



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Universidad Nacional de Colombia - sede Bogotá
Facultad de Ingeniería
Departamento de Ingeniería de Sistemas e Industrial
Curso: Ingeniería de Software I

Taller 1

Estudiantes: [Cristofer Damian Ordonez Osa](#)
[Angel David Beltran Garcia](#)
[Deivid Farid Ardila Herrera](#)
[Yony Sebastian Chaparro Mesa](#)

Grupo: JavaLí 🍷

Fecha: 17/05/2025

Índice

1. Lista inicial de requerimientos	1
2. División en Requerimientos Funcionales y No Funcionales	2
3. Priorización por MoSCoW	5
4. Estimación tiempo/esfuerzo con Fibonacci	9

1. Lista inicial de requerimientos

- req 1: evitar la duplicación de solicitudes al momento de coordinar la asignación de pedidos.
- req 2: integrar comunicación entre Software Principal y Whatsapp.
- req 3: notificar retrasos en entregas de pedidos de forma automática a las operadoras logísticas y a los clientes afectados.
- req 4: generar informe estadístico de indicadores operativos a las operadoras logísticas.
- req 5: desarrollar plataformas compatibles con dispositivos móviles y de escritorio.
- req 6: permitir al cliente crear pedidos a través de la aplicación.
- req 7: mostrar al cliente el estado actualizado de los pedidos.
- req 8: permitir al cliente crear un pedido a través del ChatBot de WhatsApp.
- req 9: permitir al cliente actualizar información a través del ChatBot de WhatsApp.
- req 10: permitir al cliente cancelar pedidos a través del ChatBot de WhatsApp.
- req 11: mostrar listado de repartidores disponibles a las operadoras logísticas.
- req 12: visualizar ubicación en tiempo real de los repartidores a las operadoras logísticas.
- req 13: presentar estado de pedidos asignados a las operadoras logísticas.
- req 14: permitir filtrar a las operadoras logísticas pedidos por su estado actual.
- req 15: registrar pedidos desde la oficina a través de un formulario simple.

- req 16: importar pedidos desde la oficina a través de hojas de cálculo.
 req 17: crear rutas para los repartidores desde la oficina.
 req 18: editar rutas para los repartidores desde la oficina.
 req 19: eliminar rutas para los repartidores desde la oficina.
 req 20: permitir al repartidor el acceso a datos básicos del pedido.
 req 21: permitir al repartidor visualizar la lista de pedidos asignados para el día.
 req 22: permitir al repartidor actualizar el estado de los pedidos.
 req 23: permitir al repartidor filtrar pedidos asignados por su estado.
 req 24: permitir al repartidor visualizar mapa del recorrido para la recepción y entrega de los pedidos.
 req 25: permitir la creación de nuevos usuarios y la asignación de un específico por usuario.
 req 26: generar una interfaz de inicio de sesión para que los usuarios puedan acceder al sistema.

2. División en Requerimientos Funcionales y No Funcionales

<i>Requerimientos Funcionales (RF)</i>	<i>Requerimientos No Funcionales (RNF)</i>
RF_1 _(antiguo req. 1) : el <i>sistema</i> notificará retrasos en entregas (por condiciones climáticas, demora por parte del establecimiento, etc.) de forma automática a los usuarios con el rol de <i>cliente</i> que sean afectados y a los usuarios con el rol de <i>operador logístico</i> , por medio de un aviso en la barra de notificaciones tanto en la aplicación móvil como en la de escritorio.	RNF_1 _(antiguo req. 1) : el <i>sistema</i> deberá prevenir la duplicación de solicitudes al momento de que usuarios con el rol <i>operador logístico</i> coordinen la asignación de pedidos.
RF_2 _(antiguo req. 3) : el <i>sistema</i> deberá generar informes estadísticos periódicos de los indicadores operativos (tiempos de entrega, zonas más problemáticas, repartidores más rápidos) a los usuarios con el rol de <i>operador logístico</i> a través de la interfaz de escritorio.	RNF_2 _(antiguo req. 2) : el <i>sistema</i> deberá integrarse con Whatsapp (por medio de una API) para permitir la administración de pedidos hechos, a través de ChatBot, por los usuarios con el rol de <i>cliente</i> .
RF_3 _(antiguo req. 11) : el <i>sistema</i> deberá mostrar el listado de los usuarios con el rol de <i>repartidor</i> disponibles (es decir, repartidores que hayan completado la ruta que les fue asignada) a los usuarios con el rol <i>operador logístico</i> a través de la aplicación de escritorio.	RNF_3 _(antiguo req. 5) : los usuarios con rol de <i>operador logístico</i> podrán acceder a los servicios ofrecidos en plataformas compatibles con ordenadores.

