes o videos, y actualizar el estado de la orden. Todo este contenido se sincroniza automáticamente con el portal web cuando hay conexión a internet. En situaciones de trabajo offline, la app descarga previamente las coordenadas y rutas, almacenándolas en local storage, permitiendo que el técnico acceda a la información sin necesidad de conexión.

Sub-pasos:

- ✓ Navegación en tiempo real utilizando Google Maps.
- ✓ Registro del inicio y fin del trabajo.
- ✓ Toma de notas y captura de imágenes o videos para documentar el trabajo.
- ✓ Actualización del estado de la orden (e.g., "En proceso", "Completada").
- ✓ Gestión de trabajo offline mediante la descarga previa de coordenadas y rutas, con almacenamiento en local storage.
- ✓ Sincronización diferida de datos cuando se restablece la conexión.

d. Actualización y Monitoreo

A lo largo de la ejecución de la orden, la aplicación móvil mantiene actualizada la información en tiempo real. Esta sincronización continua con el portal web permite que el equipo administrativo monitoree el progreso de las órdenes de servicio en tiempo real, asegurando una gestión eficiente y la capacidad de reaccionar rápidamente ante cualquier cambio o problema.

Sub-pasos:

- ✓ Sincronización en tiempo real de la información recopilada en campo con el portal web.
- ✓ Monitoreo por parte del equipo administrativo del progreso de las órdenes.

e. Finalización de la Orden

Al finalizar la orden, el técnico registra el tiempo de finalización y captura la firma del cliente directamente en la app. Esto formaliza la conclusión del trabajo. La aplicación genera un informe preliminar que incluye detalles como el tiempo empleado, los materiales utilizados, y cualquier incidencia registrada. Este informe se sincroniza automáticamente con el portal web para que esté disponible para revisión y facturación.

Sub-pasos:

- ✓ Registro del tiempo de finalización de la orden.
- ✓ Captura de la firma del cliente para confirmar la finalización del trabajo.
- ✓ Generación automática de un informe preliminar.
- ✓ Sincronización del informe con el portal web.

f. Comunicación y Notificaciones (Durante todo el proceso)

Durante todas las fases, la app móvil mantiene una comunicación fluida con el técnico mediante notificaciones push. También se puede integrar un sistema de mensajería interna o chat para mejorar la comunicación entre el técnico y el equipo administrativo.

Sub-pasos:

- ✓ Notificaciones push sobre cambios en las órdenes o nuevas asignaciones.
- ✓ Posible integración de un chat interno para comunicación en tiempo real.

g. Captura de Imágenes y Videos (Durante la Ejecución)

Durante la ejecución de la orden, el técnico puede capturar imágenes y videos para documentar su trabajo. Esta documentación se sincroniza automáticamente con el portal web, lo que permite su revisión inmediata por parte del equipo administrativo.

Sub-pasos:

- ✓ Captura de imágenes y videos.
- ✓ Sincronización automática de los archivos multimedia con el portal web.

h. Pruebas e Integración Final

Una vez desarrolladas todas las funcionalidades, se realiza una fase de pruebas exhaustivas para asegurar que la aplicación funcione correctamente bajo diferentes condiciones. Se realizan pruebas de funcionalidad, rendimiento y seguridad, y se ajustan los detalles finales antes del despliegue en producción.

Sub-pasos:

- ✓ Pruebas unitarias, de integración, y de rendimiento.
- ✓ Corrección de errores y ajustes finales.
- ✓ Preparación para el despliegue en producción.

4. Especificaciones técnicas:

Lenguajes y Frameworks Utilizados:

- ✓ **Aplicación Móvil:** Flutter, para desarrollo nativo en Android e iOS.
- ✓ **Portal Web:** Angular, para aplicaciones web modernas.

Integraciones Clave:

- ✓ Google Maps API para mapas y navegación.
- ✓ Firebase Cloud Messaging para notificaciones push.

Seguridad:

- ✓ Autenticación OAuth 2.0, con soporte para autenticación biométrica (huella digital y Face ID).
- ✓ Encriptación de datos tanto en tránsito como en reposo, utilizando SQLite para almacenamiento offline.

5. Disponibilidad:

- ✓ **Aplicación Móvil:** Funciona tanto online como offline, permitiendo a los técnicos continuar su trabajo sin interrupciones cuando no hay conexión a internet.
- ✓ **Portal Web:** Disponible siempre que haya conexión a internet, con sincronización en tiempo real con la aplicación móvil.

6. Integralidad:

- ✓ **Sincronización en Tiempo Real:** Todos los datos entre la app móvil y el portal web están sincronizados, asegurando que cualquier cambio se refleje en ambas plataformas sin demoras.
- ✓ **Gestión Integral de Órdenes:** Desde la recepción hasta el cierre de la orden, todo el proceso es administrado de manera centralizada, optimizando la operación y asegurando una trazabilidad completa.

7. Metodología y marcos de desarrollo:

Metodología Ágil: Desarrollo iterativo y adaptable, permitiendo ajustes rápidos según los feedbacks recibidos.

Fases de Desarrollo:

- ✓ Autenticación y acceso.
- ✓ Sincronización de datos y funcionalidad offline.
- ✓ Gestión de órdenes de servicio.
- ✓ Funcionalidades de mapas y GPS.
- ✓ Comunicación y notificaciones.
- ✓ Captura de imágenes y videos.
- ✓ Encuestas y formularios dinámicos.
- ✓ Pruebas e integración final.

8. Calidad:

- ✓ **Pruebas Exhaustivas:** Incluye pruebas unitarias, de integración, rendimiento, y seguridad para garantizar la funcionalidad bajo diversas condiciones.
- ✓ **Ajustes y Correcciones:** Se realizan ajustes antes del despliegue final para asegurar que la aplicación funcione de manera óptima.

9. Seguridad de la información:

Arquitectura:

- ✓ Uso de OAuth 2.0 para autenticación segura.

- ✓ Encriptación de datos sensibles.

Desarrollo:

- ✓ Incorporación de prácticas de desarrollo seguro, como la gestión de tokens de acceso.

Eficiencia:

- ✓ Optimización del rendimiento en la app móvil y portal web.

Escalabilidad:

- ✓ La solución es escalable, permitiendo la adición de más usuarios y funcionalidades en el futuro.

Trazabilidad:

- ✓ Todos los cambios y actualizaciones se registran, permitiendo un seguimiento completo del estado de las órdenes.

Flexibilidad:

- ✓ La aplicación puede adaptarse a nuevos requerimientos y funcionalidades según las necesidades del cliente.

Usabilidad:

- ✓ Interfaces amigables y optimizadas para facilitar el uso tanto en la app móvil como en el portal web.

10. Aseguramiento de la información:

- ✓ **Encriptación:** Datos en tránsito y reposo están protegidos.
- ✓ **Autenticación Segura:** OAuth 2.0 y autenticación biométrica.

11. Servicio de mantenimiento y soporte:

- ✓ **Soporte Técnico:** Servicio de mantenimiento disponible para asegurar la continuidad operativa.
- ✓ **Actualizaciones:** Se ofrecerán actualizaciones de seguridad y mejoras de funcionalidad según sea necesario.

12. Requerimiento de infraestructura:

- ✓ **Aplicación Móvil:** Requiere dispositivos compatibles con Flutter (Android e iOS).
- ✓ **Portal Web:** Requiere un servidor para hospedar la aplicación Angular y bases de datos en la nube o locales.

- ✓ **Conectividad:** Infraestructura de red adecuada para asegurar la sincronización entre la app móvil y el portal web.

13. Gobierno de datos:

- ✓ **Gestión de Datos:** Se sigue un protocolo estricto para el manejo, almacenamiento y protección de datos sensibles.
- ✓ **Cumplimiento Normativo:** La solución cumple con las normativas locales e internacionales de protección de datos.

14. Plan de implementación:

Fase 1: Autenticación y Acceso

En esta fase inicial, se implementarán las funcionalidades de autenticación y acceso, que son cruciales para la seguridad de la aplicación y el manejo de sesiones del usuario. Incluye el desarrollo de la autenticación mediante OAuth 2.0, el uso de datos biométricos, y la gestión de sesiones activas.

