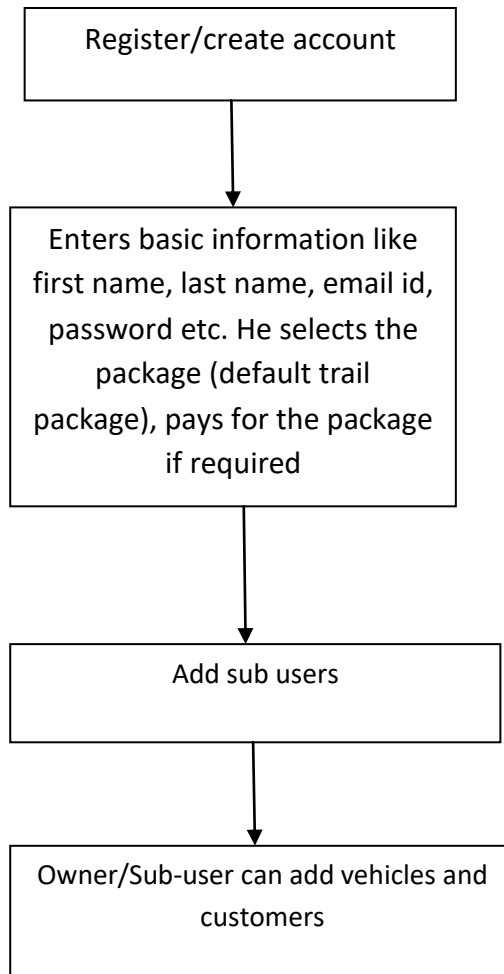
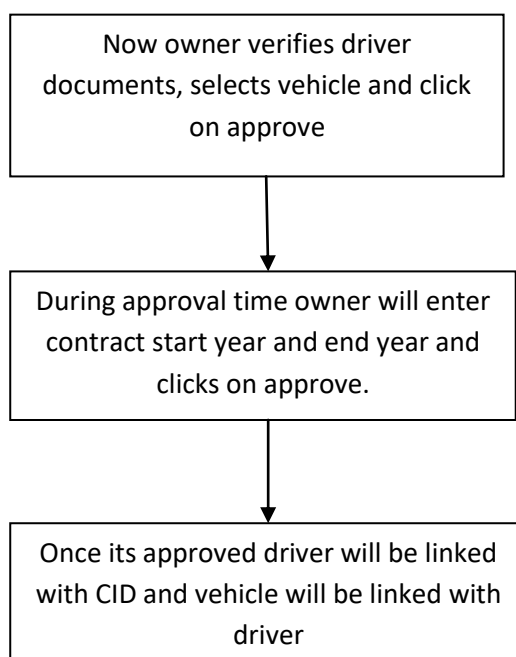


GMC

Owner workflow:



Pending request handling by Owner:



Owner journey (Vendor):

- Owner registers himself by giving basic information.
- While registering owner by default gets 1month trail period to access the application.
- If owner directly selects another package he will be redirected to payment page.

Email	<input type="text"/>
Password	<input type="text"/>
Name	<input type="text"/>
Phone no	<input type="text"/>
Photo	<input type="text"/>
House No	<input type="text"/>
Landmark	<input type="text"/>
Area/City/Pincode	<input type="text"/>
Company name	<input type="text"/>
GST	<input type="text"/>
PAN	<input type="text"/>
CIN	<input type="text"/>
Company Logo	<input type="text"/>
Package	<input type="text"/>
Gender	<input type="text"/>
Subscription	<input type="text"/>
<input type="button" value="Register"/>	

To get the subscription types an API need to be called.

