Emisión de Orden de Trabajo (OT)

Proveedor	**EXCELSIS S.A.**	**Nombre del Producto**	**Entregable 3**
Nro. de Orden	**003**	**Horas de Trabajo**	**427**
Fecha de Emisión	**12/07/2024**	**Plazo máximo de entrega**	**01/10/2024**

- ## Descripción del Producto

Análisis y desarrollo de la **Aplicación Móvil de Avisos, Notificaciones e Indicadores de la CSJ - Entregable 3.**

- ## Objetivos

1. Desarrollar la aplicación Móvil de Avisos, Notificaciones e Indicadores de la CSJ (APP), la cual será publicada por la Corte Suprema de Justicia (CSJ) para estar disponible desde cualquier tienda en línea para dispositivos móviles, asumiendo la responsabilidad del costo de las tiendas en donde será publicada.
2. La aplicación deberá ser de fácil utilización, amigable y permitir al usuario configurar las notificaciones a ser recibidas en el teléfono, generadas desde un evento que requiera su participación en algún proceso laboral o alguna notificación que por la naturaleza de su función requiera ser informado.
3. Esta aplicación deberá estar preparada para interactuar con las diferentes Bases de Datos del SIGAF, así como también con las demás bases de datos que sean administradas por la Dirección General de Tecnologías de la Información y las Comunicaciones (DGTIC) de la CSJ y que puedan adaptarse a este proceso.

- ## Requerimientos Funcionales (RF)

1. **Control de Acceso:** Corresponde a la primera pantalla que debe mostrarse al usuario en caso de no estar logueado dentro de la aplicación móvil (APP).

   1.1 **Acceso al Sistema.** Permite al usuario ingresar su nombre de usuario y contraseña. Al presionar el botón ingresar, el sistema debe verificar si los datos proporcionados pertenecen a un usuario registrado en el sistema. La APP emitirá un mensaje en caso de falla o error durante el proceso de logueo. Se deberá consumir una API proveída por CSJ que retornará el Usuario y el Rol con el cual deberá acceder el usuario. Si el usuario cierra sesión una vez dentro de la aplicación, deberá ser direccionado a la pantalla de login.

1.2 **Primer Logueo**. Si la APP detecta que es el primer ingreso desde la instalación, se realizará una inserción de un nuevo registro a la tabla usuario, con los datos del usuario y el Firebase cloud Messaging key (FCM-key). El FCM-key se genera de manera automática por primera y única vez cuando el usuario realiza el inicio de sesión. El FCM-key no puede estar duplicado en la tabla usuario. Es un valor único por dispositivo. El API de Notificación se encargará de borrar el registro con FCM-key duplicado e insertará el nuevo registro con los datos proporcionados desde el inicio de sesión.

Ejemplo del Maquetado

2. **Administración de categoría de notificaciones.** El sistema deberá desplegar un listado de categorías con datos provenientes de la tabla *categoria*. Las categorías serán creadas de manera manual en la tabla *categoria*. El comportamiento de estas categorías está sujeto al desarrollo previo en la capa de integración y el API Notificador.

2.1 **Bandeja de Categorías.** Las categorías deberán ser previamente filtradas según el Rol del usuario logueado antes de mostrarse en pantalla, es decir, un usuario solo podrá visualizar las categorías si posee el Rol asociado en su Perfil. El usuario podrá seleccionar una categoría para acceder a las notificaciones asociadas. Cada categoría deberá mostrar en pantalla los siguientes datos:

- Nombre de la categoría
- Cantidad de notificaciones sin leer
- Cantidad de procesos pendientes del usuario

2.2 **Avisos Generales.** Deberá existir una categoría especial llamada Avisos Generales. Esta categoría permitirá acceder a las notificaciones que son comunes a todos los usuarios de la aplicación.

Ejemplo del Maquetado

3. **Bandeja de notificaciones:** Presenta el listado de las Notificaciones generadas para el usuario conectado, discriminadas por Categorías.

