# Functional Requirements

USERS: passenger, passenger driver, delivery requester, delivery man (can be a driver, or a person taking public transport)

1 A user must have an account to use the services that the application offers.
1.1 The user must be prompted to sign in or sign up upon opening the application.
1.1.1 The application shall redirect the user to another page for the sign-up process.
1.1.1.1 The sign-up page must contain the fields that the user is required to fill in.
1.1.1.1.1 The user shall be required to provide his/her email address.
1.1.1.1.2 The user shall be required to provide his/her phone number.
1.1.1.1.3 The user shall be required to create a password for his/her account.
1.1.1.1.4 The user shall be required to create a username for his/her account.
1.1.1.1.5 The user shall be provided the option to set his/her profile picture for his/her account.
1.1.1.1.5.1 If the user chooses to skip this option, the application shall set the profile picture as the default.
1.1.1.2 The user must make a confirmation that his/her details are correct before the account can be created.
1.1.1.3 The user shall be directed back to the sign-in page.
1.1.2 The user must provide his/her email address and password for the sign-in process.
1.1.2.1 If the user has forgotten his/her password, an option to reset the user's account password shall be provided.
1.1.3 The user must provide more details on his/her car if the user would like to deliver via car or drive a passenger.
1.1.3.1 The user must provide his/her car's license plate number.
1.1.3.2 The user must provide his/her car's model.
1.1.3.3 The user must provide his/her car's colour.
1.1.4 The user shall be directed to the user interface after a successful log-in.
1.1.4.1 The application shall display the user's account information on the user interface.
1.1.4.1.1 The application must display the user's profile picture.
1.1.4.1.2 The application must display the user's wallet.
1.1.4.1.2.1 The user shall be given the option to top-up money.
1.1.4.1.2.2 The user shall be given the option to transfer money from the wallet to his/her personal bank account.
1.1.4.1.2.2.1 The user shall be prompted to provide his/her bank account details.
1.1.4.1.3 The application shall display the user's past transactions.
1.1.4.1.3.1 The transaction display must contain the type of service that was used.
1.1.4.1.3.2 The transaction display must contain the amount deducted or added to the user's wallet as a result of using the service.
1.1.4.1.4 The application must display the user's obtained ratings.
1.1.4.1.5 The application must display the user's obtained reviews.

1.1.4.2 If the user has provided his/her car details beforehand, the application shall provide the user with the option to edit his/her car details.
1.1.4.3 The application shall provide the user with the option to change his/her password.

2 The application shall offer a passenger service.
2.1 A user shall be able to choose between being a passenger, or a driver.

2.1.1 A passenger must be able to use the application to lodge a carpool request.
2.1.1.1 A passenger must choose a location on the map as the pick-up point.
2.1.1.1.1 A passenger must only be allowed to choose one location as the pick-up point per request.
2.1.1.1.2 A passenger shall be allowed to change the pick-up point before confirming his/her route request.
2.1.1.2 A passenger must choose a location on the map as the drop-off point.
2.1.1.2.1 A passenger must only be allowed to choose one location as the drop-off point.
2.1.1.2.2 A passenger shall be allowed to change the drop-off point before confirmation of his/her route request.
2.1.1.3 A passenger must be able to view the estimated fare for the route between the pick-up and drop-off points.
2.1.1.4 A passenger must make a confirmation on the details of their carpool request before a match can be made.
2.1.1.4.1 If the amount in the passenger's wallet is less than the carpool fare, then the application must not allow the passenger to make a confirmation.
2.1.1.4.1.1 The passenger must be notified that there are not enough funds in his/her wallet.
2.1.1.5 A passenger shall receive an alert once a match has been found.
2.1.1.5.1 The match alert display must contain the driver's data.
2.1.1.5.1.1 The match alert display must contain the driver's username.
2.1.1.5.1.2 The match alert display must contain the driver's obtained ratings.
2.1.1.5.1.3 The match alert display must contain the driver's obtained reviews.
2.1.1.5.1.4 The match alert display must contain the driver's car license plate number.
2.1.1.5.1.5 The match alert display must contain the driver's car colour.
2.1.1.5.1.6 The match alert display must contain the driver's car model.
2.1.1.5.1.7 The match alert display must contain the driver's expected time of arrival at the pick-up point.
2.1.1.6 A match shall be confirmed when the driver has accepted the passenger's carpool request.
2.1.1.7 A passenger shall be able to assign a rating to the driver after the passenger has arrived at the drop-off point.
2.1.1.7.1 The rating must be on a scale from 1 to 5; 1 being the lowest and 5 being the highest.
2.1.1.8 A passenger shall be able to choose to write a review about the driver after the passenger has arrived at the drop-off point.

2.1.1.8.1 The review must be at least 0 characters or no longer than 512 characters.
2.1.1.9 The fare shall be deducted from the passenger's wallet once the passenger has been picked up from the pick-up point.