RF_4 _(antiguo req. 22) : el usuario con rol de <i>repartidor</i> deberá poder actualizar el estado de los pedidos ¹ asignados por día a través de la aplicación móvil.	RNF_4 _(antiguo req. 5) : los usuarios con rol de <i>cliente</i> podrán acceder a los servicios ofrecidos en plataformas compatibles con dispositivos móviles.
RF_5 _(antiguo req. 13) : el usuario con el rol de <i>operador logístico</i> podrá tener acceso a la información ² de todos los pedidos generados, a través de la aplicación de escritorio.	RNF_5 _(antiguo req. 5) : los usuarios con rol de <i>repartidor</i> podrán acceder a los servicios ofrecidos en plataformas compatibles con dispositivos móviles.
RF_6 _(antiguo req. 14) : el usuario con el rol de <i>operador logístico</i> podrá filtrar pedidos con base en su estado actual, a través de un botón ubicado en la aplicación de escritorio.	
RF_7 _(antiguo req. 15) : el usuario con el rol de <i>operador logístico</i> podrá administrar el registro de pedidos a través de un formulario ³ disponible en la interfaz de escritorio.	
RF_8 _(antiguo req. 16) : el usuario con rol de <i>operador logístico</i> deberá poder importar pedidos almacenados en hojas de cálculo ⁴ a través de la interfaz de escritorio.	
RF_9 _(antiguo req. 17) : el usuario con rol de <i>operador logístico</i> deberá poder administrar la creación de rutas para los usuarios con el rol de <i>repartidor</i> a través de la interfaz de escritorio. Para crear una ruta se deberán especificar todas las direcciones por las cuales tiene que pasar el repartidor.	

¹ se comprenden como *estados del pedido* los siguientes: por recibir, recibido, en entrega y entregado.

² como mínimo, se pedirán los siguientes datos para el pedido: estado del pedido, monto que paga el cliente por el servicio prestado, información del remitente e información del destinatario. Para los dos últimos, se solicitará la dirección, número de teléfono y nombre.

³ el formulario solicitará la información del pedido, declarada en el RF_5. Todos los campos serán obligatorios.

⁴ los campos de las hojas de cálculo deberán coincidir con la información solicitada en el formulario del RF_7.

RF_10 _(antiguo req. 18) : el usuario con rol de <i>operador logístico</i> deberá poder administrar la edición de rutas para los usuarios con el rol de <i>repartidor</i> a través de la interfaz de escritorio	
RF_11 _(antiguo req. 19) : el usuario con rol de <i>operador logístico</i> deberá poder administrar la eliminación de rutas para los usuarios con el rol de <i>repartidor</i> a través de la interfaz de escritorio	
RF_12 _(antiguo req. 20) : el usuario con rol de <i>repartidor</i> podrá acceder a los datos de los pedidos asignados en forma de lista, a través de la aplicación para dispositivos móviles.	
RF_13 _(antiguo req. 21) : el usuario con rol de <i>repartidor</i> deberá poder visualizar la lista de pedidos asignados por día, a través de la aplicación para dispositivos móviles.	
RF_14 _(antiguo req. 12) : el <i>sistema</i> deberá permitir la visualización de la ubicación de todos los usuarios con el rol de <i>repartidor</i> (en tiempo real) a los usuarios con el rol de <i>operador logístico</i> , a través de un mapa en la interfaz de escritorio.	
RF_15 _(antiguo req. 24) : el usuario con rol de <i>repartidor</i> deberá poder visualizar, a través de la aplicación móvil, un mapa con la ruta para la recepción y entrega de pedidos.	
RF_16 _(antiguo req. 23) : el usuario con rol de <i>repartidor</i> deberá poder filtrar la lista de pedidos asignados por su estado a través de la aplicación móvil.	
RF_17 _(antiguo req. 6) : el usuario con rol de <i>cliente</i> deberá poder crear pedidos a través de la aplicación móvil, especificando la información del pedido (tal y como en el formulario del RF_7).	