API call: rootURL/api/User/Registration

API call: rootURL/api/Admin/Subscription (2 API's)(will be shared at last)

API Documentation:

Request

Method	URL
POST	rootURL/api/User/Registration

Type	Params	Values
POST	<div>UserResource<ul style="list-style-type: none">EmailPasswordNamePhone numberDisplayPictureGender</div> <div>Organization<ul style="list-style-type: none">OrganizationNameGSTCINPANCompanyLogoPassPhraseCid</div> <div>Address<ul style="list-style-type: none">HouseNoLandmarkAreaCityCountryLatLngTagPinCode</div> <div>Subscription<ul style="list-style-type: none">SubscriptionIDSubscriptionNameSubscriptionTypeDurationIsEnabledValidityAppNameDescription</div>	<div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>OrganizationResource</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>long</div> <div>AddressResource</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>double</div> <div>double</div> <div>string</div> <div>string</div> <div>SubscriptionResource</div> <div>Long</div> <div>String</div> <div>String</div> <div>Double</div> <div>Bool</div> <div>DateTime</div> <div>String</div> <div>String</div>

Response

Status	Response
200	UserResource
400	{"error":"Email or Phone no already exists."}

API Documentation:

Request

Method	URL
GET	rootURL/api/Admin/Subscriptions

Response

Status	Response
200	List of SubscriptionResource
400	{"error":"Email or Phone no already exists."}

- At the time of registration owner select subscription plan and once he clicks on register it will be re-directed to payment page where owner need to do payment for particular plan.
- Owner logins with registered mail id and password.

Login

Password

Login

Forgot Password????

- Once user logs into the web application he will get access to add sub-users, vehicles, customers, approve drivers, view documents, view orders, allocate vehicles, view reports, place orders, view maps, view subscription
- Here menu list will be displayed based on roles.
- For Admin : Add users, add vehicles, approve vehicles, view documents, promo code master, Change subscription,
- For Dispatcher: Add customers, place orders, view orders, allocate vehicles, maps, trip dashboard.
- For billing : Billing report, Package master
- For Tracker: Maps, Trip dashboard
- Vehicle allocator: Vehicle allocation.

API call: rootURL/api/Auth/login

returns: jwt token

- If sign in fails should display pop-up as “enter valid credentials”.

API Documentation:

Authenticate the user with the system and obtain the jwt_token.

- Here token will be in base64.
- If the token is decoded, then we get list of roles.

Request

Method	URL
POST	rootURL/api/Auth/login

Type	Params	Values
POST	Username	String
POST	Password	String

Response

Status	Response
200	<pre>{ "auth_key": <auth_key> }</pre> <p>auth_key(string) - all further API calls must have this key in header</p>
401	{"error": "Enter valid credentials."}

- If owner forgets his password, he can reset it by clicking on for password.

Email id

Submit

- Once owner clicks on submit a link will be shared to his given mail id to reset his password.

API call: rootURL/api/Master/ForgotPassword?Email=value

API call: rootURL/api/Master/UpdatePassword?Encrypted=value&Password=value

Request

Method	URL
POST	rootURL/api/Master/ForgotPassword?Email=value

Type	Params	Values
FromQuery	Email	String

Response

Status	Response
200	String (Encrypted email)
400	{"error": "Bad request."}

Request

Method	URL
POST	rootURL/api/Master/UpdatePassword?Encrypted=value&Password=value

Type	Params	Values
FromQuery	Encrypted	String
FromQuery	Password	String

Response

Status	Response
200	Bool
400	{"error": "Bad request."}

- Once user login into the web application he will get an initial screen will all his details from where he can change only his display picture.
- In this screen owner can access other web pages from menu bar.

- Add sub-users
- Add vehicles
- Approve drivers
- View documents
- Promo code
- Add customers
- Place orders
- View orders
- Allocate vehicles
- Maps
- Billing
- Subscriptions
- Change password

Email
Password
Name
Phone no
Photo

Upload

Gender
House No
Landmark
Area/City/Pincode
Company name
GST
PAN
CIN
Company Logo
Package
Subscription

Save

API call: rootURL/api/User/UserId?DisplayPicture=value

API Documentation:

Request

Method	URL
PUT	rootURL/api/User/UserId?DisplayPicture=value

Type	Params	Values
FromRoute	UserId	Long
FromQuery	DisplayPicture	string

Response

Status	Response
200	UserResource
401	{"error": "Enter valid credentials."}

Add sub-users:

To add users' owner should give basic information of users like EmailID, password, Name, Phone number, photo, company name, company logo, address.

Name

Phone number

Email ID

Password

Photo

Roles

Admin/Dispatcher/Allocator/Tracker/Billing ▾

API call: rootURL/api/User/RegisterUser

API call: rootURL/api/Admin/UserRoles

API Documentation:

Only OWNER can add sub-users.