Sub-fases:

- ✓ **Implementación de OAuth 2.0:** Desarrollo de la funcionalidad que permite a los usuarios iniciar sesión de forma segura utilizando el protocolo OAuth 2.0.
- ✓ **Generación de Token:** Implementación de la generación de tokens de acceso para la gestión de sesiones seguras.
- ✓ **Datos Biométricos:** Integración de autenticación biométrica, como huella digital y Face ID, para dispositivos compatibles.
- ✓ **Gestión de Sesiones:** Desarrollo de funcionalidades para recordar sesiones activas y manejar el cierre automático después de un periodo de inactividad configurable.
- ✓ **Recuperación de Contraseña:** Implementación de un sistema que permite a los usuarios recuperar su contraseña mediante el envío de un correo electrónico con un enlace para restablecerla.

Objetivo: Asegurar que los usuarios puedan acceder a la aplicación de manera segura y cómoda, utilizando métodos modernos de autenticación y con la capacidad de gestionar sus sesiones eficientemente.

Fase 2: Sincronización de Datos y Funcionalidad Offline

Esta fase se centra en la preparación y gestión de datos tanto en línea como fuera de línea. Incluye la creación de una base de datos local, la sincronización de datos cuando se recupera la conexión a internet, y la encriptación de los datos almacenados localmente.

Sub-fases:

- ✓ **Preparación de la Base de Datos Local (SQLite):** Configuración de una base de datos local utilizando SQLite para permitir el almacenamiento y la gestión de datos offline.
- ✓ **Sincronización de Datos:** Desarrollo de la funcionalidad que permite subir y descargar datos cuando hay conexión a internet, asegurando que la información esté siempre actualizada.
- ✓ **Modo Offline:** Implementación de un sistema de almacenamiento local que permite que la app funcione sin conexión, utilizando local storage.
- ✓ **Encriptación de Datos:** Asegurar que los datos sensibles almacenados localmente estén protegidos mediante encriptación, utilizando bibliotecas como `sqlite_sqlcipher` o `encrypt`.

Objetivo: Garantizar que los técnicos puedan continuar su trabajo sin interrupciones, incluso en áreas sin conexión, y que los datos estén seguros tanto en tránsito como en reposo.

Fase 3: Gestión de Órdenes de Servicio

Esta fase se enfoca en las funcionalidades necesarias para que el técnico gestione las órdenes de servicio. Incluye la visualización y filtrado de órdenes, la actualización de su estado, y la capacidad de realizar comentarios y registrar actividades en tiempo real.

Sub-fases:

- ✓ **Visualización y Filtrado:** Implementación de interfaces que permitan a los técnicos visualizar las órdenes de servicio asignadas y filtrarlas según diferentes criterios, como tipo de orden o prioridad.
- ✓ **Actualización de Estado:** Desarrollo de funcionalidades para que los técnicos actualicen el estado de las órdenes (en proceso, completada, etc.) y registren actividades como el tiempo de inicio y finalización, incluyendo la descripción y uso de archivos multimedia.

Objetivo: Facilitar la gestión y ejecución de las órdenes de servicio por parte del técnico, asegurando que puedan acceder a toda la información relevante y actualizar el estado de las órdenes en tiempo real.

Fase 4: Funcionalidades de Mapas y GPS

Esta fase abarca la implementación de funcionalidades relacionadas con la navegación y el seguimiento de ubicaciones. Incluye la integración con Google Maps, la captura y almacenamiento de datos de ubicación, y la gestión de mapas offline.

Sub-fases:

- ✓ **Integración de Google Maps:** Implementación de la funcionalidad de navegación en tiempo real para guiar al técnico al lugar de la orden.
- ✓ **Historial de Ubicaciones:** Desarrollo de la capacidad para capturar y almacenar un historial de ubicaciones, permitiendo la revisión posterior de las rutas seguidas por los técnicos.
- ✓ **Mapas Offline:** Implementación de mapas y rutas que se pueden descargar y utilizar sin conexión, con actualización automática cuando se restablezca la conexión.

Objetivo: Proveer a los técnicos de herramientas de navegación y seguimiento precisas, que funcionen tanto en línea como fuera de línea, mejorando la eficiencia del servicio en campo.

Fase 5: Comunicación y Notificaciones

En esta fase, se implementarán las funcionalidades de comunicación y notificaciones, asegurando que el técnico esté siempre informado sobre cambios en las órdenes y pueda comunicarse efectivamente con el equipo administrativo.

Sub-fases:

- ✓ **Notificaciones Push:** Implementación de notificaciones push utilizando Firebase Cloud Messaging para alertar a los técnicos sobre nuevas órdenes, cambios de estado, y otros eventos importantes.
- ✓ **Integración de SMS y Correo Electrónico:** Desarrollo de funcionalidades para enviar y recibir comunicaciones por SMS y correo electrónico, utilizando API como Twilio para SMS y SendGrid para correos.

Objetivo: Asegurar que los técnicos estén siempre al tanto de las actualizaciones y puedan comunicarse de manera eficiente, mejorando la coordinación y el flujo de trabajo.

Fase 6: Captura de Imágenes y Videos

Esta fase se enfoca en la implementación de la funcionalidad que permite a los técnicos capturar imágenes y videos durante la ejecución de una orden de servicio. Estos archivos multimedia se sincronizan con el portal web para su revisión posterior.

Sub-fases:

- ✓ **Captura de Imágenes y Videos:** Implementación de una interfaz sencilla para que los técnicos puedan capturar y almacenar imágenes y videos directamente desde la aplicación.

- ✓ **Sincronización de Multimedia:** Asegurar que los archivos multimedia se sincronicen automáticamente con el portal web, para que estén disponibles para su revisión por parte del equipo administrativo.

Objetivo: Permitir que los técnicos documenten visualmente su trabajo, proporcionando evidencia y detalles adicionales que pueden ser útiles para la revisión y el cumplimiento de las órdenes de servicio.

Fase 7: Encuestas y Formularios Dinámicos

Esta fase abarca la creación e implementación de encuestas y formularios dinámicos que los técnicos completarán según el tipo de orden de servicio. Estos formularios ayudarán a recoger información adicional que puede ser relevante para la calidad del servicio.

Sub-fases:

- ✓ **Diseño de Encuestas:** Desarrollo de encuestas con diferentes tipos de input (fecha, lista desplegable, texto, número) y validaciones, adaptadas a las necesidades de cada orden.
- ✓ **Ejecución de Encuestas:** Implementación de la lógica que varía las encuestas según el tipo de orden, asegurando que se recoja la información adecuada en cada caso.

Objetivo: Recolectar feedback valioso y datos específicos sobre cada orden de servicio, lo que ayudará a mejorar la calidad del servicio y a tomar decisiones informadas.

Fase 8: Pruebas e Integración Final

La fase final se dedicará a las pruebas exhaustivas de la aplicación y a la integración de todas las funcionalidades desarrolladas. Se realizarán ajustes y correcciones según sea necesario para asegurar que la aplicación funcione de manera óptima antes de su despliegue.

Sub-fases:

- ✓ **Pruebas de Funcionalidad:** Realización de pruebas unitarias y de integración para verificar que cada funcionalidad de la aplicación funcione correctamente.
- ✓ **Pruebas de Rendimiento:** Evaluación de la aplicación en diferentes condiciones para asegurar que cumpla con los requisitos de rendimiento y escalabilidad.
- ✓ **Correcciones y Ajustes:** Resolución de cualquier problema detectado durante las pruebas, asegurando que la aplicación esté lista para su uso en producción.

