



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**

### **Estudiantes:**

Laura Vanesa Alvarez Lafont

Santiago Andres Fetecua Pulgarin

Nicolas Andres Bolanos Fernandez

Juan Pablo Corredor Castaneda

### **Lista de Requerimientos**

**req 1.** El sistema ha de tener una interfaz responsive para computadora.

**req 2.** El sistema ha de tener una versión móvil que trabaje en iOS y Android.

**req 3.** El usuario debe poder autenticar su identidad y rol mediante un login.

**req 4.** El usuario debe poder acceder a la información pertinente a su rol de forma actualizada y consistente.

**req 5.** El sistema debe asignar de forma automática las rutas a los repartidores.

**req 6.** El sistema debe tener bien establecidas las funciones y permisos para 3 roles diferentes: Operadoras de logística, Repartidores, Clientes.

Que el rol Operadoras de logística pueda:

**req 7.** Ver estadísticas sobre el desempeño de los repartidores.

**req 8.** Recibir los pedidos directamente desde Whatsapp a su interfaz.

**req 9.** Montar un nuevo pedido a partir de un archivo

**req 10.** Asignar entregas

**req 11.** Ver el estado de los repartidores

**req 12.** Ver pedidos y su estado

**req 13.** Crear y cancelar rutas

**req 14.** Hacer cambios sobre rutas definidas

**req 15.** Tener seguimiento en tiempo real, es decir, poder ver en qué parte de la ciudad están los repartidores

Que el rol Repartidor pueda:

**req 16.** Ver su lista de entregas del día

**req 17.** Marcar qué pedidos ya se entregaron

**req 18.** Recibir actualizaciones si hay cambios

**req 19.** Visualizar datos básicos (como las direcciones de origen y destino), de su ruta asignada

**req 20.** Visualizar su ruta a través de un mapa.

Que el rol Cliente pueda:

**req 21.** Visualizar el estado actual de una entrega

**req 22.** Cargar pedidos directamente desde la app

### Requerimientos Funcionales vs No Funcionales

Requerimientos Funcionales	Requerimientos No Funcionales
<b>RF_1.</b> El usuario debe poder autenticar su identidad y rol mediante un login. (antes <b>req3</b> )	<b>RNF_1.</b> El sistema ha de tener una interfaz responsive para computadora. (antes <b>req1</b> )
<b>RF_2.</b> El usuario debe poder acceder a la información pertinente a su rol de forma actualizada y consistente. (antes <b>req4</b> )	<b>RNF_2.</b> El sistema ha de tener una versión móvil que trabaje en iOS y Android. (antes <b>req2</b> )
<b>RF_3.</b> Que el rol Operadoras de logística pueda ver estadísticas sobre el desempeño de los repartidores (antes <b>req 7</b> )	<b>RNF_3.</b> El sistema debe tener bien establecidas las funciones y permisos para 3 roles diferentes: Operadoras de logística, Repartidores, Clientes. (antes <b>req6</b> )
<b>RF_4.</b> Que el rol Operadoras de logística pueda recibir los pedidos directamente desde Whatsapp a su interfaz. (antes <b>req 8</b> )	
<b>RF_5.</b> Que el rol Operadoras de logística pueda montar un nuevo pedido a partir de un archivo. (antes <b>req 9</b> )	
<b>RF_6.</b> Que el rol Operadoras de logística pueda asignar entregas (antes <b>req 10</b> )	
<b>RF_7.</b> Que el rol Operadoras de logística pueda ver el estado de los repartidores.	

(antes <b>req 11</b> )	
<b>RF_8.</b> El sistema debe asignar de forma automática las rutas a los repartidores. (antes <b>req5</b> )	
<b>RF_9.</b> Que el rol Operadoras de logística pueda ver pedidos y su estado. (antes <b>req 12</b> )	
<b>RF_10.</b> Que el rol Operadoras de logística pueda crear y cancelar rutas. (antes <b>req 13</b> )	
<b>RF_11.</b> Que el rol Operadoras de logística pueda hacer cambios sobre rutas definidas. (antes <b>req 14</b> )	
<b>RF_12.</b> Que el rol Operadoras de logística pueda tener seguimiento en tiempo real, es decir, poder ver en qué parte de la ciudad están los repartidores (antes <b>req 15</b> )	
<b>RF_13.</b> Que el rol Repartidor pueda ver su lista de entregas del día. (antes <b>req 16</b> )	
<b>RF_14.</b> Que el rol Repartidor pueda marcar qué pedidos ya se entregaron. (antes <b>req 17</b> )	
<b>RF_15.</b> Que el rol Repartidor pueda recibir actualizaciones si hay cambios. (antes <b>req 18</b> )	
<b>RF_16.</b> Que el rol Repartidor pueda visualizar datos básicos de su ruta asignada. (antes <b>req 19</b> )	
<b>RF_17.</b> Que el rol Repartidor pueda visualizar su ruta a través de un mapa. (antes <b>req 20</b> )	
<b>RF_18.</b> Que el rol Cliente pueda visualizar el estado actual de una entrega. (antes <b>req 21</b> )	
<b>RF_19.</b> Que el rol Cliente pueda cargar pedidos directamente desde la app (antes <b>req 22</b> )	

