



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

Universidad Nacional de Colombia - sede Bogotá  
Facultad de Ingeniería  
Departamento de Sistemas e Industrial  
Curso: Ingeniería de Software 1 (2016701)

**Estudiantes:** Juan David Ardila Pacheco , Mauro Alejandro Suesca López ,  
Juan Camilo Camargo Prieto , Samuel Camilo Burgos Rojas

### Lista de requerimientos:

1. Los **administradores** deben tener acceso a la ubicación de los **repartidores**.
2. Los **administradores** deben identificar qué **repartidor** está disponible.
3. Los **administradores** deben ver el estado de las entregas de los **repartidores**.
4. Los **administradores** deben tener la capacidad de crear rutas para los repartidores, por medio del uso de aplicaciones ya existentes como waze o google maps para crear la ruta a partir de un punto de salida y uno de llegada.
5. Los **administradores** deben tener la capacidad de modificar las rutas de los repartidores de forma sencilla, específicamente en menos de 5 pasos, haciendo uso de aplicaciones ya existentes como waze o google maps para modificar la ruta a partir de un punto de salida y uno de llegada.
6. Los **administradores** deben tener la capacidad de ver los pedidos existentes.
7. Los **administradores** deben tener la capacidad de modificar los pedidos de los repartidores de forma sencilla, seleccionando al repartidor de una lista de repartidores y los pedidos de ese repartidor de una lista de pedidos del repartidor.
8. Los **administradores** deben tener la capacidad de cargar los pedidos desde un archivo csv al sistema.
9. Los **repartidores** deben ver las rutas de sus propias entregas.
10. Los **repartidores** deben tener la capacidad de marcar sus entregas como "completadas", "retrasadas" o "en camino".
11. Los **repartidores** deben ver las actualizaciones realizadas a sus pedidos.
12. Los **repartidores** deben tener la capacidad de ver su lista de pedidos del día.
13. Los **clientes** deben tener la capacidad de crear pedidos.
14. Los **clientes** deben tener acceso al estado de sus propias entregas.
15. Los **clientes** deben tener la capacidad de modificar el destino de sus propios pedidos.
16. Los **clientes** deben tener la capacidad de modificar el contenido de sus propios pedidos.
17. Los **clientes** deben tener la capacidad de cancelar sus propios pedidos.
18. Los **administradores** deben tener acceso al sistema a través de una interfaz para computador.
19. Los **repartidores** deben tener acceso al sistema a través de una interfaz para celular.

20. La **interfaz** para celular debe ocupar menos de 100 MB de almacenamiento y 500 MB de memoria RAM.
21. El **sistema** debe notificar a los **clientes** cuando su entrega está en camino.
22. El **sistema** debe notificar a los **clientes** cuando su entrega se encuentra retrasada.
23. El **sistema** debe notificar a los **clientes** cuando su entrega se completa.
24. El **sistema** debe funcionar en tiempo real, actualizando la ubicación de los repartidores y el estado de los pedidos en el momento en el que ocurre.
25. El **sistema** debe generar informes de estadísticas con la información de tiempos de entrega, zonas “problemáticas” y repartidores más veloces. (**zonas “problemáticas” es demasiado ambiguo y es necesario hablar con el cliente al respecto**)
26. Los **clientes** deben tener la capacidad de crear pedidos a través de WhatsApp.
27. El **sistema** debe generar respuestas automáticas informando que el negocio está fuera del horario de atención.
28. El **sistema** debe tener una funcionalidad de “log in” para autenticar a los distintos usuarios (**administradores, repartidores y clientes**).

Funcionales	No Funcionales
RF_1: Los <b>administradores</b> deben tener acceso a la ubicación de los <b>repartidores</b> .	RNF_1: Los <b>administradores</b> deben tener acceso al sistema a través de una interfaz para computador.
RF_2: Los <b>administradores</b> deben identificar qué <b>repartidor</b> está disponible.	RNF_2: Los <b>repartidores</b> deben tener acceso al sistema a través de una interfaz para celular.
RF_3: Los <b>administradores</b> deben ver el estado de las entregas de los <b>repartidores</b> .	RNF_3: La <b>interfaz</b> para celular debe ocupar menos de 100 MB de almacenamiento y 500 MB de memoria RAM.
RF_4: Los <b>administradores</b> deben tener la capacidad de crear rutas para los repartidores, por medio del uso de aplicaciones ya existentes como waze o google maps para crear la ruta a partir de un punto de salida y uno de llegada.	RNF_4: El <b>sistema</b> debe funcionar en tiempo real, actualizando la ubicación de los repartidores y el estado de los pedidos en el momento en el que ocurre.
RF_5: Los <b>administradores</b> deben tener la capacidad de modificar las rutas de los repartidores de forma sencilla, específicamente en menos de 5 pasos, haciendo uso de aplicaciones ya existentes como waze o google maps para modificar la ruta a partir de un punto de salida y uno de llegada.	
RF_6: Los <b>administradores</b> deben tener la capacidad de modificar los pedidos de los	