15. Cronograma de actividades: ambas funcionalidades se realizan en paralelo

Fase / Funcionalidad	Duración Estimada	Mes 1	Mes 2	Mes 3	Mes 4
1. Autenticación y Acceso	3 semanas	X	X		
- Implementación de OAuth 2.0	1 semana	X			
- Generación de Token	1 día	X			
- Integración de Datos Biométricos (Huella/Face ID)	3 días	X			
- Gestión de Sesiones (Recordar sesión)	1 día		X		
- Cierre de Sesión Automático	½ día		X		
- Recuperación de Contraseña	1 día		X		
2. Sincronización de Datos y Funcionalidad Offline	4 semanas		X	X	
- Preparación de la Base de Datos Local (SQLite)	2 semanas		X		
- Implementación de Sincronización Offline (SQLite)	1 semana			X	
- Encriptación de Datos Locales	1 semana			X	
3. Gestión de Órdenes de Servicio	3 semanas			X	
- Visualización y Filtrado de Órdenes	2 semanas			X	
- Actualización de Estado de Órdenes	1 semana			X	
4. Funcionalidades de Mapas y GPS	4 semanas				X
- Implementación de Google Maps y Navegación	2 semanas				X
- Historial de Ubicaciones y Rutas	1 semana				X
- Mapas Offline y Descarga de Rutas (Local Storage)	1 semana				X
5. Comunicación y Notificaciones	5 semanas				X
- Notificaciones Push (Firebase Cloud Messaging)	2 semanas				X
- Integración de SMS y Correo Electrónico	3 semanas				X
6. Captura de Imágenes y Videos	2 semanas				X
- Captura de Imágenes y Videos	1 semana				X
- Sincronización con el Portal Web	1 semana				X
7. Encuestas y Formularios Dinámicos	2 semanas				X
- Diseño e Implementación de Encuestas	1 semana				X
- Ejecución de Encuestas Dinámicas	1 semana				X
8. Pruebas e Integración Final	4 semanas				X

Descripción de actividades del cronograma:

Mes 1: Semanas 1 a 4

1. Semana 1:

- **Implementación de OAuth 2.0:** Comienza el desarrollo de la autenticación de usuarios mediante el protocolo OAuth 2.0, asegurando un inicio de sesión seguro en la aplicación.
- **Generación de Token:** Desarrollo del sistema de generación de tokens de acceso para gestionar sesiones de usuarios de manera segura.

2. Semana 2:

- **Finalización de OAuth 2.0:** Se concluye la implementación de OAuth 2.0 y se realizan pruebas iniciales.
 - **Integración de Datos Biométricos (Huella/Face ID):** Se comienza la integración de métodos de autenticación biométrica como huella digital y Face ID.
3. **Semana 3:**
- **Continuación de la Integración Biométrica:** Se finaliza la implementación de datos biométricos, completando las pruebas necesarias.
 - **Gestión de Sesiones:** Se trabaja en la funcionalidad de recordar sesión activa mediante la recuperación de tokens.
4. **Semana 4:**
- **Cierre de Sesión Automático:** Implementación de la funcionalidad para cerrar sesión automáticamente después de un periodo de inactividad.
 - **Recuperación de Contraseña:** Desarrollo del sistema de recuperación de contraseña a través de correo electrónico.

Mes 2: Semanas 5 a 8

5. **Semana 5:**
- **Preparación de la Base de Datos Local (SQLite):** Se inicia la configuración y preparación de la base de datos local utilizando SQLite, lo que permitirá el almacenamiento de datos offline.
6. **Semana 6:**
- **Continuación de la Preparación de la Base de Datos:** Se continúa trabajando en la base de datos local para asegurar su correcta integración con la app.
7. **Semana 7:**
- **Sincronización Offline (SQLite):** Se implementa la funcionalidad que permite la sincronización de datos entre el dispositivo y el servidor cuando se recupera la conexión a internet.
8. **Semana 8:**
- **Encriptación de Datos Locales:** Se añaden capas de seguridad a los datos almacenados localmente mediante técnicas de encriptación para proteger la información sensible.

Mes 3: Semanas 9 a 12

9. **Semana 9:**

- **Visualización y Filtrado de Órdenes:** Desarrollo de las interfaces y funciones que permiten a los técnicos visualizar y filtrar las órdenes de servicio según diferentes criterios.

10. **Semana 10:**

- **Continuación de la Visualización y Filtrado de Órdenes:** Finalización de la implementación de estas funcionalidades, incluyendo pruebas y ajustes necesarios.

11. **Semana 11:**

- **Actualización de Estado de Órdenes:** Desarrollo de la capacidad para que los técnicos actualicen el estado de las órdenes de servicio en tiempo real.

12. **Semana 12:**

- **Inicio de Funcionalidades de Mapas y GPS:** Se comienza la implementación de Google Maps y la funcionalidad de navegación para los técnicos.

Mes 4: Semanas 13 a 16

13. **Semana 13:**

- **Implementación de Google Maps y Navegación:** Continuación del trabajo en la integración de mapas y navegación, asegurando su correcta funcionalidad tanto online como offline.

14. **Semana 14:**

- **Historial de Ubicaciones y Rutas:** Desarrollo de la funcionalidad que permite registrar y visualizar el historial de ubicaciones y rutas seguidas por los técnicos durante su trabajo.

15. **Semana 15:**

- **Mapas Offline y Descarga de Rutas (Local Storage):** Se completa la implementación para que los técnicos puedan acceder a los mapas y rutas offline, descargándolos previamente.

16. **Semana 16:**

- **Notificaciones Push (Firebase Cloud Messaging):** Implementación de notificaciones push para mantener informados a los técnicos sobre las actualizaciones de las órdenes de servicio.

17. **Semana 17:**

- **Integración de SMS y Correo Electrónico:** Integración de sistemas de comunicación por SMS y correo electrónico para notificar a los técnicos y clientes sobre el estado de las órdenes.
18. **Semana 18:**
- **Captura de Imágenes y Videos:** Desarrollo de la funcionalidad que permite a los técnicos capturar imágenes y videos desde la app, sincronizándolos con el portal web.
19. **Semana 19:**
- **Sincronización con el Portal Web:** Finalización de la integración de la app móvil con el portal web, asegurando que toda la información se sincronice correctamente.
20. **Semana 20:**
- **Diseño e Implementación de Encuestas:** Se desarrolla la funcionalidad de encuestas y formularios dinámicos que los técnicos pueden completar durante su trabajo.
21. **Semana 21:**
- **Ejecución de Encuestas Dinámicas:** Implementación de la lógica para que las encuestas varíen según el tipo de orden de servicio.
22. **Semanas 22 a 24:**
- **Pruebas e Integración Final:** Se dedica el último mes a pruebas exhaustivas de todas las funcionalidades implementadas, asegurando que la aplicación funcione correctamente en diferentes escenarios antes de su despliegue final.

16. REQUERIMIENTOS NO FUNCIONALES DE AMBAS SOLUCIONES:

Requerimientos No Funcionales App Movil

Además de los requisitos funcionales mencionados en el cronograma, se implementarán los siguientes requisitos no funcionales que son críticos para el éxito de la aplicación móvil de Gescomer:

1. Frameworks y Librerías

El desarrollo de la aplicación se basará en el patrón de arquitectura MVVM (Model-View-ViewModel) para facilitar la separación de responsabilidades, pruebas unitarias, y mantenimiento. Se utilizará Retrofit para gestionar las solicitudes HTTP hacia la API, y Room para la persistencia local de datos, garantizando un acceso rápido y eficiente a la información almacenada localmente.

2. Diagramas de Flujo y Arquitectura

Se crearán diagramas de flujo que representen el proceso completo de gestión de órdenes de servicio, desde la recepción hasta la finalización, incluyendo la interacción entre la aplicación móvil, el servidor y la base de datos local. También se elaborarán diagramas de arquitectura que detallen la interacción entre los componentes principales del sistema, incluyendo el frontend (app móvil), el backend (API REST) y el almacenamiento local.

3. Seguridad

La seguridad será una prioridad en el desarrollo de la aplicación. Se utilizarán protocolos TLS/SSL para cifrar las comunicaciones entre la aplicación móvil y el servidor. Además, los datos sensibles almacenados localmente en el dispositivo serán cifrados utilizando técnicas avanzadas de encriptación, asegurando que la información esté protegida contra accesos no autorizados.

4. Documentación y Manuales

Se proporcionará una documentación completa que incluirá un manual del sistema, documentación del código fuente, y un manual de usuario. Esto garantizará que tanto los desarrolladores como los usuarios finales tengan acceso a toda la información necesaria para el uso y mantenimiento de la aplicación.