RF_18 _(antiguo req. 7) : el usuario con rol de <i>cliente</i> deberá poder visualizar, a través de la aplicación móvil, el estado actualizado de los pedidos que ha realizado.	
RF_19 _(antiguo req. 8) : el usuario con el rol de <i>cliente</i> podrá crear un pedido a través del ChatBot de WhatsApp, especificando la información del pedido (tal y como en el formulario del RF_7).	
RF_20 _(antiguo req. 9) : el usuario con el rol de <i>cliente</i> podrá actualizar la información de los pedidos asociados a él, a través del ChatBot de WhatsApp.	
RF_21 _(antiguo req. 10) : el usuario con el rol de <i>cliente</i> podrá cancelar pedidos asociados a él, a través del ChatBot de WhatsApp.	
RF_22 _(antiguo req. 25) : cualquier usuario podrá solicitar la creación de una cuenta con un rol específico (<i>operador logístico, repartidor, cliente</i>). La cuenta será creada cuando el personal responsable verifique la información y la autorice.	
RF_23 _(antiguo req. 26) : el usuario con el rol de <i>operador logístico</i> podrá acceder a la aplicación de escritorio a través de una interfaz de inicio de sesión.	
RF_24 _(antiguo req. 26) : el usuario con el rol de <i>repartidor</i> podrá acceder a la aplicación móvil a través de una interfaz de inicio de sesión.	
RF_25 _(antiguo req. 26) : el usuario con el rol de <i>cliente</i> podrá acceder a la aplicación móvil a través de una interfaz de inicio de sesión.	

3. Priorización por MoSCoW

MUST

- RF_3: el *sistema* debe mostrar el listado de los usuarios con el rol de *repartidor* que se encuentren disponibles (es decir, repartidores que hayan completado la ruta que les fue asignada) a los usuarios con el rol *operador logístico* a través de la aplicación de escritorio. El listado de repartidores debe contener, como mínimo, los siguientes datos: nombre del repartidor, número de contacto y ubicación actual.
- RF_4: el usuario con rol de *repartidor* debe poder actualizar el estado de los pedidos asignados por día, a través de la aplicación móvil. Esta actualización sólo podrá ser realizada para los pedidos que hayan sido entregados o recibidos, así mismo, el repartidor debe disponer de una lista para seleccionar el pedido a actualizar y una opción para establecerlo como *entregado* o *recibido*.
- RF_5: el usuario con el rol de *operador logístico* debe tener acceso a la información de todos los pedidos generados, a través de la aplicación de escritorio. Los datos de todos los pedidos se presentarán por medio de una tabla.
- RF_7: el usuario con el rol de *operador logístico* debe poder administrar el registro de pedidos a través de un formulario disponible en la interfaz de escritorio.
- RF_8: el usuario con rol de *operador logístico* debe poder importar pedidos almacenados en hojas de cálculo, a través de la interfaz de escritorio.
- RF_9: el usuario con rol de *operador logístico* debe poder administrar la creación de rutas⁵ a través de la interfaz de escritorio. Para crear una ruta, se deben especificar todas las direcciones por las cuales tiene que pasar el repartidor.
- RF_10: el usuario con rol de *operador logístico* debe poder administrar la edición de rutas a través de la interfaz de escritorio. Al momento de editar una ruta en uso⁶ solo se debe permitir modificar las direcciones por las cuales no haya pasado el repartidor.
- RF_11: el usuario con rol de *operador logístico* debe poder administrar la eliminación de rutas a través de la interfaz de escritorio. Solo se debe permitir la eliminación de rutas que no estén en uso.
- RF_12: el usuario con rol de *repartidor* debe acceder a todos los datos de los pedidos que se le han asignado en el día actual, a través de la aplicación para dispositivos móviles.
- RF_13: el usuario con rol de *repartidor* debe poder visualizar una lista que contenga los pedidos que se le hayan asignado por día, a través de la aplicación para dispositivos móviles.
- RF_14: el *sistema* debe permitir la visualización de la ubicación de todos los usuarios con el rol de *repartidor* (en tiempo real) a los usuarios con el rol de *operador logístico*, a través de un mapa en la interfaz de escritorio. Para la implementación del mapa deben utilizarse una o varias APIs externas.
- RF_19: el usuario con el rol de *cliente* debe poder crear un pedido a través del ChatBot de WhatsApp⁷, especificando la información del mismo (tal y como en el formulario del RF_7).
- RF_20: el usuario con el rol de *cliente* debe poder actualizar la información de los pedidos asociados a él, a través del ChatBot de WhatsApp. Solo se debe permitir modificar pedidos que no estén en curso.

⁵ solo se pueden asignar rutas a repartidores.

⁶ se considera que una ruta está en uso cuando está asignada a un repartidor que no ha terminado su recorrido.

⁷ el ChatBot será quien guíe al cliente al momento de crear, editar o eliminar pedidos, ofreciéndole diferentes opciones con el fin de que este solo deba manifestar cuál de ellas elige.