## Priorización MoSCoW

### MUST

- **RF\_1.** El usuario debe poder autenticar su identidad y rol mediante un login.
- **RF\_2.** El usuario debe poder acceder a la información pertinente a su rol de forma actualizada y consistente.
- **RF\_5.** Que el rol Operadoras de logística pueda montar un nuevo pedido a partir de un archivo.
- **RF\_6.** Que el rol Operadoras de logística pueda asignar entregas
- **RF\_7.** Que el rol Operadoras de logística pueda ver el estado de los repartidores.
- **RF\_9.** Que el rol Operadoras de logística pueda ver pedidos y su estado.
- **RF\_10.** Que el rol Operadoras de logística pueda crear y cancelar rutas.
- **RF\_13.** Que el rol Repartidor pueda ver su lista de entregas del día.
- **RF\_14.** Que el rol Repartidor pueda marcar qué pedidos ya se entregaron.
- **RF\_16.** Que el rol Repartidor pueda visualizar datos básicos de su ruta asignada.
- **RF\_18.** Que el rol Cliente pueda visualizar el estado actual de una entrega.
- **RNF\_1.** El sistema ha de tener una interfaz responsive para computadora.
- **RNF\_2.** El sistema ha de tener una versión móvil que trabaje en iOS y Android.
- **RNF\_3.** El sistema debe tener bien establecidas las funciones y permisos para 3 roles diferentes: Operadoras de logística, Repartidores, Clientes.

### SHOULD

- **RF\_11.** Que el rol Operadoras de logística pueda hacer cambios sobre rutas definidas.
- **RF\_12.** Que el rol Operadoras de logística pueda tener seguimiento en tiempo real, es decir, poder ver en qué parte de la ciudad están los repartidores
- **RF\_15.** Que el rol Repartidor pueda recibir actualizaciones si hay cambios.

### COULD

- **RF\_3.** Que el rol Operadoras de logística pueda ver estadísticas sobre el desempeño de los repartidores
- **RF\_17.** Que el rol Repartidor pueda visualizar su ruta a través de un mapa.

### WON'T

- **RF\_4.** Que el rol Operadoras de logística pueda recibir los pedidos directamente desde Whatsapp a su interfaz.
- **RF\_8.** El sistema debe asignar de forma automática las rutas a los repartidores.
- **RF\_19.** Que el rol Cliente pueda cargar pedidos directamente desde la app

**Puntos Fibonacci con argumentación**

Requisito	Estimación	Argumento
RF_1	5	A pesar de no contar con mucha experiencia haciendo un login, actualmente se cuentan con muchas herramientas como apis de autenticación que facilitan bastante el proceso.
MUST		
RF_2	3	No se cuenta con experiencia manejando sockets o APIs para mantener la información actualizada de manera recurrente
MUST		
RF_3	0.5	Hay plenitud de librerías para análisis de datos.
COULD		
RF_4	21	No se cuenta con experiencia para integrar aplicaciones externas (Como Whatsapp) dentro de un sistema propio
WONT		
RF_5	2	La lectura de archivos es una tarea fácil de implementar, la mayor parte del tiempo será empleado para definir el formato del archivo.
MUST		
RF_6	5	Lo más complejo de esta tarea es definir las reglas del negocio, como los criterios y atributos necesarios para asignar una entrega.
MUST		
RF_7	2	Visualizar el estado de los repartidores implica consultar su ubicación y estado, lo cual puede resolverse con una consulta a la base de datos o integración con GPS.
MUST		
RF_8	13	Requiere conocer de forma profunda la lógica del negocio, saber en qué se basan las Operadoras al asignar de forma manual para automatizar el proceso.
WONT		
RF_9	2	Similar a RF_7, consiste en consultar y filtrar pedidos según su estado, una funcionalidad básica de CRUD.
MUST		
RF_10	5	Lo más complejo es conseguir una API de geolocalización que facilite trazar rutas nuevas.
MUST		
RF_11	2	Modificar rutas existentes es menos complejo que crearlas (RF_10), ya que solo

SHOULD		requiere actualizar registros en la base de datos con validaciones similares.
RF_12	5	Al igual que RF_10, lo más complejo es el API para la geolocalización
SHOULD		
RF_13	0.5	Solo requiere un par de consultas sobre la base de datos del sistema.
MUST		
RF_14	0.5	Es una de las operaciones CRUD sobre la base de datos, su implementación es tan compleja como RF_13
MUST		
RF_15	2	A pesar de que el requerimiento se satisface con una notificación, no tenemos experiencia para implementar fácilmente.
SHOULD		
RF_16	0.5	Solo requiere un par de consultas sobre la base de datos del sistema.
MUST		
RF_17	5	Al igual que RF_10, lo más complejo es el API para la geolocalización
COULD		
RF_18	2	Similar a RF_7, consiste en consultar y filtrar pedidos según su estado, una funcionalidad básica de CRUD.
MUST		
RF_19	3	Similar a RF_8, es necesario conocer la lógica del negocio para desarrollar una interfaz que sea intuitiva a los usuarios y tenga las herramientas que suelen usar actualmente.
WONT		
RNF_1	8	Considerando la complejidad de mantener consistencia visual en todos los roles, y que es importante establecer elementos críticos a la hora de adaptar la interfaz.
MUST		
RNF_2	21	Por la falta de experiencia. Nunca hemos hecho una aplicación móvil, por lo que una investigación es crucial para implementar una interfaz que trabaje en Android y iOS.
MUST		
RNF_3	0,5	Es una tarea conceptual más que de implementación
MUST		