Requerimientos No Funcionales Portal Web

1. Rendimiento

Tiempo de Respuesta: El portal web debe responder a las solicitudes de los usuarios en menos de 2 segundos en el 95% de los casos. Esto incluye la carga de páginas, la ejecución de consultas en la base de datos, y la actualización de los dashboards en tiempo real.

Capacidad de Carga: El sistema debe ser capaz de manejar múltiples usuarios concurrentes sin degradar significativamente su rendimiento. Debería soportar al menos 1000 usuarios concurrentes sin que el tiempo de respuesta supere los 3 segundos.

Optimización de Consultas: Las consultas a la base de datos deben estar optimizadas para minimizar el tiempo de procesamiento, especialmente para reportes y análisis de datos.

2. Escalabilidad

Escalabilidad Horizontal y Vertical: El portal web debe estar diseñado para ser escalable tanto horizontalmente (añadiendo más servidores) como verticalmente (mejorando la capacidad del hardware existente) para manejar un incremento en la cantidad de usuarios y datos sin comprometer el rendimiento.

Soporte para Crecimiento de Datos: La arquitectura del sistema debe permitir el crecimiento del volumen de datos almacenados sin degradar el rendimiento, utilizando bases de datos distribuidas o particionadas si es necesario.

3. Seguridad

Control de Acceso Basado en Roles (RBAC): El portal debe implementar un sistema de control de acceso que asegure que solo los usuarios autorizados puedan acceder a ciertas funcionalidades o datos. Los roles deben ser configurables y asignados según las necesidades específicas de la empresa.

Encriptación de Datos: Todos los datos sensibles, tanto en tránsito como en reposo, deben estar encriptados. Esto incluye la encriptación de comunicaciones (usando TLS) y la encriptación de datos almacenados en la base de datos.

Autenticación y Autorización: Uso de OAuth 2.0 para la autenticación segura de usuarios, junto con la autenticación de dos factores (2FA) para acceso a áreas críticas del portal.

Auditoría y Registro de Eventos: El portal debe mantener un registro detallado de todas las acciones realizadas por los usuarios, incluyendo accesos, cambios de datos, y administración de roles, para asegurar un seguimiento y auditoría efectiva.

4. Usabilidad

Interfaz de Usuario Intuitiva: El portal debe ofrecer una interfaz de usuario clara y fácil de usar, con una curva de aprendizaje baja para los administradores. Debe seguir las mejores prácticas de diseño UX/UI.

Accesibilidad: El portal debe cumplir con las normas de accesibilidad (como las WCAG 2.1) para asegurar que pueda ser utilizado por personas con diferentes capacidades.

Compatibilidad Multinavegador: El portal web debe ser completamente funcional en los principales navegadores web (Chrome, Firefox, Safari, Edge) y estar optimizado para diferentes resoluciones de pantalla.

5. Disponibilidad y Fiabilidad

Alta Disponibilidad: El portal debe estar disponible al menos el 99.9% del tiempo, excluyendo las ventanas de mantenimiento programadas. Esto requiere una arquitectura redundante y medidas de recuperación ante desastres.



Recuperación ante Fallos: En caso de fallo del sistema, debe haber mecanismos para la recuperación rápida, como backups automáticos y replicación de bases de datos, para minimizar la pérdida de datos y el tiempo de inactividad.

Monitoreo Continuo: Implementación de herramientas de monitoreo continuo para detectar y responder rápidamente a problemas de rendimiento, seguridad, o disponibilidad.

6. Mantenibilidad

Documentación Completa: Todo el código y las configuraciones del sistema deben estar bien documentados para facilitar la comprensión y el mantenimiento por parte de los equipos técnicos.

Modularidad: El portal debe estar diseñado de manera modular para que las diferentes partes del sistema puedan ser mantenidas, actualizadas, o reemplazadas de manera independiente sin afectar a otras áreas.

Facilidad de Actualización: Debe ser fácil implementar actualizaciones y parches sin interrumpir significativamente el servicio. Esto incluye la capacidad de realizar actualizaciones incrementales.

7. Portabilidad

Compatibilidad con Diferentes Plataformas: El portal debe ser compatible con diferentes sistemas operativos y entornos de hosting, permitiendo su despliegue en servidores locales o en la nube, dependiendo de las necesidades de la empresa.

Despliegue en la Nube: Debe ser posible desplegar el portal en entornos de nube como AWS, Azure, o Google Cloud, con opciones para escalabilidad automática y administración de recursos.

8. Integración

Integración con la Aplicación Móvil: El portal debe estar completamente integrado con la app móvil, asegurando que toda la información relevante (órdenes de servicio, datos de ubicación, comunicaciones) se sincronice de manera efectiva entre ambos sistemas.

APIs Restful: El portal debe ofrecer APIs bien documentadas y seguras para la integración con otros sistemas, facilitando la interoperabilidad y la extensión de funcionalidades.

ANEXO TÉCNICO DESCRIPTIVO PAGINA WEB

1. OBJETIVO:

El portal web de Gescomer está diseñado como una plataforma centralizada para la gestión integral de las órdenes de servicio, la coordinación de los técnicos en campo, y la administración de los recursos de la empresa. Su objetivo principal es proporcionar una interfaz administrativa robusta que permita a los gestores de Gescomer supervisar, controlar, y optimizar las operaciones de servicio técnico, asegurando que todas las órdenes de servicio se manejen de manera eficiente y efectiva.

2. DESCRIPCIÓN:

a. Centralización de la Gestión de Órdenes de Servicio:

El portal web centraliza la recepción, asignación, seguimiento y finalización de todas las órdenes de servicio. Esto permite a los administradores tener una visión completa y en tiempo real de todas las actividades de servicio técnico que están en curso, así como de las que ya han sido completadas.

b. Coordinación y Monitoreo en Tiempo Real:

Facilitar la coordinación de los técnicos en campo, permitiendo a los administradores asignar tareas, monitorizar el progreso, y reaccionar rápidamente ante cualquier problema o incidencia. El portal web ofrece herramientas de monitoreo en tiempo real, incluyendo la visualización de rutas y el seguimiento del estado de las órdenes.

c. Optimización de Recursos y Eficiencia Operativa:

Proporcionar información detallada sobre el rendimiento de los técnicos, los tiempos de respuesta, y el uso de materiales. Esto permite a los administradores optimizar la asignación de recursos y mejorar la eficiencia operativa.

d. Generación de Reportes y Análisis de Datos:

El portal web permite generar reportes detallados sobre diversos aspectos del servicio técnico, lo que facilita la toma de decisiones basada en datos. Los administradores pueden analizar tendencias, identificar áreas de mejora, y realizar ajustes estratégicos en las operaciones de la empresa.

e. Automatización de la Facturación y Gestión de Clientes:

El portal web integra la funcionalidad de facturación, generando automáticamente facturas basadas en los servicios prestados. Además, gestiona la información de los clientes, incluyendo el historial de servicios y la facturación, facilitando la administración financiera y la relación con los clientes.

f. Seguridad y Control de Acceso:

Implementar medidas de seguridad robustas, asegurando que solo el personal autorizado pueda acceder a información sensible. Esto incluye el control de acceso basado en roles (RBAC) y la encriptación de datos.

3. Funcionalidad del Portal Web de Gescomer

El portal web se divide en varios módulos clave, cada uno con funcionalidades específicas que permiten a los administradores gestionar y monitorizar las operaciones de manera efectiva. A continuación, se describen las principales funcionalidades del portal web:

a. Gestión de Órdenes de Servicio

- **Recepción y Clasificación de Órdenes:** El portal permite registrar órdenes de servicio, ya sea de manera manual o automática. Los administradores pueden clasificar las órdenes según diferentes criterios, como tipo de servicio, urgencia, o ubicación.

- **Asignación de Técnicos:** Una vez que se recibe una orden, los administradores pueden asignarla a un técnico específico. La asignación se basa en factores como la disponibilidad del técnico, su ubicación, y sus habilidades.