2.1.2 A driver shall be able to use the application to accept a passenger looking to carpool.
2.1.2.1 A driver must choose a location on the map as the start point of his/her route.
2.1.2.1.1 A driver must only be allowed to choose one location as the start point of the driver's route.
2.1.2.1.2 A driver shall be allowed to change the start point of the route before confirmation of the route details.
2.1.2.2 A driver must choose a location on the map as the end point of his/her route.
2.1.2.2.1 A driver must only be allowed to choose one location as the end point of the driver's route.
2.1.2.2.2 A driver shall be allowed to change the end point of the route before confirmation of the route details.
2.1.2.3 A driver must make a confirmation on the details of his/her route before carpool requests can be recommended.
2.1.2.4 A driver shall be able to view a list of carpool requests that have been sorted according to the driver's route using a predetermined algorithm.
2.1.2.4.1 The application must display carpool requests in the order of closeness of the match between the passenger's requested route matches with the driver's route
2.1.2.4.2 If the user's requested route deviates from the driver's route, the application must report the estimated extra time that it would take the driver to pick up the passenger and drop him/her off.
2.1.2.5 A carpool request display must contain passenger data and information about the shared route.
2.1.2.5.1 A carpool request display must contain the passenger's username.
2.1.2.5.2 A carpool request display must contain the passenger's obtained ratings.
2.1.2.5.3 A carpool request display must contain the passenger's pick-up point.
2.1.2.5.4 A carpool request display must contain the passenger's drop-off point.
2.1.2.6 A driver must input the estimated time of arrival at the passenger's pick-up point.
2.1.2.7 A driver must make a confirmation that he/she has accepted the carpool request to ferry the passenger.
2.1.2.8 The application shall assume a match is made when the passenger accepts the pick-up.
2.1.2.9 A driver shall be able to assign a rating to the passenger after he/she has alighted at the drop-off point.
2.1.2.9.1 The rating must be on a scale from 1 to 5; 1 being the lowest and 5 being the highest.
2.1.2.10 A driver shall be able to choose to write a review about the passenger after he/she has alighted at the drop-off point.
2.1.2.10.1 The review must be at least 0 characters or no longer than 512 characters.

2.1.2.11 The fare will be awarded to the driver's wallet after the passenger has alighted at the drop-off point.

3 The application shall offer a delivery service.
3.1 A user shall be able to choose between being a delivery man, or the delivery requester.
3.1.1 A delivery requester must be able to send a package through a matched delivery man via this application.
3.1.1.1 A delivery requester must choose a location on the map to be the pick-up point.
3.1.1.1.1 A delivery requester must only be allowed to choose one location to be the pick-up point, per request.
3.1.1.1.2 A delivery requester shall be allowed to change the pick-up point before confirmation of the delivery request.
3.1.1.2 A delivery requester must choose a location on the map to be the drop-off point.
3.1.1.2.1 A delivery requester must only be allowed to choose one location to be the drop-off point.
3.1.1.2.2 A delivery requester shall be allowed to change the drop-off point before confirmation of the delivery request.
3.1.1.3 A delivery requester must be able to add a description of the receiver of the delivered items for the delivery man to verify.
3.1.1.4 A delivery requester must be able to view the estimated delivery fee for the route between the pick-up and drop-off points.
3.1.1.5 A delivery requester must make a confirmation of the details of the delivery before a match can be made.
3.1.1.5.1 If the amount in the delivery requester's wallet is less than what is required for the delivery fee, the application must not allow the delivery requester to make the confirmation.
3.1.1.5.1.1 The delivery requester must be notified that there are not enough funds in his/her wallet.
3.1.1.6 A delivery requester shall receive an alert when a match has been made.
3.1.1.6.1 The match alert display must contain delivery man data.
3.1.1.6.1.1 The match alert display must contain the delivery man's username.
3.1.1.6.1.2 The match alert display must contain the delivery man's obtained ratings.
3.1.1.6.1.3 The match alert display must contain the delivery man's obtained reviews.
3.1.1.6.1.4 If the delivery man will be delivering via his/her own car, details about the delivery man's car must be contained in the match alert display.
3.1.1.6.1.4.1 The match alert display must contain the delivery man's car license plate number.
3.1.1.6.1.4.2 The match alert display must contain the delivery man's car model.
3.1.1.6.1.4.3 The match alert display must contain the delivery man's car brand.
3.1.1.6.1.5 If the delivery man is delivering using other modes of transport, details about the delivery man in the form of a description must be displayed.
3.1.1.6.1.6 The match alert display must contain the delivery man's expected time of arrival at the pick-up point.
3.1.1.7 A match shall be made when the delivery man has accepted the delivery request.