3.1 **Notificaciones por Categorías.** La pantalla debe mostrar el listado de notificaciones asociadas al usuario conectado perteneciente a una categoría. La pantalla deberá

estar dotada de acciones preestablecidas que permitan filtrar las notificaciones. Los filtros definidos deberán estar agrupados por rango de tiempo (hoy, 3 días, 7 días y personalizado) y por estado de la notificación (no leído, procesos pendientes). También, como parte del filtro, deberá contener un campo texto, donde permitirá al usuario escribir y filtrar coincidencias. Cada notificación que aparece en pantalla deberá mostrar los siguientes datos:

- Título de la notificación
- Descripción de la notificación
- Fecha de creación de la notificación
- Una marca que indique que la notificación tiene asociada un proceso pendiente por parte del usuario.
- Una marca que indique que la notificación aún no fue leída.

3.2 **Funcionalidades en bandeja de notificaciones.** El usuario deberá seleccionar una notificación para acceder al detalle de la misma y podrá desplazarse por la pantalla para acceder a las notificaciones que no se muestran en primer plano. La pantalla debe permitir las cargas parciales mediante paginado (infinite scroll, paginado de a 10) que evita traer todos los datos de una vez. Incorporar manejo de cache para mantener en memoria la lista de notificaciones previamente cargadas y una funcionalidad de actualización de listado completo mediante un gesto del usuario en la pantalla (onPullToRefresh)

Ejemplo del maquetado

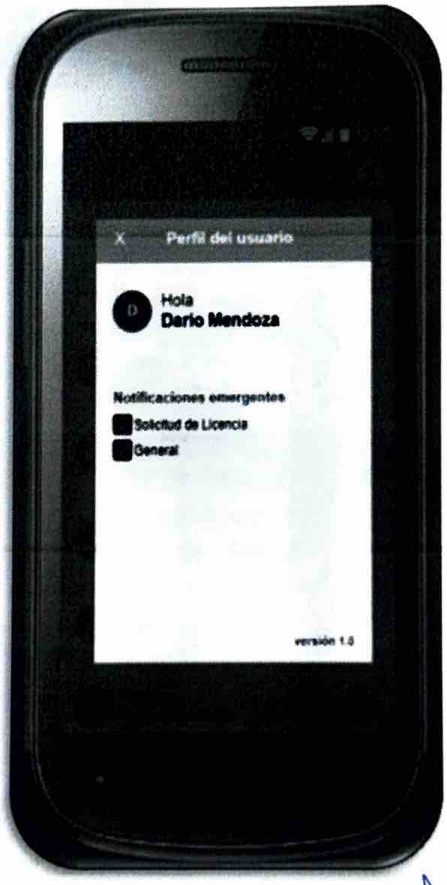
4. **Visualizar detalle de notificación.** Corresponde a la pantalla que se visualiza una vez que se selecciona una notificación de la pantalla de listado de Notificaciones.
   4.1 **Acceso al detalle.** La aplicación deberá mostrar todos los datos de la notificación que sean relevantes para el usuario. La aplicación deberá recuperar los datos de la tabla notificaciones y actualizar la columna *leido_fecha* con la fecha y hora actual del sistema. Los datos que deberán aparecer en pantalla son:
   - Categoría
   - Fecha de creación de la notificación
   - Título de la notificación
   - Descripción de la notificación
   - Detalles adicionales de la notificación (opcional)
   - Una marca que muestre que la notificación tiene asociado un proceso pendiente del usuario

Ejemplo del maquetado

5. **Administrar perfil de usuario.** El usuario deberá poder realizar configuraciones específicas para su perfil.
   5.1 **Datos visualizados.** Los datos que deberán aparecer en pantalla son los siguientes:
   - Datos del usuario conectado
   - Preferencias del usuario acerca de las notificaciones emergentes

   5.2 **Configuración de preferencias.** El usuario deberá poder seleccionar, por categoría, las notificaciones emergentes que desea recibir en su dispositivo móvil. Estas preferencias se almacenan en la tabla *usuario_preferencia*

Ejemplo del maquetado



6. **Creación y envío de notificaciones.** Se contemplan dos (2) tipos de notificaciones:
   - **Emergentes:** Las notificaciones emergentes son aquellas que aparecen directamente en la pantalla principal del usuario sin que éste se encuentre dentro de la aplicación. Al seleccionar la notificación emergente, la aplicación deberá direccionar al detalle de la notificación recibida, sólo si el usuario no ha cerrado sesión previamente
   - **Estáticas**. Las notificaciones estáticas son aquellas que son solicitadas bajo demanda al API de Notificación.