- **Seguimiento en Tiempo Real:** A medida que los técnicos trabajan en las órdenes, el portal web permite a los administradores monitorizar el progreso en tiempo real. Los cambios de estado, como “En proceso” o “Completada”, se reflejan automáticamente en el portal, proporcionando una visión actualizada de todas las actividades en campo.

b. Monitoreo y Seguimiento de Actividades

- **Dashboard de Monitoreo:** El portal web incluye un panel de control que ofrece una visión general de todas las órdenes de servicio activas, mostrando indicadores clave como tiempos de respuesta, eficiencia de los técnicos, y cualquier incidencia registrada.

- **Mapa de Rutas:** Integrando los datos de ubicación proporcionados por la app móvil, el portal web permite a los administradores visualizar las rutas que han seguido los técnicos. Esto es crucial para optimizar la logística y asegurar que los técnicos lleguen a sus destinos de manera eficiente.

- **Gestión de Incidencias:** Si un técnico encuentra problemas durante la ejecución de una orden, puede reportar incidencias a través de la app móvil, las cuales se registran y gestionan desde el portal web. Los administradores pueden tomar decisiones rápidas para resolver estos problemas, minimizando el impacto en el servicio.

c. Comunicación y Notificaciones



- **Envío de Notificaciones Push:** El portal permite enviar notificaciones push directamente a los dispositivos móviles de los técnicos. Estas notificaciones pueden incluir nuevas asignaciones de órdenes, cambios en las órdenes existentes, o alertas sobre problemas urgentes.

- **Mensajería Interna:** El portal puede incluir un sistema de mensajería interna que permite la comunicación directa entre administradores y técnicos. Esto facilita la coordinación en tiempo real, permitiendo que los técnicos reciban instrucciones adicionales o resuelvan dudas rápidamente.

4. Reportes y Análisis de Datos

- **Generación de Reportes:** Los administradores pueden generar reportes automáticos o personalizados que proporcionan información detallada sobre diversos aspectos del servicio técnico, como tiempos de respuesta, materiales utilizados, y niveles de satisfacción del cliente.

- **Análisis de Datos:** El portal web incluye herramientas de análisis de datos que permiten a los administradores identificar tendencias y patrones en las operaciones. Esto ayuda a tomar decisiones informadas para mejorar la eficiencia y calidad del servicio.

5. Facturación y Gestión de Clientes

- **Generación Automática de Facturas:** Basado en los datos recopilados durante la ejecución de las órdenes de servicio, el portal web genera automáticamente facturas que se pueden enviar a los clientes por correo electrónico.

- **Gestión de Clientes:** El portal mantiene un registro detallado de cada cliente, incluyendo el historial de servicios prestados y la facturación. Esto facilita la administración de las relaciones con los clientes y asegura que se les facture de manera precisa y oportuna.

6. Seguridad y Control de Acceso

- **Control de Acceso Basado en Roles (RBAC):** El portal web implementa un sistema que define los permisos de acceso según los roles de los usuarios, asegurando que solo el personal autorizado pueda acceder a ciertas áreas del sistema.

- **Autenticación y Encriptación:** Se utilizan tecnologías como OAuth 2.0 para la autenticación y TLS para la encriptación de datos, asegurando que las comunicaciones y los datos almacenados estén protegidos contra accesos no autorizados.

Cronograma de Implementación portal Web

Cronograma de Implementación del Portal Web

Fase / Funcionalidad	Duración Estimada	Mes 1	Mes 2	Mes 3	Mes 4
----------------------	-------------------	-------	-------	-------	-------

1. Gestión de Órdenes de Servicio	5 semanas	X	X		
- Recepción y Clasificación de Órdenes	2 semanas	X			
- Asignación de Técnicos	1 semana		X		
- Seguimiento en Tiempo Real	2 semanas		X		
2. Monitoreo y Seguimiento de Actividades	4 semanas		X	X	
- Dashboard de Monitoreo	2 semanas		X		
- Mapa de Rutas y Gestión de Incidencias	2 semanas			X	
3. Comunicación y Notificaciones	3 semanas			X	
- Envío de Notificaciones Push	2 semanas			X	
- Mensajería Interna	1 semana				X
4. Reportes y Análisis de Datos	3 semanas				X
- Generación de Reportes	2 semanas				X
- Análisis de Datos	1 semana				X
5. Facturación y Gestión de Clientes	2 semanas				X
- Generación de Facturas	1 semana				X
- Gestión de Clientes	1 semana				X
6. Seguridad y Control de Acceso	3 semanas	X			X
- Control de Acceso Basado en Roles (RBAC)	2 semanas	X			
- Autenticación y Encriptación	1 semana				X
7. Pruebas e Integración Final	4 semanas				X

Mes 1: Fase Inicial y Gestión de Órdenes de Servicio

- **Gestión de Órdenes de Servicio (5 semanas):** Durante el primer mes y parte del segundo, se desarrollan las funcionalidades básicas para la recepción, clasificación y asignación de órdenes de servicio. Esto incluye el desarrollo del módulo de seguimiento en tiempo real, asegurando que cualquier cambio en el portal se refleje inmediatamente en la app móvil.

Mes 2: Monitoreo y Seguimiento

- **Monitoreo y Seguimiento de Actividades (4 semanas):** Se implementan los dashboards y mapas de rutas que permiten a los administradores monitorear el progreso de las órdenes en tiempo real. Esto está directamente relacionado con la app móvil, de la cual se reciben actualizaciones constantes.

Mes 3: Comunicación y Reportes

- **Comunicación y Notificaciones (3 semanas):** Durante este mes, se integran las notificaciones push y el sistema de mensajería interna. Esto permite mantener una comunicación fluida entre los administradores y los técnicos, facilitando la coordinación de tareas.
- **Reportes y Análisis de Datos (3 semanas):** Se implementan las herramientas de generación de reportes y análisis de datos, permitiendo a los administradores evaluar el rendimiento del equipo y tomar decisiones basadas en datos.

Mes 4: Facturación, Seguridad, y Pruebas Finales

- **Facturación y Gestión de Clientes (2 semanas):** Se desarrollan las funcionalidades para automatizar la facturación y gestionar la información de los clientes. Esto se relaciona directamente con la finalización de las órdenes en la app móvil, de donde se extrae la información para la facturación.
- **Seguridad y Control de Acceso (3 semanas):** Se implementan las medidas de seguridad necesarias para proteger la información sensible, incluyendo el control de acceso basado en roles y la encriptación de datos.
- **Pruebas e Integración Final (4 semanas):** El último mes se dedica a pruebas exhaustivas e integración final, asegurando que todas las funcionalidades del portal web estén completamente sincronizadas con la app móvil y funcionen sin problemas antes del despliegue en producción.

Requerimientos No Funcionales Portal Web

1. Rendimiento

- **Tiempo de Respuesta:** El portal web debe responder a las solicitudes de los usuarios en menos de 2 segundos en el 95% de los casos. Esto incluye la carga de páginas, la ejecución de consultas en la base de datos, y la actualización de los dashboards en tiempo real.
- **Capacidad de Carga:** El sistema debe ser capaz de manejar múltiples usuarios concurrentes sin degradar significativamente su rendimiento. Debería soportar al menos 1000 usuarios concurrentes sin que el tiempo de respuesta supere los 3 segundos.
- **Optimización de Consultas:** Las consultas a la base de datos deben estar optimizadas para minimizar el tiempo de procesamiento, especialmente para reportes y análisis de datos.

2. Escalabilidad

- **Escalabilidad Horizontal y Vertical:** El portal web debe estar diseñado para ser escalable tanto horizontalmente (añadiendo más servidores) como verticalmente (mejorando la capacidad del hardware existente) para manejar un incremento en la cantidad de usuarios y datos sin comprometer el rendimiento.
- **Soporte para Crecimiento de Datos:** La arquitectura del sistema debe permitir el crecimiento del volumen de datos almacenados sin degradar el rendimiento, utilizando bases de datos distribuidas o particionadas si es necesario.

3. Seguridad

- **Control de Acceso Basado en Roles (RBAC):** El portal debe implementar un sistema de control de acceso que asegure que solo los usuarios autorizados puedan acceder a ciertas funcionalidades o datos. Los roles deben ser configurables y asignados según las necesidades específicas de la empresa.