Request

Method	URL
POST	rootURL/api/User/RegisterUser

Type	Params	Values
POST POST	UserResource <ul style="list-style-type: none">EmailPasswordNamePhone numberDisplayPictureRole	string string string string string String

Response

Status	Response
200	UserResource
400	{"error": "Bad request."}
401	Unauthorized

Request

Method	URL
GET	rootURL/api/Admin/UserRoles

Type	Params	Values
GET		

Response

Status	Response
200	Role ListResource

Add vehicles:

To add vehicles owner should give vehicle information like manufacturer, colour, Model year, Photo, Contract, Short name, contract type.

Add Vehicle

Manufacturer

Reg no

Colour

City

Model year

Occupancy

Contract type

Contract

Start date

End date

Short name

Photo

Upload

Add

API call: rootURL/api/Master/Vehicle

API call: rootURL/api/Master/VehicleType

API Documentation:

OWNER and USER can add vehicles.

Request

Method	URL
POST	rootURL/api/Master/Vehicle

Type	Params	Values
POST POST	VehicleResource <ul style="list-style-type: none">RegNoManufacturer VehicleTypeResource	String VehicleTypeResource

	<ol style="list-style-type: none"> 1. VTypeId 2. ManufacturerName 3. Name 4. Occupance 5. Icon 6. ManufacturerIcon <ul style="list-style-type: none"> • Color • City • ModelYear • Photo • Contract 	Long String String Int String String String String Long String ContractResource
	ContractResource <ol style="list-style-type: none"> 1. ContractType 2. StartContract 3. EndContract • ShortName 	String DateTime DateTime String

Response

Status	Response
200	VehicleResource
401	{"error": "Unauthorized."}

Request

Method	URL
GET	rootURL/api/Master/VehicleType

Response

Status	Response
200	List of VehicleTypeResource
401	{"error": "Unauthorized."}

Approve drivers:

- Here owner will approve driver by verifying his details and uploaded documents once he gets approval request from driver.

- At the time of approval owner will select vehicle, enters contract start year and end year and clicks on approve.
- Once its approved driver will be linked with CID and vehicle will be linked with driver

List of drivers

Driver name, Driverid, phone no

- Once owner clicks on driver name he will get screen where can see all the driver details with allocate vehicle button.

Driver ID XYZ
Contract type XYZ
Vehicle number

APX0258B

Allocate vehicle

Start year

End year

Approve

API call: rootURL/api/Master/Link/DriverId?ContractType=value&EndContract=DateTime

API call: rootURL/api/Master/Vehicles

API Documentation:

OWNER and USER can approve drivers.

Request

Method	URL
PUT	rootURL/api/Master/Link/DriverId?ContractType=value&EndContract=DateTime

Type	Params	Values
FromBody	VehicleResource <ul style="list-style-type: none"> RegNo Manufacturer VehicleTypeResource <ol style="list-style-type: none"> VTypeId ManufacturerName Name Occupance Icon ManufacturerIcon <ul style="list-style-type: none"> Color City ModelYear Photo Contract ContractResource <ol style="list-style-type: none"> ContractType StartContract EndContract <ul style="list-style-type: none"> ShortName 	String VehicleTypeResource Long String String Int String String String String String String Long String ContractResource String DateTime DateTime String
FromRoute	DriverId	Long
FromQuery	ContractType	String
FromQuery	EndContract	DateTime

Response

Status	Response
200	DriverResource
401	{"error": "Unauthorized."}

- Here Cid value should be logged in user Cid.

Request

Method	URL
GET	rootURL/api/Master/Vehicles

Type	Params	Values
FromQuery	Cid	long

Response

Status	Response
200	List of VehicleResource
401	{"error":"Unauthorized."}

View documents:

- Here owner can view all the documents that are uploaded by driver for verification.
- Here Cid will be hard coded.

Document type
Status

View

API call: rootURL/api/User/Documents

API Documentation:

OWNER,USER,SuperAdmin can access or view documents.