   6.1 **Funcionalidad de las Notificaciones Emergentes.**

   6.1.1 Las notificaciones emergentes son aquellas que aparecen directamente en la pantalla principal del usuario sin que éste se encuentre dentro de la aplicación.

   6.1.2 Las notificaciones emergentes deberán ser generadas y enviadas de manera automática. Está acción la realiza el API de Notificación consultando previamente la tabla *notificacion* y enviando un request por API al sistema Firebase cloud Messaging (FCM).

   6.1.3 El FCM es el encargado de enviar las notificaciones emergentes al dispositivo móvil del usuario.

   6.1.4 Las notificaciones que se encuentran bajo la categoría de *Avisos Generales*, requerirán una inserción manual en la tabla *notificacion* pero las mismas también tendrán el comportamiento de una notificación emergente. Una vez que el usuario cierra sesión, el usuario ya no recibirá notificaciones emergentes. La aplicación deberá persistir en la tabla de notificaciones del usuario e inactivar todas las preferencias; una vez que el usuario inicia sesión la API deberá volver a activar todas las preferencias.

   6.2 **Funcionalidad de las notificaciones Estáticas.**

   6.2.1 Las notificaciones estáticas deberán ser enviadas a la APP bajo demanda, es decir, cuando el usuario ingresa a la pantalla de listado de Notificaciones se actualizan las mismas.

7. **Recuperación de datos de los Sistemas Fuente.**

   7.1 **Captura de datos.** Los cambios producidos en la Base de Datos serán capturados por Kafka Connect a través de Debezium, el cual posee configuraciones para conectarse a distintas Bases de Datos y de esa manera, escuchar permanentemente cualquier cambio que ocurra en las tablas asociadas.

   7.2 **Configuración Debezium.** El mismo deberá ser configurado para que Debezium pueda reaccionar a las inserciones, modificaciones y eliminaciones de las tablas asociadas. Cuando recibe un estímulo por parte de la Base de Datos anfitrión, se encarga de transmitir los datos a los tópicos de Kafka.

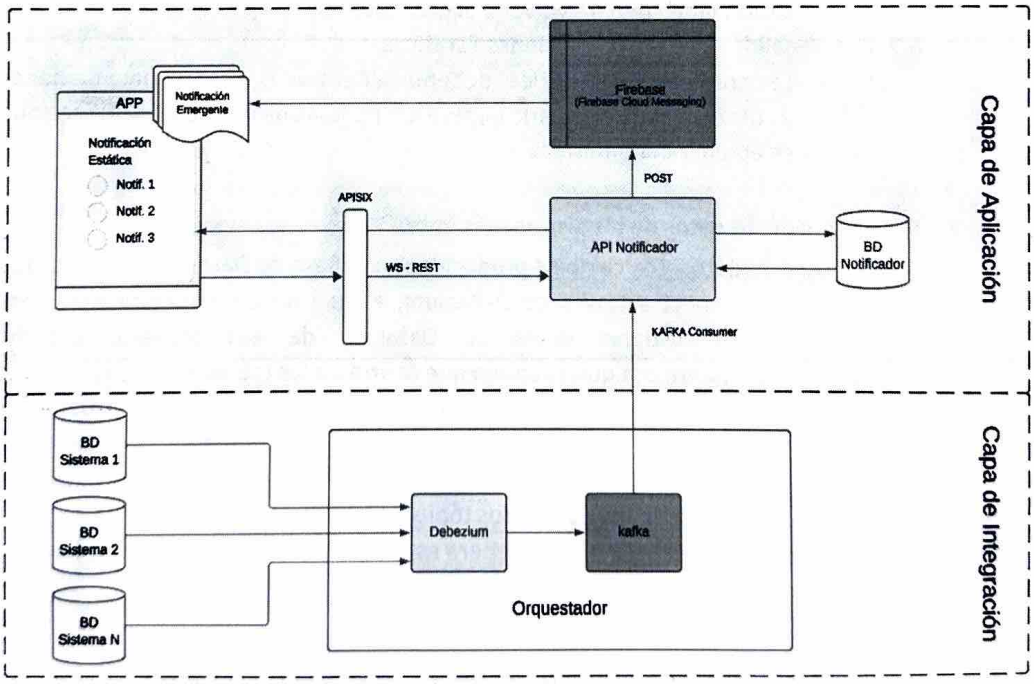
   7.3 **Configuración Kafka.** La misma deberá estar diseñada para administrar los flujos de datos provenientes de Debezium y enviarlos al API de Notificación.