- **Encriptación de Datos:** Todos los datos sensibles, tanto en tránsito como en reposo, deben estar encriptados. Esto incluye la encriptación de comunicaciones (usando TLS) y la encriptación de datos almacenados en la base de datos.
- **Autenticación y Autorización:** Uso de OAuth 2.0 para la autenticación segura de usuarios, junto con la autenticación de dos factores (2FA) para acceso a áreas críticas del portal.
- **Auditoría y Registro de Eventos:** El portal debe mantener un registro detallado de todas las acciones realizadas por los usuarios, incluyendo accesos, cambios de datos, y administración de roles, para asegurar un seguimiento y auditoría efectiva.

4. Usabilidad

- **Interfaz de Usuario Intuitiva:** El portal debe ofrecer una interfaz de usuario clara y fácil de usar, con una curva de aprendizaje baja para los administradores. Debe seguir las mejores prácticas de diseño UX/UI.
- **Accesibilidad:** El portal debe cumplir con las normas de accesibilidad (como las WCAG 2.1) para asegurar que pueda ser utilizado por personas con diferentes capacidades.
- **Compatibilidad Multinavegador:** El portal web debe ser completamente funcional en los principales navegadores web (Chrome, Firefox, Safari, Edge) y estar optimizado para diferentes resoluciones de pantalla.

5. Disponibilidad y Fiabilidad

- **Alta Disponibilidad:** El portal debe estar disponible al menos el 99.9% del tiempo, excluyendo las ventanas de mantenimiento programadas. Esto requiere una arquitectura redundante y medidas de recuperación ante desastres.
- **Recuperación ante Fallos:** En caso de fallo del sistema, debe haber mecanismos para la recuperación rápida, como backups automáticos y replicación de bases de datos, para minimizar la pérdida de datos y el tiempo de inactividad.
- **Monitoreo Continuo:** Implementación de herramientas de monitoreo continuo para detectar y responder rápidamente a problemas de rendimiento, seguridad, o disponibilidad.

6. Mantenibilidad

- **Documentación Completa:** Todo el código y las configuraciones del sistema deben estar bien documentados para facilitar la comprensión y el mantenimiento por parte de los equipos técnicos.
- **Modularidad:** El portal debe estar diseñado de manera modular para que las diferentes partes del sistema puedan ser mantenidas, actualizadas, o reemplazadas de manera independiente sin afectar a otras áreas.



- **Facilidad de Actualización:** Debe ser fácil implementar actualizaciones y parches sin interrumpir significativamente el servicio. Esto incluye la capacidad de realizar actualizaciones incrementales.

7. Portabilidad

- **Compatibilidad con Diferentes Plataformas:** El portal debe ser compatible con diferentes sistemas operativos y entornos de hosting, permitiendo su despliegue en servidores locales o en la nube, dependiendo de las necesidades de la empresa.
- **Despliegue en la Nube:** Debe ser posible desplegar el portal en entornos de nube como AWS, Azure, o Google Cloud, con opciones para escalabilidad automática y administración de recursos.

8. Integración

- **Integración con la Aplicación Móvil:** El portal debe estar completamente integrado con la app móvil, asegurando que toda la información relevante (órdenes de servicio, datos de ubicación, comunicaciones) se sincronice de manera efectiva entre ambos sistemas.
- **APIs Restful:** El portal debe ofrecer APIs bien documentadas y seguras para la integración con otros sistemas, facilitando la interoperabilidad y la extensión de funcionalidades.

Comunicación y Servicios Web para la Operación de la App Móvil y Portal Web de Gescomer

Objetivo del Anexo

Este anexo técnico complementa el proyecto de desarrollo de la aplicación móvil y el portal web de Gescomer, proporcionando una guía estructurada sobre la gestión de la comunicación entre estos componentes y las bases de datos internas de la empresa. El objetivo es asegurar una integración fluida y eficiente mediante el uso de servicios web (APIs) que facilitarán la interacción entre la app móvil, el portal web y los sistemas internos de Gescomer.

Estructura de los Servicios Web

Los servicios web (APIs) descritos en este anexo son fundamentales para garantizar que la aplicación móvil y el portal web funcionen de manera coherente, permitiendo una comunicación bidireccional con las bases de datos y otros sistemas internos de Gescomer. Los siguientes puntos destacan las funcionalidades clave que cada API ofrecerá, junto con ejemplos de las operaciones que se podrán realizar.

Estimación de Endpoints Requeridos

Se ha estimado un conjunto de endpoints que serán implementados para cubrir las necesidades de comunicación entre los diferentes componentes del sistema. Estos endpoints abarcan desde la autenticación y gestión de usuarios hasta la captura de multimedia y la geolocalización en tiempo real. Esta estimación se tomará como punto de partida, pero es importante tener en cuenta que estará sujeta a cambios durante el desarrollo, y los tiempos de implementación podrán variar según las necesidades y ajustes que surjan durante el proyecto.

1. API para Autenticación y Autorización

- Gestionará el acceso seguro y la autorización de usuarios en la aplicación móvil y el portal web.

2. API para la Gestión de Órdenes de Servicio

- Facilitará la creación, actualización y seguimiento de las órdenes de servicio, sincronizando la información entre la app y el portal.

3. API para Consultas Maestras

- Permitirá el acceso a datos maestros esenciales como tipos de órdenes, estados, y listas de materiales, asegurando que ambas plataformas estén siempre actualizadas.

4. API para Notificaciones Push

- Gestionará el envío y la configuración de notificaciones push, asegurando que los usuarios estén informados de cualquier actualización relevante.

5. API para Gestión de Usuarios

- Administrará los usuarios, roles y permisos dentro del sistema, asegurando un control de acceso adecuado.

6. API para Gestión de Inventarios y Materiales

- Controlará el inventario de materiales utilizados en las órdenes de servicio, permitiendo consultas y actualizaciones en tiempo real.

7. API para Geolocalización y Mapas

- Proveerá servicios de geolocalización y rutas optimizadas para los técnicos en campo, además de registrar ubicaciones y rutas.

8. API para Facturación

- Gestionará la facturación de las órdenes de servicio, permitiendo la generación y actualización de facturas basadas en servicios completados.

9. API para Captura y Almacenamiento de Multimedia

- Facilitará la captura, almacenamiento y consulta de archivos multimedia asociados a las órdenes de servicio.

10. API para Encuestas y Feedback

- Gestionará la creación, actualización y recopilación de respuestas de encuestas, facilitando el feedback de técnicos y clientes.

11. API para Auditoría y Seguridad

- Asegurará la trazabilidad y seguridad del sistema mediante el registro de actividades y cambios en el sistema.

Gestión y Documentación de APIs

Para garantizar una correcta gestión de los servicios web y su integración en los proyectos, se utilizarán las siguientes herramientas:

1. Servicios de Documentación de APIs:

- **Swagger:** Será utilizado para la documentación completa de las APIs, proporcionando una interfaz interactiva que facilita la comprensión y prueba de los endpoints por parte

de los desarrolladores. Swagger permitirá generar automáticamente la documentación en función del código, lo que asegura que esté siempre actualizada y precisa.

- **OpenAPI:** En conjunto con Swagger, se utilizará la especificación OpenAPI para describir las capacidades y estructuras de las APIs, facilitando la interoperabilidad entre diferentes sistemas y servicios.

2. Gestión de Pruebas con Postman:

- **Postman:** Se empleará Postman para la gestión y prueba de los endpoints de las APIs. Postman permitirá crear colecciones de pruebas automatizadas, verificar el comportamiento de las APIs bajo diferentes condiciones, y asegurar que cada endpoint funcione correctamente antes de su integración en los sistemas productivos. Las colecciones de Postman también serán compartidas entre los equipos de desarrollo y QA para facilitar las pruebas y garantizar la calidad del software.

VALOR DE LA SOLUCION:

Desarrollo de la Aplicación Móvil y Portal Web

1. Recursos Humanos

1. Director de Proyecto

- Responsabilidades: Supervisión general del proyecto, coordinación entre equipos, gestión de recursos, y aseguramiento de la entrega en tiempo y forma.
- Participación: En todas las fases del proyecto, proporcionando dirección y tomando decisiones clave.