repartidores de forma sencilla, seleccionando al repartidor de una lista de repartidores y los pedidos de ese repartidor de una lista de pedidos del repartidor.	
RF_7: Los <b>administradores</b> deben tener la capacidad de ver los pedidos existentes.	
RF_8: Los <b>administradores</b> deben tener la capacidad de cargar los pedidos desde un archivo csv al sistema.	
RF_9: Los <b>repartidores</b> deben ver las rutas de sus propias entregas.	
RF_10: Los <b>repartidores</b> deben tener la capacidad de marcar sus entregas como "completadas", "retrasadas" o "en camino".	
RF_11: Los <b>repartidores</b> deben ver las actualizaciones realizadas a sus pedidos.	
RF_12: Los <b>repartidores</b> deben tener la capacidad de ver su lista de pedidos del día.	
RF_13: Los <b>clientes</b> deben tener la capacidad de crear pedidos.	
RF_14: Los <b>clientes</b> deben tener acceso al estado de sus propias entregas.	
RF_15: Los <b>clientes</b> deben tener la capacidad de modificar el destino de sus propios pedidos.	
RF_16: Los <b>clientes</b> deben tener la capacidad de modificar el contenido de sus propios pedidos.	
RF_17: Los <b>clientes</b> deben tener la capacidad de cancelar sus propios pedidos.	
RF_18: El <b>sistema</b> debe notificar a los <b>clientes</b> cuando su entrega está en camino.	
RF_19: El <b>sistema</b> debe notificar a los <b>clientes</b> cuando su entrega se encuentra retrasada.	
RF_20: El <b>sistema</b> debe notificar a los <b>clientes</b> cuando su entrega se completa.	

RF_21: Los <b>clientes</b> deben tener la capacidad de crear pedidos a través de WhatsApp.	
RF_22: El <b>sistema</b> debe generar respuestas automáticas informando que el negocio está fuera del horario de atención.	
RF_23: El <b>sistema</b> debe generar informes de estadísticas con la información de tiempos de entrega, zonas “problemáticas” y repartidores más veloces. ( <b>zonas “problemáticas” es demasiado ambiguo y es necesario hablar con el cliente al respecto</b> )	
RF_24: El <b>sistema</b> debe tener una funcionalidad de “log in” para autenticar a los distintos usuarios ( <b>administradores, repartidores y clientes</b> ).	

Must have	Should have	Could have	Won't have
RF_1: Los <b>administradores</b> deben tener acceso a la ubicación de los <b>repartidores</b> .	RF_6: Los <b>administradores</b> deben tener la capacidad de modificar los pedidos de los repartidores de forma sencilla, seleccionando al repartidor de una lista de repartidores y los pedidos de ese repartidor de una lista de pedidos del repartidor.	RF_8: Los <b>administradores</b> deben tener la capacidad de cargar los pedidos desde un archivo csv al sistema.	RNF_3: La <b>interfaz</b> para celular debe ocupar menos de 100 MB de almacenamiento y 500 MB de memoria RAM.
RF_2: Los <b>administradores</b> deben identificar qué <b>repartidor</b> está disponible.	RF_15: Los <b>clientes</b> deben tener la capacidad de modificar el destino de sus propios pedidos.	RF_21: Los <b>clientes</b> deben tener la capacidad de crear pedidos a través de WhatsApp.	
RF_3: Los <b>administradores</b> deben ver el estado	RF_16: Los <b>clientes</b> deben tener la capacidad de	RF_22: El <b>sistema</b> debe generar respuestas	

de las entregas de los <b>repartidores</b> .	modificar el contenido de sus propios pedidos.	automáticas informando que el negocio está fuera del horario de atención.	
RF_4: Los <b>administradores</b> deben tener la capacidad de crear rutas para los repartidores, por medio del uso de aplicaciones ya existentes como waze o google maps para crear la ruta a partir de un punto de salida y uno de llegada.	RF_17: Los <b>clientes</b> deben tener la capacidad de cancelar sus propios pedidos.	RF_23: El <b>sistema</b> debe generar informes de estadísticas con la información de tiempos de entrega, zonas “problemáticas” y repartidores más veloces. ( <b>zonas “problemáticas” es demasiado ambiguo y es necesario hablar con el cliente al respecto</b> )	
RF_5: Los <b>administradores</b> deben tener la capacidad de modificar las rutas de los repartidores de forma sencilla, específicamente en menos de 5 pasos, haciendo uso de aplicaciones ya existentes como waze o google maps para modificar la ruta a partir de un punto de salida y uno de llegada.			
RF_7: Los <b>administradores</b> deben tener la capacidad de ver los pedidos existentes.			
RF_9: Los <b>repartidores</b> deben ver las rutas de sus propias entregas.			