- RF_21: el usuario con el rol de *cliente* debe poder cancelar pedidos asociados a él, a través del ChatBot de WhatsApp. Solo se debe permitir anular pedidos que no estén en curso.
- RF_22: cualquier usuario podrá solicitar la creación de una cuenta con un rol específico (*operador logístico*, *repartidor*, *cliente*). La cuenta será creada cuando el personal responsable verifique la información y la autorice. Al momento de la solicitud, el usuario deberá especificar, como mínimo, la siguiente información: nombre completo, nombre de usuario (único por cada usuario), número de contacto, rol y clave de acceso.
- RF_23: el usuario con el rol de *operador logístico* debe acceder a la aplicación de escritorio a través de una interfaz de inicio de sesión, con sus credenciales correspondientes (nombre de usuario y contraseña), las cuales son verificadas en la base de datos.
- RF_24: el usuario con el rol de *repartidor* debe acceder a la aplicación móvil a través de una interfaz de inicio de sesión, con sus credenciales correspondientes (nombre de usuario y contraseña), las cuales son verificadas en la base de datos.
- RF_25: el usuario con el rol de *cliente* debe acceder a la aplicación móvil a través de una interfaz de inicio de sesión, con sus credenciales correspondientes (nombre de usuario y contraseña), las cuales son verificadas en la base de datos.
- RNF_1: el *sistema* debe prevenir la duplicación de solicitudes al momento de que los usuarios con el rol *operador logístico* coordinen la asignación de pedidos.
- RNF_2: el *sistema* debe integrarse con Whatsapp (por medio de una API) para permitir la administración de pedidos hechos, a través de ChatBot, por los usuarios con el rol de *cliente*.
- RNF_3: los usuarios con rol de *operador logístico* deben poder acceder a los servicios ofrecidos en plataformas compatibles con ordenadores cuyo sistema operativo sea *Windows*, *GNU/Linux* o *MacOS*.
- RNF_4: los usuarios con rol de *cliente* deben poder acceder a los servicios ofrecidos por medio de una aplicación móvil nativa disponible para *Android* e *iOS*.
- RNF_5: los usuarios con rol de *repartidor* deben poder acceder a los servicios ofrecidos por medio de una aplicación móvil nativa disponible para *Android* e *iOS*.

SHOULD

- RF_1: el *sistema* debería notificar retrasos en entregas⁸ (por condiciones climáticas, demora por parte del establecimiento, etc.) de forma automática a los usuarios con el rol de *cliente* que sean afectados y a los usuarios con el rol de *operador logístico*, por medio de una notificación *push* tanto en la aplicación móvil como en la de escritorio.
- RF_6: el usuario con el rol de *operador logístico* debe poder filtrar pedidos con base en su estado actual (por medio de una lista desplegable) a través de la aplicación de escritorio.
- RF_16: el usuario con rol de *repartidor* debe poder filtrar la lista de pedidos asignados a él con base en su estado, a través de la aplicación móvil. Para realizar

⁸ se considera que hay un retraso en la entrega cuando se excede el tiempo establecido por el *sistema*.

el filtrado el repartidor debe disponer de una lista desplegable donde pueda seleccionar el estado.

COULD

- RF_2: el *sistema* podría generar informes estadísticos periódicos⁹ (mediante gráficos y tablas) de los indicadores operativos (tiempos de entrega, zonas más problemáticas, repartidores más rápidos) para los usuarios con el rol de *operador logístico*, a través de la interfaz de escritorio.
- RF_15: el usuario con rol de *repartidor* podría visualizar, a través de la aplicación móvil, un mapa (utilizando una API externa) con la ruta para la recepción y entrega de pedidos. Además, el mapa se deberá actualizar cada 15 segundos.

WON'T

- RF_17: el usuario con rol de *cliente* podría poder crear pedidos a través de la aplicación móvil, especificando la información del pedido (tal y como en el formulario del RF_7).
- RF_18: el usuario con rol de *cliente* podría visualizar, a través de la aplicación móvil, el estado de los pedidos que ha realizado. Dicho estado se actualizará al momento de que se genere un cambio en el estado del pedido en el *sistema*.

⁹ la periodicidad de los informes será diaria, semanal y mensual. El usuario podrá especificarla mediante una lista desplegable.