2. Desarrollador de Aplicación Móvil (2 desarrolladores)

- Responsabilidades: Desarrollo de la aplicación móvil en Flutter, integración con el backend PHP, pruebas unitarias, y resolución de problemas.
- Participación: Principalmente en el desarrollo de la aplicación móvil y la integración con el backend.

3. Desarrollador de Portal Web (2 desarrolladores)

- Responsabilidades: Desarrollo del portal web en Angular y PHP, integración con la base de datos y el backend, desarrollo de APIs, y pruebas unitarias.
- Participación: Desarrollo del portal web, integración con el backend, y desarrollo de APIs.

4. Ingeniero de Calidad (QA)

- Responsabilidades: Realización de pruebas funcionales, de rendimiento, y de seguridad en ambos proyectos. Garantizar la calidad del software antes del despliegue.
- Participación: En las fases de pruebas y corrección de errores para ambos proyectos.

5. Ingeniero de Infraestructura

- Responsabilidades: Configuración y mantenimiento de la infraestructura (servidores, redes, seguridad) en entornos de desarrollo y producción, asegurando alta disponibilidad, escalabilidad, y seguridad.
- Participación: En la configuración inicial de la infraestructura, durante el monitoreo continuo, y en la fase de despliegue para garantizar una operación sin problemas.

6. Desarrollador Back APIs -> Gescomer

- Responsabilidades: Desarrollo e integración de las APIs para la comunicación entre la app móvil, el portal web, y las bases de datos, incluyendo pruebas y optimización.
- Participación: Durante el desarrollo de las APIs, colaborando con los equipos de frontend y backend, y en la fase de pruebas y despliegue para asegurar el correcto funcionamiento de los servicios.

Distribución de Roles y costos Desarrollo App Movil y Portal Web

7. Fase / Actividad	Rol	Horas (Aplicación Móvil)	Horas (Portal Web)
Fase 1: Autenticación y Acceso / Gestión de Órdenes de Servicio			
- Desarrollo	Desarrolladores (App)	240 horas	
- Desarrollo	Desarrolladores (Web)		400 horas
- Pruebas	Ingeniero QA	20 horas	20 horas
- Supervisión	Director de Proyecto	60 horas	100 horas
Fase 2: Sincronización de Datos y Funcionalidad Offline / Monitoreo y Seguimiento de Actividades			
- Desarrollo	Desarrolladores (App)	160 horas	
- Desarrollo	Desarrolladores (Web)		320 horas
- Pruebas	Ingeniero QA	20 horas	40 horas
- Supervisión	Director de Proyecto	40 horas	80 horas
Fase 3: Gestión de Órdenes de Servicio / Comunicación y Notificaciones			
- Desarrollo	Desarrolladores (App)	80 horas	
- Desarrollo	Desarrolladores (Web)		240 horas
- Pruebas	Ingeniero QA	20 horas	20 horas
- Supervisión	Director de Proyecto	20 horas	60 horas

Fase 4: Funcionalidades de Mapas y GPS / Reportes y Análisis de Datos			
- Desarrollo	Desarrolladores (App)	320 horas	
- Desarrollo	Desarrolladores (Web)		240 horas
- Pruebas	Ingeniero QA	40 horas	20 horas
- Supervisión	Director de Proyecto	80 horas	60 horas
Fase 5: Comunicación y Notificaciones / Facturación y Gestión de Clientes			
- Desarrollo	Desarrolladores (App)	240 horas	
- Desarrollo	Desarrolladores (Web)		160 horas
- Pruebas	Ingeniero QA	20 horas	20 horas
- Supervisión	Director de Proyecto	60 horas	40 horas
Fase 6: Captura de Imágenes y Videos / Seguridad y Control de Acceso			
- Desarrollo	Desarrolladores (App)	160 horas	
- Desarrollo	Desarrolladores (Web)		240 horas
- Pruebas	Ingeniero QA	20 horas	20 horas
- Supervisión	Director de Proyecto	40 horas	60 horas
Fase 7: Encuestas y Formularios Dinámicos / Pruebas e Integración Final			
- Desarrollo	Desarrolladores (App)	160 horas	
- Desarrollo	Desarrolladores (Web)		320 horas
- Pruebas	Ingeniero QA	40 horas	60 horas
- Supervisión	Director de Proyecto	40 horas	80 horas
Fase 8: Pruebas e Integración Final (Aplicación Móvil)			
- Pruebas	Ingeniero QA	80 horas	
- Supervisión	Director de Proyecto	80 horas	

Distribución del Trabajo API comunicación con los servicios internos de Gescomer

API/Servicio	Descripción	Horas Estimadas (Desarrollo + Pruebas + Despliegue)
1. API para Autenticación y Autorización	Maneja la autenticación segura de usuarios y la gestión de tokens.	80 horas
2. API para la Gestión de Órdenes de Servicio	Permite la creación, actualización y consulta de órdenes de servicio.	120 horas
3. API para Consultas Maestras	Proporciona acceso a datos maestros como tipos de órdenes y materiales.	60 horas
4. API para Notificaciones Push	Gestiona el envío y configuración de notificaciones push para los usuarios.	60 horas
5. API para Gestión de Usuarios	Administra los usuarios, roles y permisos del sistema.	80 horas
6. API para Gestión de Inventarios y Materiales	Controla el inventario y el uso de materiales en las órdenes de servicio.	100 horas
7. API para Geolocalización y Mapas	Proporciona geolocalización y rutas optimizadas para técnicos en campo.	100 horas
8. API para Facturación	Maneja la facturación de órdenes de servicio, incluyendo la generación de facturas.	80 horas
9. API para Captura y Almacenamiento de Multimedia	Facilita la captura, almacenamiento y consulta de archivos multimedia asociados a las órdenes.	80 horas

10. API para Encuestas y Feedback	Gestiona la creación, envío y recolección de encuestas y feedback.	60 horas
11. API para Auditoría y Seguridad	Proporciona servicios de auditoría para la trazabilidad y seguridad del sistema.	60 horas
Total		940 horas

Horas Totales por Rol:

Rol	Horas en Aplicación Móvil	Horas en Portal Web	Total de Horas	Costo
Director de Proyecto	380 horas	480 horas	860 horas	
Desarrolladores (App)	1,680 horas	-	1,680 horas	
Desarrolladores (Web)	-	1,920 horas	1,920 horas	
Ingeniero QA	260 horas	240 horas	500 horas	
Ingeniero de Infraestructura	120 horas	120 Horas	240 horas	
Desarrollador Back APIs			940 horas	
Costo Total				

Infraestructura Técnica

La infraestructura técnica para el proyecto de Gescomer estará basada en un entorno robusto y escalable utilizando tecnologías de Linux, Nginx, balanceadores de carga, y bases de datos SQL Server, con un enfoque en servicios de Azure para desarrollo y producción.

1. Servidores y Entorno

1.1. Servidores de Desarrollo

- **Propósito:** Entorno para el desarrollo, pruebas, y staging antes del despliegue en producción.
- **Configuración:**

- Sistema Operativo: Linux (Ubuntu Server 20.04 LTS)
- Servidor Web: Nginx para manejar el tráfico HTTP/HTTPS.
- Servidor de Aplicaciones: PHP 8.x para backend.
- Base de Datos: SQL Server en Azure (Azure SQL Database).
- Herramientas CI/CD: Azure DevOps para la integración y despliegue continuos.
- Control de Versiones: Git, integrado con Azure Repos.
- Entorno de Contenerización: Docker para aislar servicios y aplicaciones.
- Virtualización: Azure Virtual Machines para instancias de desarrollo, con copias exactas del entorno de producción para asegurar la consistencia.