Request

Method	URL
POST	rootURL/api/User/Documents

Type	Params	Values
GET	-----	-----

Response

Status	Response
200	List of DocumentResource
401	{"error":"Unauthorized."}
404	Not found

Promo code Master:

Promocode	<input type="text"/>
Discpercentage	<input type="text"/>
Maximum discount	<input type="text"/>
Start date	<input type="text"/>
End date	<input type="text"/>
Applicable times	<input type="text"/>
	<input type="button" value="Submit"/>

- Only OWNER, User can register a Promocode.

API call: rootURL/api/User/PromoCode

API documentation:

Request

Method	URL
POST	rootURL/api/User/PromoCode

Type	Params	Values
FromBody	PromoCodeResource <ul style="list-style-type: none">• PromocodeID	Long

	<ul style="list-style-type: none"> • Promocode • Discpercentage • MaxDisc • StartDate • EndDate • Cid • Enabled • ApplicableTimes 	String Double Double DateTime DateTime Long Bool Int
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------

Response

Status	Response
200	PromoCodeResource
401	{"error": "Unauthorized."}

Add customers:

Customers can be either created by owner or by sub-users or by customer himself.

Name	<input type="text"/>
Phone No	<input type="text"/>
Email ID	<input type="text"/>
House No	<input type="text"/>
Landmark	<input type="text"/>
Area/City/Pincode	<input type="text"/>
Gender	<input type="text"/>
Photo	<input type="text" value="Upload"/>
<input type="button" value="Submit"/>	

API call: rootURL/api/User/RegisterCustomer

API Documentation:

Request

Method	URL
POST	rootURL/api/User/RegisterCustomer

Type	Params	Values
POST	<div>UserResource<ul style="list-style-type: none">EmailPasswordNameGenderPhone numberDisplayPicture</div> <div>Organization<ul style="list-style-type: none">OrganizationNameGSTCINPANCompanyLogoPassPhraseCid</div> <div>Address<ul style="list-style-type: none">HouseNoLandmarkAreaCityCountryLatLngTagPinCode</div>	<div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>OrganizationResource</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>long</div> <div>AddressResource</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>string</div> <div>double</div> <div>double</div> <div>string</div> <div>string</div>

Response

Status	Response
200	UserResource
401	{"error": "Unauthorized."}
400	{"error": "Badrequest."}

Place orders:

Owner can also place orders for customers.

- Here est cost and est kms should be displayed once owner selects particular package from drop-down.

Customer Id (+)	<input type="text"/>	
Date	<input type="text"/>	
Booking time	<input type="text"/>	
Order type	Point-to-point/package ▾	
Source (+)	<input type="text"/>	Map view
Destn	<input type="text"/>	Map view
Occupancy	<input type="text"/>	
Promo code	<input type="text"/>	
Package	<input type="text"/> ▾	Est cost Est kms
<div>Place order</div>		

API call: rootURL/api/User/Order?PackageId=value&CustomerId=value

API call: rootURL/api/User/Packages (for package drop-down)

API call: rootURL/api/Distance/EstBilling (to display est cost & est kms)

API Documentation:

Request

Method	URL
POST	rootURL/api/User/Order?PackageId=value&CustomerId=value

Type	Params	Values
FromBody	OrderResource <ul style="list-style-type: none">OrderId	Long

	<ul style="list-style-type: none"> • Cid • Customer 	Long
	UserResource	UserResource
	1. Email	string
	2. Password	
	3. Name	string
	4. Phone number	string
	5. DisplayPicture	string
	Address	string
	1. HouseNo	AddressResource
	2. Landmark	string
	3. Area	string
	4. City	string
	5. Country	string
	6. Lat	string
	7. Lng	double
	8. Tag	double
	9. PinCode	string
	• Source	string
	LocationInfo	LocationInfo
	1. Location	
	2. Lat	String
	3. Lng	Double
	• Destination	double
	LocationInfo	LocationInfo
	1. Location	
	2. Lat	String
	3. Lng	Double
	• OrderType	Double
	• EstOccupancy	String
	• Comments	Int
	• SchedulePickup	String
	• EstKms	DateTime
	• ScheduledDrop	Double
	• Package	DateTime
	PackageResource	PackageResource
	1. Packageld	Long
	2. PackageType	String
	3. PAckageName	String
	4. EstCost	Double
	5. EstKm	Double
	6. EstTime	Long
	7. EstModel	
	8. ExtCost	String
	9. ExtKm	Double
	10. ExtTime	Double