4. Estimación tiempo/esfuerzo con Fibonacci

Requisito		Estimación	Argumento
RF_1	El <i>sistema</i> debería notificar retrasos en entregas (por condiciones climáticas, demora por parte del establecimiento, etc.) de forma automática a los usuarios con el rol de <i>cliente</i> que sean afectados y a los usuarios con el rol de <i>operador logístico</i> , por medio de una notificación push tanto en la aplicación móvil como en la de escritorio.	13	<p>No sabemos cómo obtener los datos del estado del pedido en tiempo real, probablemente se requieran <i>listeners</i>, con los cuales hemos trabajado muy poco.</p> <p>Por otro lado, aunque tenemos cierta intuición de las notificaciones <i>push</i>, nunca las hemos implementado. Probablemente para esto se deban utilizar dos APIs diferentes, pues las notificaciones deben aparecer tanto en la aplicación móvil como en la aplicación de escritorio.</p>
SHOULD			
RF_2	El <i>sistema</i> podría generar informes estadísticos periódicos (mediante gráficos y tablas) de los indicadores operativos (tiempos de entrega, zonas más problemáticas, repartidores más rápidos) para los usuarios con el rol de <i>operador logístico</i> , a través de la interfaz de escritorio.	8	<p>Debemos realizar una <i>query</i> eficiente y escalable. Aunque tenemos conocimientos en consultas, realmente no tenemos mucha experiencia al respecto, así que el planteamiento puede tardar un tiempo considerable.</p> <p>Para el posterior análisis de los datos se necesitan saber los indicadores que le serán útiles al cliente, por lo cual se deberá estar en constante contacto con este último.</p> <p>Finalmente, debemos investigar e implementar librerías que trabajen con componentes gráficos y</p>
COULD			

			tablas dinámicas.
RF_3	el <i>sistema</i> debe mostrar el listado de usuarios con el rol de <i>repartidor</i> que se encuentren disponibles (es decir, repartidores que hayan completado la ruta que les fue asignada) a los usuarios con el rol <i>operador logístico</i> a través de la aplicación de escritorio. El listado de repartidores debe contener, como mínimo, los siguientes datos: nombre del repartidor, número de contacto y ubicación actual.	8	No sabemos cómo obtener los datos de la disponibilidad de los repartidores en tiempo real, probablemente se necesiten desarrollar APIs para poder obtener tal información en la aplicación de escritorio. Adicionalmente, para esta implementación se necesitan los datos de geolocalización de todos los repartidores; investigamos que se pueden utilizar <i>WebSockets</i> , sin embargo, hay varios conceptos asociados que desconocemos.
MUST			
RF_4	El usuario con rol de <i>repartidor</i> debe poder actualizar el estado de los pedidos que se le han asignado por día, a través de la aplicación móvil. Esta actualización sólo podrá ser realizada para los pedidos que hayan sido entregados o recibidos, así mismo, el repartidor debe disponer de una lista para seleccionar el pedido a actualizar y una opción para establecerlo como <i>entregado</i> o <i>recibido</i> .	8	No tenemos conocimiento sobre cómo notificar (desde la aplicación móvil) cambios en los estados de los pedidos; probablemente se necesiten desarrollar APIs para poder enviar tal información a la aplicación de escritorio. Adicionalmente, se debe plantear, con base en la lógica del negocio, cómo validar que, efectivamente, un pedido ha sido recibido o entregado.
MUST			
RF_5	El usuario con el rol de <i>operador logístico</i> debe tener acceso a la información de todos los pedidos generados, a través de la aplicación de escritorio. Los datos de todos los pedidos se presentarán por medio de una tabla.	8	Para tener acceso a la información de todos los pedidos generados es imprescindible la actualización en tiempo real. Se considera que será necesaria la implementación de una API; no obstante, carecemos de conocimientos y, por ende, de experiencia en esta área.

MUST			Asimismo, debemos procurar que el sistema cumpla con el criterio de escalabilidad cuando existan numerosos pedidos.
RF_6	El usuario con el rol de <i>operador logístico</i> debe poder filtrar pedidos con base en su estado actual (por medio de una lista desplegable) a través de la aplicación de escritorio.	3	Se trata de una consulta básica a la base de datos que considera un parámetro de estado indicado por el operador logístico, en este punto consideramos que no debería haber mayor complejidad. Únicamente debemos procurar que la base de datos esté correctamente definida e indexada.
SHOULD			
RF_7	El usuario con el rol de <i>operador logístico</i> debe poder registrar los pedidos a través de un formulario disponible en la interfaz de escritorio.	2	Es una tarea sencilla de rutina y verificación. Únicamente debemos desarrollar un formulario amigable para los operadores logísticos, además de procurar que los campos estén debidamente diligenciados.
MUST			
RF_8	El usuario con rol de <i>operador logístico</i> debe poder importar pedidos almacenados en hojas de cálculo, a través de la interfaz de escritorio.	2	Para importar pedidos almacenados en hojas de cálculo es necesario el uso de diferentes librerías, algunas de las cuales ya resultan familiares para nuestro equipo. Por otro lado, debemos procurar que sea posible importar pedidos desde los diferentes formatos de hojas de cálculo que existen, tales como x/s, csv, entre otros.
MUST			

