**System Requirement Specifications**

**Document: System Requirement Specification Document**

**Title**: Riders Point.

**Objective:**

It is going to provide communication environment between users (riders and pillions). This application is to help people who loves travelling in group. It will help people to gather together and go an on ride to discover places. Even if someone don't have a bike, he/she can go on ride with others.

**Scope:**

* Riders can set their route, date and see the lists of other riders available on that route to go with group by meeting at specific location decided by them after mutual agreement.
* Rider can set their route and date and pillion can communicate with them via phone number. After mutual agreement with each other, they can share the ride together.
* Users can access or communicate with each other via phone numbers.
* At some extent, this system will bring advantages like less traffic and air pollution will be decreased.
* Every user has their own profiles and they can have access with given password to the system.

**Overview:**

* Anyone can use this application who are above 18 age and go on a ride but in order to go on a ride, a user needs to log in using their unique email-id & password.
* By visiting the registration page, unregistered members can do so. The default roles are Rider and Pillion when a user registers on application.
* This system can be used by any user. However, for riders it is must that every rider has his driving license and bike document details.

**Functional Requirement:**

This section provides a requirement overview of system. Various functional modules that can be implemented by system will be:

* The users will be accessible for viewing their personal details and their added rides will be visible to every user.
* **Registration:** New users can sign up by using their unique email-id & password.
* **Login:** Only registered users can login to the system. Users must have valid login email-id and password to enter the application.
* **Add ride route:** User may add route by specifying date, start and end points and time period.
* **Delete ride route:** A user may delete his/her ride route. After deleting route, it will be informed to pillion if any.
* **Accept Request:** Rider will be able to see requests sent by users who wants to share a ride with him/her and will be able to accept it.
* **Search Route**: A user (pillion) can search for route so that he can share ride by specifying start and end point, time & date.
* **Send Request:** A user may want to share ride with rider so he can send a ride request to the rider.
* **Cancellation request from pillion:** A pillion may want to cancel his request so after cancellation it will be informed to the rider.
* **Logout:** A user may want to logout the system So he/she can do it by clicking the logout button.

**Non- Functional Requirements:**

**Performance Requirements:**

* The server must be able to support an unlimited number of devices, i.e., it must place no restrictions if the number of users use it simultaneously.
* Application should work fast and should be able to handle increasing data traffic.

**Availability:**

* This application is available for 24 hours anywhere, anytime.

**Accessibility:**

* After registration, only logged-in users will be able to add or update the ride.
* Through a personalized dashboard, the team will be able to monitor daily, weekly, monthly, and annual growth of users.

**Security Requirements:**

* Passwords of the user shall be encrypted in DBMS for security purposes.
* Protection against malfunction.

**Efficiency:**

* The system will be able to manage all users efficiently.

**Scalability:**

* The system should be scalable to accommodate changes in application and future growth.

**Assumptions and Dependencies:**

* Users will have stable internet connection.
* Users will provide accurate personal details.
* Riders must have driving license and provide accurate bike details.