	11. Description	Long String
FromQuery	Packageld	Long
FromQuery	CustomerId	Long

Response

Status	Response
200	OrderResource
401	{"error":"Unauthorized."}

Request

Method	URL
GET	rootURL/api/User/Packages

Response

Status	Response
200	List of PackageResource
401	{"error":"Unauthorized."}

- Here List of Source should be Source lat&lng and Destination lat&lng

Request

Method	URL
POST	rootURL/api/Distance/EstBilling

Type	Params	Values
FromBody	List of Source <ul style="list-style-type: none"> • Lat • Lng 	Double Double
FromQuery	Packageld	Long

FromQuery	PromoCode	String
-----------	-----------	--------

Response

Status	Response
200	EstBillResource (EstKms,EstTimes,EstCost)
401	{"error":"Unauthorized."}

View orders:

Here vendor/owner can view all the orders that are raised by customers.

Start date/Time

End date/Time

API call: rootURL/api/Trip/ManageOrders?StartTime=value&EndTime=value

API Documentation:

Request

Method	URL
GET	rootURL/api/Trip/ManageOrders?StartTime=value&EndTime=value

Type	Params	Values
FromQuery	StartTime	DateTime
FromQuery	EndTime	DateTime

Response

Status	Response
200	List of OrderResource
401	{"error":"Unauthorized."}

Allocate vehicles:

Here owner will allocate vehicles for received orders.

Date

Start date

End date

View

List of trips:

Order id	Allocate
	▽

API call: rootURL/api/Trip/ManageOrders?StartTime=value&&EndTime=value

API call:rootURL/api/Driver/OnlineDrivers

API call:rootURL/api/Trip/Allocate?OrderId=value

API Documentation:

Request

Method	URL
GET	rootURL/api/Trip/ManageOrders?StartTime=value&&EndTime=value

Type	Params	Values
GET	StartTime	DateTime
GET	EndTime	DateTime

Response

Status	Response
200	List of Orders
401	{"error":"Unauthorized."}

Request

Method	URL
GET	rootURL/api/Driver/OnlineDrivers

Response

Status	Response
200	List of DriverResource
401	{"error":"Unauthorized."}

Request

Method	URL
POST	rootURL/api/Trip/Allocate?OrderId=value

Type	Params	Values
FromQuery	DriverResource <ul style="list-style-type: none"> • DriverId • FirstName • LastName • Cid • DriverLicenseNo • DriverLicenseDoc • NotificationId 	Long String String Long String String String

Response

Status	Response
200	DriverResource
401	{"error":"Unauthorized."}

Trip dashboard:


In trip dashboard last four hours completed trips, running trips and next 4 hours trips will be displayed based on logged in user Cid.

53
Started

16
Delayed

15
Not started


20
Ended

	Trip ID	Est start time	Status	Occ	Vehicle	Act Start	ActEnd	Loc	Comment	Pickup Area
	2624	06:45	Started	1	KL22H5522 8289943397	06:57	07:30	MLKP		Koti

S.No	Type	Customer	Area	Est.Time	Act time	Status
1	Pickup	9995080110 Maya K S	Koti	07:00	07:05	On time
2	Drop	9995080110 Maya K S	Madhapur	09:00		

Map view

Vech no Name Phnno Speed Location



API Documentation:

The user with roles like OWNER, USER can be authenticated.