1.2. Servidores de Producción

- **Propósito: Entorno para la operación en vivo de la aplicación móvil y el portal web.**
- **Configuración:**
 - Sistema Operativo: Linux (Ubuntu Server 20.04 LTS)
 - Servidor Web: Nginx con soporte para SSL/TLS, configurado como proxy inverso y manejando múltiples sitios.
 - Servidor de Aplicaciones: PHP 8.x, optimizado para rendimiento en producción.
 - Base de Datos: SQL Server en Azure (Azure SQL Database Managed Instance) con replicación geográfica y alta disponibilidad.
 - Balanceador de Carga: Azure Load Balancer para distribuir el tráfico entre múltiples instancias de la aplicación y asegurar la alta disponibilidad.
 - Caché: Azure Cache for Redis para mejorar la velocidad de las consultas y reducir la carga en SQL Server.
 - Almacenamiento de Archivos: Azure Blob Storage para almacenar archivos multimedia, documentos, y respaldos.
 - Monitoreo y Alertas: Azure Monitor y Azure Log Analytics para el monitoreo en tiempo real y alertas sobre el rendimiento y la seguridad.
 - Respaldo y Recuperación: Azure Backup para respaldos automatizados de la base de datos y máquinas virtuales, con retención de 30 días.

2. Servicios en Azure

2.1. Azure SQL Database

- **Descripción: Servicio de base de datos relacional como servicio (DBaaS) basado en SQL Server.**
- **Uso:**
 - Desarrollo: Azure SQL Database para entornos de desarrollo y pruebas, con escalabilidad ajustable según la carga de trabajo.
 - Producción: Azure SQL Database Managed Instance para producción, asegurando alta disponibilidad, seguridad, y rendimiento.

2.2. Azure Load Balancer

- Descripción: Servicio de balanceo de carga que distribuye el tráfico de red entrante a múltiples instancias de aplicaciones.
- Uso: Balancear el tráfico entre múltiples servidores de aplicaciones en producción, asegurando que las solicitudes sean manejadas de manera eficiente y que la infraestructura sea tolerante a fallos.

2.3. Azure Virtual Machines

- **Descripción: Instancias de máquinas virtuales escalables para ejecutar aplicaciones y servicios.**
- **Uso:**
 - Desarrollo: Instancias dedicadas para entornos de desarrollo y pruebas, replicando el entorno de producción.
 - Producción: Máquinas virtuales configuradas para alta disponibilidad, con réplicas en distintas zonas geográficas para garantizar la continuidad del servicio.

2.4. Azure Blob Storage

- Descripción: Servicio de almacenamiento en la nube para datos no estructurados, como documentos, imágenes y videos.
- Uso: Almacenamiento de archivos multimedia y documentos generados por la aplicación móvil y el portal web, con acceso rápido y seguro desde cualquier parte del mundo.

2.5. Azure DevOps

- Descripción: Conjunto de herramientas para la gestión del ciclo de vida de aplicaciones, que incluye repositorios Git, CI/CD, y gestión de proyectos.
- Uso: Implementación de pipelines de CI/CD para automatizar la integración y despliegue de código en los entornos de desarrollo y producción. Gestión de código fuente y seguimiento de tareas mediante Azure Repos y Azure Boards.

2.6. Azure Monitor y Log Analytics

- Descripción: Servicios para el monitoreo en tiempo real y análisis de logs en Azure.
- Uso: Monitorización continua de las aplicaciones y la infraestructura, con alertas automáticas en caso de fallos o rendimiento degradado. Análisis detallado de logs para diagnosticar problemas y optimizar el rendimiento.

2.7. Azure Backup

- Descripción: Solución de respaldo en la nube que protege las máquinas virtuales, bases de datos, y archivos.

- Uso: Respaldo automatizado de las bases de datos, máquinas virtuales, y datos críticos, con planes de recuperación ante desastres para minimizar el tiempo de inactividad.

3. Seguridad y Red

3.1. Nginx como Proxy Inverso

- Descripción: Nginx se utilizará como un proxy inverso para dirigir el tráfico entrante a los servidores de backend adecuados y manejar las conexiones SSL/TLS.
- Uso: Proteger las aplicaciones mediante la terminación SSL/TLS, mejora del rendimiento con caching y compresión, y distribución de la carga de trabajo.

3.2. Azure Firewall

- Descripción: Firewall de red que proporciona una protección centralizada para las aplicaciones.
- Uso: Implementación de reglas de firewall para proteger el tráfico entrante y saliente, permitiendo solo las conexiones necesarias y asegurando que los datos estén protegidos contra accesos no autorizados.

3.3. VPN en Azure

- Descripción: Red privada virtual para asegurar las conexiones entre la infraestructura en Azure y los usuarios autorizados.
- Uso: Acceso seguro a los servidores de desarrollo y producción para administradores y desarrolladores, protegiendo las comunicaciones sensibles mediante cifrado.

4. Plan de Implementación y Escalabilidad

4.1. Despliegue Inicial

- Desarrollo: Configuración de los servidores de desarrollo y la infraestructura de CI/CD en Azure. Creación de entornos replicados que simulen la producción para asegurar que las pruebas sean precisas y representativas.
- Producción: Despliegue inicial en Azure con configuraciones para alta disponibilidad, incluyendo la configuración del balanceador de carga, servidores web, y base de datos.

4.2. Escalabilidad

- Horizontal: Aumentar la capacidad de la infraestructura agregando más instancias de máquinas virtuales o bases de datos conforme aumente la carga.
- Vertical: Mejorar los recursos de las máquinas existentes (más CPU, RAM) cuando sea necesario, asegurando que los tiempos de respuesta se mantengan óptimos.

Estimación de Costos

Componente	Descripción	Configuración	Costo Estimado
Servidores de Desarrollo	Entorno para desarrollo, pruebas y staging.	Linux (Ubuntu Server), Nginx, PHP, SQL Server en Azure, Docker, Azure DevOps, Azure Virtual Machines.	
Servidores de Producción	Entorno para operación en vivo de la app y el portal.	Linux (Ubuntu Server), Nginx, PHP, Azure SQL Database Managed Instance, Azure Load Balancer, Azure Cache for Redis, Azure Blob Storage.	
Azure SQL Database	Base de datos relacional en la nube.	Azure SQL Database para desarrollo y Azure SQL Database Managed Instance para producción, con replicación geográfica.	
Azure Load Balancer	Balaneo de carga para alta disponibilidad.	Distribución de tráfico entre múltiples instancias de aplicación en producción.	
Azure Virtual Machines	Máquinas virtuales escalables.	VM para desarrollo, staging, y producción; configuración para alta disponibilidad y replicación en zonas geográficas.	
Azure Blob Storage	Almacenamiento de archivos y multimedia en la nube.	Almacenamiento de documentos, imágenes, videos, y respaldos, con acceso seguro y rápido.	
Azure DevOps	Herramientas para CI/CD y gestión del ciclo de vida.	Pipelines de CI/CD, repositorios Git, y seguimiento de tareas, integración con Azure Repos y Azure Boards.	
Azure Monitor y Log Analytics	Monitoreo y análisis de logs en tiempo real.	Monitoreo continuo, alertas automáticas, dashboards personalizados, análisis detallado de logs.	
Azure Backup	Solución de respaldo en la nube.	Respaldo automatizado de bases de datos, máquinas virtuales y datos críticos, con planes de recuperación ante desastres.	



Nginx (Proxy Inverso)	Servidor web para manejo de tráfico HTTP/HTTPS.	Terminación SSL/TLS, proxy inverso para servidores backend, caching, y compresión de tráfico.	
Azure Firewall	Protección centralizada de la red.	Reglas de firewall para proteger el tráfico entrante y saliente, integración con Azure Security Center.	
VPN en Azure	Red privada virtual para conexiones seguras.	Acceso seguro para administradores y desarrolladores a los servidores de desarrollo y producción, cifrado de comunicaciones sensibles.	
Azure Cache for Redis	Almacenamiento en caché para mejorar rendimiento.	Cacheo de consultas SQL para reducir la carga de la base de datos y mejorar la velocidad de las aplicaciones.	
Azure Security Center	Supervisión de seguridad.	Análisis continuo de amenazas, recomendaciones de seguridad, gestión de la postura de seguridad, y monitoreo de conformidad.	
Azure Active Directory	Gestión de identidades y control de acceso.	Autenticación de usuarios, gestión de identidades, implementación de políticas de seguridad y acceso basado en roles (RBAC).	
Total, Costo Infraestructura			