3.1.1.8 A delivery requester must be notified after the package has been delivered.
3.1.1.8.1 The notification must contain the time of delivery.
3.1.1.9 The delivery fee shall be deducted from the delivery requester's wallet after the package has been delivered.
3.1.1.10 A delivery requester shall be able to leave a rating for the delivery man's delivery.
3.1.1.10.1 The rating must be on a scale of 1 to 5; 1 being the lowest and 5 being the highest.
3.1.1.11 A delivery requester shall be able to write a review for the delivery man's delivery.
3.1.1.11.1 The review must at least be 0 characters, but no longer than 512 characters.

3.1.2 A delivery man shall be able to use the application to accept a delivery.
3.1.2.1 A delivery man must choose a location on the map to be the start of the delivery man's route.
3.1.2.1.1 A delivery man must only be allowed to choose one location to be the start point of his/her route.
3.1.2.1.2 A delivery man shall be allowed to change the start point of his/her route before confirmation of the route is given.
3.1.2.2 A delivery man must choose a location on the map to be the end of the delivery man's route.
3.1.2.2.1 A delivery man must only be allowed to choose one location to be the end point of his/her route.
3.1.2.2.2 A delivery man shall be allowed to change the end point of his/her route before confirmation of the route is given.
3.1.2.3 A delivery man must be prompted by the application to provide a confirmation on the details of the route before delivery requests can be recommended.
3.1.2.4 A delivery man shall be able to view a list of delivery requests that are sorted according to the delivery man's route.
3.1.2.4.1 The application must display delivery requests in the order of closeness of the match between the route of the delivery request and the route of the delivery man.
3.1.2.4.2 If the delivery route deviates from the delivery man's route, the application must report the estimated extra time that it would take for the delivery man to make the delivery.
3.1.2.5 A delivery request display must contain delivery requester data and information about the shared route.
3.1.2.5.1 A delivery request display must contain the delivery requester's username.
3.1.2.5.2 A delivery request display must contain the delivery requester's phone number.
3.1.2.5.3 A delivery request display must contain the description of the delivery receiver, which shall be provided by the delivery requester.
3.1.2.5.4 A delivery request display must contain the delivery's pick-up point.
3.1.2.5.5 A delivery request display must contain the delivery's drop-off point.
3.1.2.6 A delivery man must indicate whether he/she will be delivering the package by car.
3.1.2.6.1 A delivery man must be prompted to provide details of his/her car if the delivery man has not yet provided them beforehand.

3.1.2.6.2 If the delivery man indicates that he/she will be delivering the package by other modes of transportation, the delivery man must input a description of himself/herself.
3.1.2.6.2.1 The description must be at least 50 characters, but no longer than 600 characters.
3.1.2.7 A delivery man must make a confirmation that he/she has accepted the delivery request.
3.1.2.8 The application shall assume that a match has been made when the delivery requester has confirmed the match.
3.1.2.9 The delivery fee shall be awarded to the delivery man's wallet after the delivery items have been dropped off at the drop-off point.

# Non-Functional Requirements Document

1 The application must fulfill the following security requirements.
1.1 All user data must be stored securely to ensure that no unauthorized access is possible.
1.1.1 The user's password must be hashed in the frontend and encrypted in the backend to ensure password security.
1.2 The application shall make use of security measures such as JWT tokens or session tokens to ensure that user sessions are not compromised and accessible to others.

2 The application must fulfill the following reliability requirements.
2.1 The application must have a 99.99% uptime.

3 The application must fulfill the following usability requirements.
3.1 95% of users must be able to make a carpool or delivery request within a minute of entering the application.
3.2 95% of users must indicate that they are familiar with navigating the user interface within 5 minutes.
3.3 A user shall be given the option to change the language options of the application.
3.4 A user must be able to reset his/her password within 5 minutes.
3.5 The application must verify each user action with proper exception handling to ensure that the application does not crash due to erroneous user actions. No crashes should happen as a result of user actions.

3 The application must fulfill the following portability requirements.
3.1 The application shall be written in a flexible design to integrate future extended capabilities without having to replace the entire codebase.
3.2 The database must be replaceable with any commercial product supporting standard SQL queries.

4 The application must have the following performance requirements.
4.1 The application must be made to scale so that users do not face any compromise in performance.
4.1.1 The load capacity must be a minimum of 1.4 times the expected/average load capacity at any instance.
4.1.2 If more than one backend server is used, a load balancer must be used to equally balance the load among all the servers to prevent crashes.
4.2 All computationally intensive workloads such as finding a match for passengers must be offloaded to a backend server to reduce the load on the local CPU.

5 The application must have the following reliability requirements.

5.1 The application must make provisions for regular data backups to prevent loss of data in case of any failure.

5.1.1 The database must be backed up every day and the backup should be maintained for at least a period of 2 days before being removed.