Request

Method	URL
GET	rootURL/api/Trip/TripDashBoard

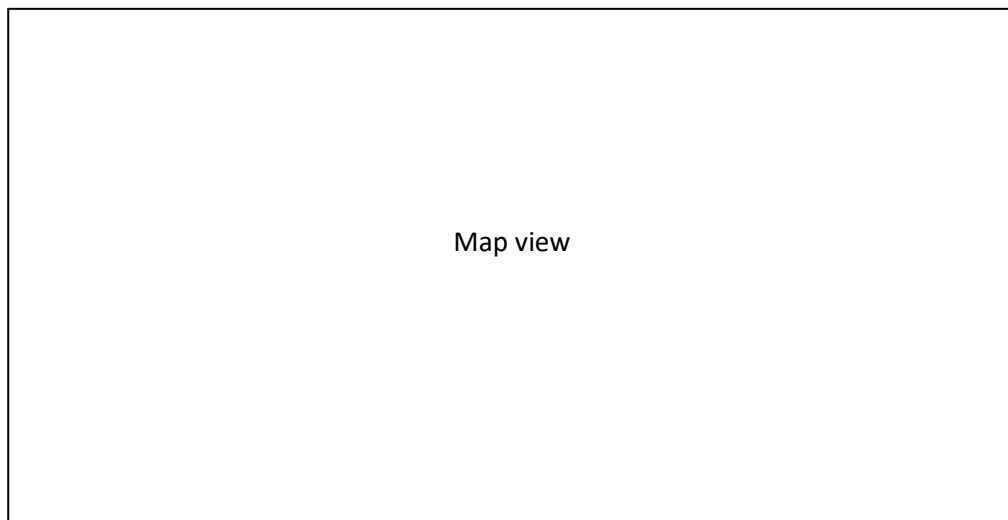
Type	Params	Values
GET	-----	-----

Response

Status	Response
200	List of TripResource
401	{"error":"Unauthorized."}

View maps:

Here owner can track vehicles in map view.



API call: rootURL/api/Master/Map

API Documentation:

Request

Method	URL
GET	rootURL/api/Master/Map

Response

Status	Response
200	List of GPSCurrent
401	{"error":"Unauthorized."}

Billing:

Here Owner, User can create packages. Billing page will be accessed by certain roles that are given by the owner.

PackageType	<input type="text"/>
Package Name	<input type="text"/>
Est.cost	<input type="text"/>
Est Km	<input type="text"/>
Est time	<input type="text"/>
Est model	<input type="text"/>
Ext cost	<input type="text"/>
Ext Km	<input type="text"/>
Ext time	<input type="text"/>
Description	<input type="text"/>
	<input type="button" value="Submit"/>

API call: rootURL/api/Master/Package

API Documentation:

Request

Method	URL
POST	rootURL/api/Master/Package

Type	Params	Values
FromBody	PackageResource <ul style="list-style-type: none">PackageIDPackageTypePackageNameEstcostEstKmEsttimeEstModelExtcostExtKmExtTimeDescription	Long String String Double Double Long String Double Double Long String

Response

Status	Response
200	PackageResource
400	{"error": "Bad request."}

- In this page vendor will also get billing report by giving start date and end date.
- This will be implemented in next phase.

Change Password:

New password	<input type="password"/>
Confirm new password	<input type="password"/>
<input type="button" value="Save"/>	

- Owner can change password by selecting “change password” option from Menu list.

API call: rootURL/api/User/UserId?Password=value

API Documentation:

Request

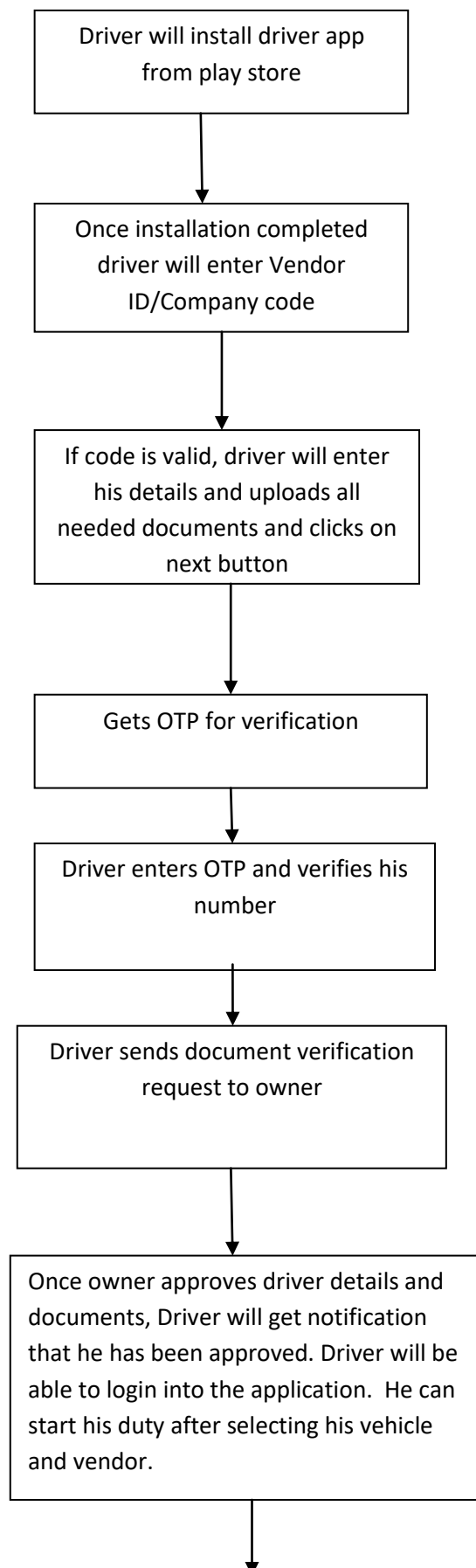
Method	URL
PUT	rootURL/api/User/UserId?Password=value