RF_9	El usuario con rol de <i>operador logístico</i> debe poder administrar la creación de rutas a través de la interfaz de escritorio. Para crear una ruta se deben especificar todas las direcciones por las cuales tiene que pasar el usuario con el rol de <i>repartidor</i> .	2	Es una tarea sencilla de rutina y verificación. Únicamente debemos desarrollar una serie de campos en los que los operadores logísticos puedan ingresar las direcciones, además de procurar que estas últimas sean reales.
MUST			
RF_10	El usuario con rol de <i>operador logístico</i> debe poder administrar la edición de rutas a través de la interfaz de escritorio. Al momento de editar una ruta en uso solo se debe permitir modificar las direcciones por las cuales no haya pasado el usuario con el rol de <i>repartidor</i> .	8	En un principio debemos crear una comunicación entre la interfaz de escritorio y la aplicación móvil para enviar y recibir información de las rutas asignadas, garantizando que las modificaciones solo se realicen en las partes de la ruta que no hayan sido completadas.
MUST			
RF_11	El usuario con rol de <i>operador logístico</i> debe poder administrar la eliminación de rutas a través de la interfaz de escritorio. Solo se debe permitir la eliminación de rutas que no estén en uso.	8	Únicamente debemos desarrollar validaciones alineadas con la lógica de negocio al momento de eliminar rutas, para evitar la afectación de pedidos en curso.
MUST			
RF_12	El usuario con rol de <i>repartidor</i> debe acceder a todos los datos de los pedidos que se le han asignado en el día actual, a través de la aplicación para dispositivos móviles.	8	Debemos enviar y recibir los datos de los pedidos asignados a los repartidores en el día actual, teniendo en cuenta que durante el día se maneja un flujo de pedidos constante.
MUST			
RF_13	El usuario con rol de <i>repartidor</i> debe poder visualizar una lista que contenga los pedidos que se le han asignado por día, a través de la aplicación para dispositivos móviles.	8	Aunque la mayoría de los pedidos van a ser asignados al inicio de la jornada, va a haber pedidos que tengan que ser asignados durante la jornada. Por lo que la comunicación con la base de datos en tiempo real es indispensable.
MUST			

RF_14	El <i>sistema</i> debe permitir la visualización de la ubicación de todos los usuarios con el rol de <i>repartidor</i> (en tiempo real) a los usuarios con el rol de <i>operador logístico</i> , a través de un mapa en la interfaz de escritorio. Para la implementación del mapa deben utilizarse una o varias APIs externas.	8	<p>En un principio debemos crear una comunicación entre la interfaz de escritorio y la aplicación móvil para enviar y recibir cada una de las ubicaciones de los repartidores.</p> <p>De igual manera, para integrar la funcionalidad de mapas en la interfaz de escritorio, necesitaremos configurar mínimo una API externa para poder visualizar un mapa con la ubicación de cada uno de los repartidores.</p> <p>Además, este intercambio de información se debe realizar cada cierto tiempo, lo que implica utilizar los datos de geolocalización de cada repartidor de manera constante.</p> <p>Esta actualización constante, muy seguramente consumirá una cantidad de recursos considerables, por lo que debemos pensar en el rendimiento y optimización de esta funcionalidad.</p>
MUST			
RF_15	El usuario con rol de <i>repartidor</i> podría visualizar, a través de la aplicación móvil, un mapa (utilizando una API externa) con la ruta para la recepción y entrega de pedidos. Además, el mapa se deberá actualizar cada 15 segundos.	21	<p>En un principio debemos crear una comunicación entre la interfaz de escritorio y la aplicación móvil para enviar y recibir información de las rutas asignadas.</p> <p>Por otro lado, para integrar la funcionalidad de mapas en la aplicación, se necesitará configurar una API externa para visualizar la ruta asignada al repartidor, sobre lo cual, no tenemos conocimiento alguno.</p>
COULD			