7.4 **Configuración Kafka Consumer:** El API de Notificaciones debe programarse para escuchar los datos provenientes del sistema Kafka y esto se logra incorporando la librería del Kafta Consumer. Esta librería requiere de programación y configuraciones adicionales como lo son las suscripciones a los tópicos de Kafka.

7.5 **Transformación de Datos.** Antes de realizar la inserción de datos a la base de datos de notificación, el API de Notificación deberá recibir el set de datos por parte del Kafka mediante la suscripción a los tópicos de Kafta, el mismo es capaz de reaccionar a cualquier señal, este realizará una transformación de datos según la fuente origen. El API de notificación podrá reconocer la fuente de datos y accionar para transformar los datos antes de la persistencia. realiza una transformación de datos según la fuente origen. El API de notificación podrá reconocer la fuente de datos y accionar para transformar los datos antes de la persistencia.

7.6 **Activación y Configuración de CDC.** Activar y configurar la captura de datos modificados (CDC) en la Base de Datos indicada y las tablas afectadas.

Arquitectura del Sistema



- **Requerimientos No Funcionales (RNF)**

1. **Manejo de contraseñas.** La contraseña para los usuarios será proveída y gestionada por la DGTIC de la CSJ.
2. **Manejo de Campos Obligatorios.** Los campos obligatorios deben estar correctamente identificados y emitir mensajes de validación apropiados.
3. **Modificaciones, mejoras o sugerencias.** Cualquier librería adicional, sentencias, procedimientos almacenados, mejoras y/o metodologías tecnológicas a ser utilizadas deberán ser autorizadas por el equipo técnico de la DGTIC de la CSJ.
4. **Diseño de la Base de Datos.** La estructura y el diseño de la Base de Datos correspondiente deberá ser consensuado y autorizado por el equipo técnico de la DGTIC de la CSJ a fin de precautelar el mantenimiento y la escalabilidad del producto.
5. **Repositorio de Datos.** Durante el desarrollo de los proyectos se deberá utilizar los repositorios oficiales indicados por el equipo técnico de la DGTIC de la CSJ, modificando y entregando los productos en dichos repositorios, así como las documentaciones correspondientes. El avance referente a los requerimientos detallados en la presente OT, deberán ser actualizados periódicamente, teniendo en cuenta la frecuencia definida de común acuerdo entre el equipo técnico de la DGTIC de la CSJ y el Proveedor.
6. **Evidencias de Pruebas y Control de Calidad.** Se deberán realizar todas las pruebas y el control de calidad presentando las documentaciones correspondientes en atención a lo indicado en el PBC.
7. **Documentación.** Las activaciones (CDC) y configuraciones de las nuevas tecnologías a implementar deberán ser debidamente documentadas y de ser necesario, acompañadas por el equipo técnico de la DGTIC de la CSJ.
8. **Consideraciones generales.**
   - En la presente OT ya fueron considerados el tiempo destinado a las reuniones de definición, análisis a procedimientos, relevamiento de datos y elaboración de minutas.
   - Las personas involucradas en la OT deberán cumplir estrictamente todo lo concerniente a la cláusula 3.3 referente a la confidencialidad detallada en el PBC.

Lic. Diego A. López Gallas
Director
Dirección General de TIC
Administrador del Contrato
CSJ

Joel García
Gerente de Proyecto
EXCELSIS S.A.

Recibido: Proveedor
En Fecha: 15/07/2024

Ing. Claudia Vargas de Figueredo
Jefe de Departamento
Dirección General de TIC
Administrador Auxiliar
CSJ