RF_10: Los <b>repartidores</b> deben tener la capacidad de marcar sus entregas como "completadas", "retrasadas" o "en camino".			
RF_11: Los <b>repartidores</b> deben ver las actualizaciones realizadas a sus pedidos.			
RF_12: Los <b>repartidores</b> deben tener la capacidad de ver su lista de pedidos del día.			
RF_13: Los <b>clientes</b> deben tener la capacidad de crear pedidos.			
RF_14: Los <b>clientes</b> deben tener acceso al estado de sus propias entregas.			
RF_18: El <b>sistema</b> debe notificar a los <b>clientes</b> cuando su entrega está en camino.			
RF_19: El <b>sistema</b> debe notificar a los <b>clientes</b> cuando su entrega se encuentra retrasada.			
RF_20: El <b>sistema</b> debe notificar a los <b>clientes</b> cuando su entrega se completa.			

RF_24: El <b>sistema</b> debe tener una funcionalidad de "log in" para autenticar a los distintos usuarios ( <b>administradores, repartidores y clientes</b> ).			
RNF_1: Los <b>administradores</b> deben tener acceso al sistema a través de una interfaz para computador.			
RNF_2: Los <b>repartidores</b> deben tener acceso al sistema a través de una interfaz para celular.			
RNF_4: El <b>sistema</b> debe funcionar en tiempo real, actualizando la ubicación de los repartidores y el estado de los pedidos en el momento en el que ocurre.			

Requisito		Estimación	Argumento
RF_1	Los <b>administradores</b> deben tener acceso a la ubicación de los <b>repartidores</b> .	8	Nunca hemos manejado ubicaciones, sería necesario hacer integración con alguna aplicación con el gps.
MUST			
RF_2	Los <b>administradores</b> deben identificar qué	1	Es una consulta sencilla, dado que otro requisito le da la posibilidad a los repartidores de marcar

MUST	<b>repartidor</b> está disponible.		el estado de sus pedidos como completados, y si no tienen pedidos incompletos, significaría que están disponibles.
RF_3	Los <b>administradores</b> deben ver el estado de las entregas de los <b>repartidores</b> .	1	Es una consulta con información que debería estar en la base de datos.
MUST			
RF_4	Los <b>administradores</b> deben tener la capacidad de crear rutas para los repartidores, por medio del uso de aplicaciones ya existentes como waze o google maps para crear la ruta a partir de un punto de salida y uno de llegada.	8	Sería necesario hacer integración con una de las aplicaciones mencionadas en el requisito.
MUST			
RF_5	Los <b>administradores</b> deben tener la capacidad de modificar las rutas de los repartidores de forma sencilla, específicamente en menos de 5 pasos, haciendo uso de aplicaciones ya existentes como waze o google maps para modificar la ruta a partir de un punto de salida y uno de llegada.	8	Sería necesario hacer integración con una de las aplicaciones mencionadas en el requisito.
MUST			
RF_7	Los <b>administradores</b> deben tener la capacidad de ver los pedidos existentes.	1	Es una consulta con información que debería estar en la base de datos.
MUST			
RF_9	Los <b>repartidores</b> deben ver las rutas de sus propias entregas.	1	Es una consulta con información que debería estar en la base de datos.
MUST			
RF_10	Los <b>repartidores</b> deben tener la capacidad de marcar sus entregas como "completadas", "retrasadas" o "en camino".	1	Es una consulta sencilla y modificación en la base de datos.
MUST			