			Además, la aplicación deberá actualizar el mapa cada cierto tiempo, lo que implica utilizar los datos de geolocalización constantemente, así que debemos pensar en el rendimiento y el ahorro de batería al utilizar la funcionalidad.
RF_16	El usuario con rol de <i>repartidor</i> debe poder filtrar la lista de pedidos asignados con base en su estado, a través de la aplicación móvil. Para realizar el filtrado el repartidor debe disponer de una lista desplegable donde pueda seleccionar el estado.	3	Es una tarea sencilla que implica una consulta básica sobre los datos ya almacenados en la base de datos. Únicamente debemos crear una interfaz amigable con el repartidor para que pueda utilizar la funcionalidad de filtrado.
SHOULD			
RF_17	El usuario con rol de <i>cliente</i> podría poder crear pedidos a través de la aplicación móvil, especificando la información del pedido (tal y como en el formulario del RF_7).	13	En un inicio debemos crear una comunicación entre la aplicación móvil y la base de datos. De manera que se pueda respetar la integridad de los datos, validar correctamente la creación y evitar la duplicación del pedido.
WON'T			
RF_18	El usuario con rol de <i>cliente</i> podría visualizar, a través de la aplicación móvil, el estado de los pedidos que ha realizado. Dicho estado se actualizará al momento de que se genere un cambio en el estado del pedido en el <i>sistema</i> .	13	No sabemos cómo obtener los datos del estado del pedido en tiempo real, probablemente se requieran <i>listeners</i> , con los cuales hemos trabajado muy poco. Implica la correcta validación del estado de cada uno de los pedidos asociados al <i>cliente</i> , así como una comunicación directa con la base de datos en tiempo real.
WON'T			

RF_19	El usuario con el rol de <i>cliente</i> debe poder crear un pedido a través del ChatBot de WhatsApp ¹⁰ , especificando la información del mismo (tal y como en el formulario del RF_7).	13	<p>No tenemos conocimiento acerca de la integración entre el ChatBot y la base de datos.</p> <p>Implementar esta funcionalidad presenta cierto grado de complejidad, ya que implica varios desafíos técnicos como son respetar la integridad de los datos, validar correctamente la creación y evitar la duplicación del pedido.</p>
MUST			
RF_20	El usuario con el rol de <i>cliente</i> debe poder actualizar la información de los pedidos asociados a él, a través del ChatBot de WhatsApp. Solo se debe permitir modificar pedidos que no estén en curso.	13	<p>No tenemos conocimiento acerca de la integración entre el ChatBot y la base de datos.</p> <p>Implementar esta funcionalidad tiene cierto grado de complejidad, ya que implica desafíos técnicos como lo son validar correctamente el estado del pedido (para que no se puedan modificar pedidos en curso), evitar realizar modificaciones indebidas, manejar la comunicación en tiempo real con el ChatBot y garantizar que solo el <i>cliente</i> autorizado pueda realizar modificaciones.</p>
MUST			
RF_21	El usuario con el rol de <i>cliente</i> debe poder cancelar pedidos asociados a él, a través del ChatBot de WhatsApp. Solo se debe permitir anular pedidos que no estén en curso.	13	<p>No tenemos conocimiento acerca de la integración entre el ChatBot y la base de datos.</p> <p>Implementar esta funcionalidad tiene cierto grado de complejidad, ya que implica desafíos técnicos como lo son validar correctamente el estado del pedido para evitar cancelaciones indebidas, manejar la comunicación en tiempo real con el ChatBot y</p>
MUST			

¹⁰ el ChatBot será quien guíe al cliente al momento de crear, editar o eliminar pedidos, ofreciéndole diferentes opciones con el fin de que este solo deba manifestar cuál de ellas elige.

			garantizar la seguridad para que solo el <i>cliente</i> autorizado pueda realizar cancelaciones.
RF_22	Cualquier usuario podrá solicitar la creación de una cuenta con un rol específico (<i>operador logístico, repartidor, cliente</i>). La cuenta será creada cuando el personal responsable verifique la información y la autorice. Al momento de la solicitud, el usuario deberá especificar, como mínimo, la siguiente información: nombre completo, nombre de usuario, número de contacto, rol y clave de acceso.	13	<p>Debemos garantizar la integridad de los datos de cada usuario. También debemos verificar que las contraseñas cumplan ciertas reglas mínimas de seguridad como los estándares <i>NIST</i> u otros estándares para la seguridad y la privacidad de todos los usuarios.</p> <p>De acuerdo a lo anterior, concluimos que existe la necesidad de profundizar más en temas relacionados a la integridad de los datos y la ciberseguridad, por lo cual, se puede afirmar que tomará tiempo implementar esta funcionalidad.</p>
MUST			
RF_23	El usuario con el rol de <i>operador logístico</i> debe acceder a la aplicación de escritorio a través de una interfaz de inicio de sesión, con sus credenciales correspondientes (nombre de usuario y contraseña), las cuales son verificadas en la base de datos.	21	<p>De acuerdo a lo investigado, debemos desarrollar una API segura para el manejo de las credenciales, en tiempo real, utilizando algoritmos robustos para comparar la contraseña ingresada con el hash en la base de datos.</p> <p>Asimismo, debemos tener planes de acción en los casos en que no se encuentre el usuario, la contraseña sea incorrecta, o la cuenta esté pendiente por aprobación.</p> <p>Teniendo en cuenta lo anterior, concluimos que los conocimientos del equipo con respecto a la seguridad no es suficiente, por lo que tomará tiempo</p>
MUST			
RF_24	El usuario con el rol de <i>repartidor</i> debe acceder a la aplicación móvil a través de una interfaz de inicio de sesión, con sus credenciales correspondientes (nombre de usuario y contraseña), las cuales son verificadas en la base de datos.	21	
MUST			

RF_25	El usuario con el rol de <i>cliente</i> debe acceder a la aplicación móvil a través de una interfaz de inicio de sesión, con sus credenciales correspondientes (nombre de usuario y contraseña), las cuales son verificadas en la base de datos.	21	implementar esta funcionalidad.
MUST			
RNF_1	El <i>sistema</i> debe prevenir la duplicación de solicitudes al momento de que usuarios con el rol <i>operador logístico</i> coordinen la asignación de pedidos.	3	Al hacer la implementación de este sistema de control se requiere validar, en tiempo real, cada nueva solicitud con respecto a las existentes. Esto se puede solucionar desarrollando una lógica clara adecuada.
MUST			
RNF_2	El <i>sistema</i> debe integrarse con Whatsapp (por medio de una API) para permitir la administración de pedidos hechos, a través de ChatBot, por los usuarios con el rol de <i>cliente</i> .	34	<p>Nos parece difícil porque se necesita una integración compleja con APIs, de las cuales no tenemos conocimiento.</p> <p>Integrar un ChatBot no representa desafíos técnicos significativos, ya que requiere más investigación y preparación por parte del equipo de desarrollo.</p> <p>Esto implica comprender y manejar correctamente la API de WhatsApp, para así garantizar la seguridad y privacidad de los datos. Diseñar un flujo conversacional que sea intuitivo y efectivo para los usuarios. Y por último, manejar eventos asincrónicos y posibles errores en la comunicación para mantener la estabilidad del sistema.</p>
MUST			
RNF_3	Los usuarios con rol de <i>operador logístico</i> deben poder acceder a los servicios ofrecidos	55	Es complejo ya que el equipo no cuenta con experiencia previa en desarrollo multiplataforma, lo

MUST	en plataformas compatibles con ordenadores cuyo sistema operativo sea <i>Windows</i> , <i>GNU/Linux</i> o <i>MacOS</i> .		que aumentaría el tiempo de desarrollo y la complejidad del mismo. Desde la investigación de frameworks adecuados hasta la realización de pruebas en distintos entornos (<i>Windows</i> , <i>GNU/Linux</i> o <i>MacOS</i>) y posiblemente enfrentar problemas específicos en cada sistema operativo, esto haría que fuese un desafío técnico considerable.
RNF_4	Los usuarios con rol de <i>cliente</i> deben poder acceder a los servicios ofrecidos por medio de una aplicación móvil nativa disponible para <i>Android</i> e <i>iOS</i> .	55	<p>No tenemos conocimiento acerca de compatibilidad. También implica un diseño robusto que puede tomar tiempo y experiencia previa.</p> <p>En el equipo contamos con experiencia desarrollando apps móviles para <i>Android</i> lo cual facilita esa parte del requerimiento, aún así no contamos con experiencia previa en el desarrollo <i>iOS</i>, lo que representa un desafío adicional que requerirá tiempo para aprender nuevas herramientas y lenguajes específicos de esa plataforma.</p>
MUST			
RNF_5	Los usuarios con rol de <i>repartidor</i> deben poder acceder a los servicios ofrecidos por medio de una aplicación móvil nativa disponible para <i>Android</i> e <i>iOS</i> .	55	<p>No tenemos conocimiento acerca de compatibilidad. También implica un diseño robusto que puede tomar tiempo y experiencia previa.</p> <p>Similar al RNF_4, el equipo cuenta con experiencia en el desarrollo de aplicaciones móviles para <i>Android</i>, lo que facilitará la creación de esa versión. Sin embargo, al no contar con experiencia en desarrollo para <i>iOS</i>, será necesario dedicar tiempo para aprender sus particularidades y herramientas,</p>
MUST			

			además de realizar pruebas específicas para garantizar una experiencia óptima en ambas plataformas.
--	--	--	---