RF_11	Los <b>repartidores</b> deben ver las actualizaciones realizadas a sus pedidos.	1	Es una consulta con información que debería estar en la base de datos.
MUST			
RF_12	Los <b>repartidores</b> deben tener la capacidad de ver su lista de pedidos del día.	1	Es una consulta con información que debería estar en la base de datos.
MUST			
RF_13	Los <b>clientes</b> deben tener la capacidad de crear pedidos.	2	Es necesario crear las funciones que reciban los datos y revisen el formato de los mismos para ingresar los datos del pedido y luego hacer una inserción en la base de datos.
MUST			
RF_14	Los <b>clientes</b> deben tener acceso al estado de sus propias entregas.	1	Es una consulta con información que debería estar en la base de datos.
MUST			
RF_18	El <b>sistema</b> debe notificar a los <b>clientes</b> cuando su entrega está en camino.	5	Se necesitaría crear una función/trigger que detecte los cambios en el estado de los pedidos y haga que el programa envíe una notificación a los clientes.
MUST			
RF_19	El <b>sistema</b> debe notificar a los <b>clientes</b> cuando su entrega se encuentra retrasada.	5	Se necesitaría crear una función/trigger que detecte los cambios en el estado de los pedidos y haga que el programa envíe una notificación a los clientes.
MUST			
RF_20	El <b>sistema</b> debe notificar a los <b>clientes</b> cuando su entrega se completa.	5	Se necesitaría crear una función/trigger que detecte los cambios en el estado de los pedidos y haga que el programa envíe una notificación a los clientes.
MUST			
RF_24	El <b>sistema</b> debe tener una funcionalidad de "log in" para autenticar a los distintos usuarios ( <b>administradores</b> , <b>repartidores</b> y <b>clientes</b> ).	5	Autenticar usuarios es una tarea un poco complicada, porque necesitamos manejar bien los usuarios con sus respectivas contraseñas y para esto se necesitan hacer varias pruebas y tener una interfaz intuitiva para los usuarios.
MUST			
RNF_1	Los <b>administradores</b> deben tener acceso al sistema a través de una interfaz para computador.	13	Crear el frontend de escritorio para que sea comprensible, fácil de usar y estético es una tarea demandante que implica pruebas, bocetos, iteraciones y muchas decisiones.
MUST			
RNF_2	Los <b>repartidores</b> deben tener acceso al sistema a través de una interfaz para celular.	13	Crear el frontend de celular para que sea comprensible, fácil de usar y estético es una tarea demandante que implica pruebas, bocetos, iteraciones y muchas decisiones.
MUST			
RNF_4	El <b>sistema</b> debe funcionar en tiempo	2	Teóricamente no es una tarea compleja, por lo

MUST	real, actualizando la ubicación de los repartidores y el estado de los pedidos en el momento en el que ocurre.		tanto se le otorga un puntaje bajo, sin embargo, dada la nula experiencia del equipo de desarrolladores en este tema, podría llegar a tomar más tiempo que el esperado por la complejidad misma del requisito.
RF_6	Los <b>administradores</b> deben tener la capacidad de modificar los pedidos de los repartidores de forma sencilla, seleccionando al repartidor de una lista de repartidores y los pedidos de ese repartidor de una lista de pedidos del repartidor.	2	Es necesario crear las funciones que reciban los datos y revisen el formato de los mismos para ingresar los nuevos datos del pedido y luego hacer una modificación en la base de datos.
SHOULD			
RF_15	Los <b>clientes</b> deben tener la capacidad de modificar el destino de sus propios pedidos.	2	Es necesario crear las funciones que reciban los datos y revisen el formato de los mismos para ingresar los nuevos datos del pedido y luego hacer una modificación en la base de datos.
SHOULD			
RF_16	Los <b>clientes</b> deben tener la capacidad de modificar el contenido de sus propios pedidos.	2	Es necesario crear las funciones que reciban los datos y revisen el formato de los mismos para ingresar los nuevos datos del pedido y luego hacer una modificación en la base de datos.
SHOULD			
RF_17	Los <b>clientes</b> deben tener la capacidad de cancelar sus propios pedidos.	1	Es una consulta sencilla y eliminación en la base de datos.
SHOULD			
RF_8	Los <b>administradores</b> deben tener la capacidad de cargar los pedidos desde un archivo csv al sistema.	1	Es una tarea sencilla, ya que abrir y recuperar datos de un tipo de archivo en específico es muy fácil.
COULD			
RF_21	Los <b>clientes</b> deben tener la capacidad de crear pedidos a través de WhatsApp.	8	Sería necesario hacer una integración con whatsapp y posiblemente habría que crear un bot para que reciba los mensajes.
COULD			
RF_22	El <b>sistema</b> debe generar respuestas automáticas informando que el negocio está fuera del horario de atención.	3	Dado el mismo bot que se piensa crear en la integración con whatsapp, añadirle la funcionalidad de que detecte la hora con el fin de que responda automáticamente con un mensaje predeterminado según el horario del negocio no debería ser difícil teóricamente. Se coloca una
COULD			

			estimación más alta porque la falta de experiencia del equipo de desarrolladores en este concepto podría aumentar el tiempo que se necesita.
RF_23	El <b>sistema</b> debe generar informes de estadísticas con la información de tiempos de entrega, zonas “problemáticas” y repartidores más veloces. <b>(zonas “problemáticas” es demasiado ambiguo y es necesario hablar con el cliente al respecto)</b>	N/A	Se necesita hablar y consultar con el cliente directamente para averiguar el significado de “zonas problemáticas”, dado que es un término muy ambiguo.
COULD			
RNF_3	La <b>interfaz</b> para celular debe ocupar menos de 100 MB de almacenamiento y 500 MB de memoria RAM.	2	Las aplicaciones móviles en promedio no suelen ocupar esa cantidad de espacio o memoria RAM, por lo cual es una tarea que se podría cumplir por sí sola al terminar el programa de celular. Es por eso además, que se coloca en won't, porque no será un enfoque principal del desarrollo.
WONT			