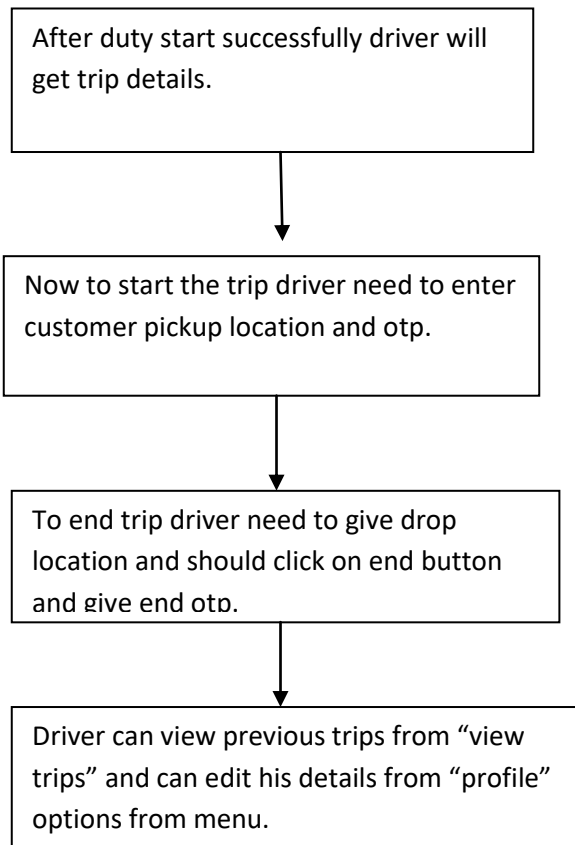
Type	Params	Values
FromRoute FromQuery	Userid Password	Long string

Response

Status	Response
200	PackageResource
400	{"error": "Bad request."}

Driver app workflow:





Driver App:

- Driver install the mobile app from given apk file/ from app store.
- Driver gets unique code from the owner to login through SMS/ email. Here code will be unique from vendor to vendor.
- Driver enters given code.
- If GPS location is off driver cannot enter into the mobile application.

Code

API call: rootURL/api/Driver/SearchCode?Code=value

API Documentation:

Response

Method	URL
GET	rootURL/api/Driver/SearchCode?Code=value

Type	Params	Values
FromQuery	Code	String

Response

Status	Response
200	OrganizationResource
400	{"error": "Bad request."}

- Once driver clicks on next,if code is valid, driver will enter his details and uploads all needed documents and clicks on next button.

First Name

Last Name

Phone no

Cid

Driver License no

Driver License doc

Upload

Photo

Upload

Register

API call: rootURL/api/Driver/

API Documentation:

Request

Method	URL
POST	rootURL/api/Driver/Request

Type	Params	Values
POST	DriverResource <ul style="list-style-type: none">• DriverId• FirstName• LastName• Cid• DriverLicenseNo• DriverLicenseDoc• PhoneNo• Notification	Long String String Long String String String string

Response

Status	Response
200	Pin(string)
400	{"error": "Bad request"}

- After driver clicks on register button he will get a screen where he needs to enter his registered phone no with received OTP.

Phone No

OTP

Send

API call: rootURL/api/Driver/Validate?Otp=value&PhoneNo=value

API Documentation:

Request

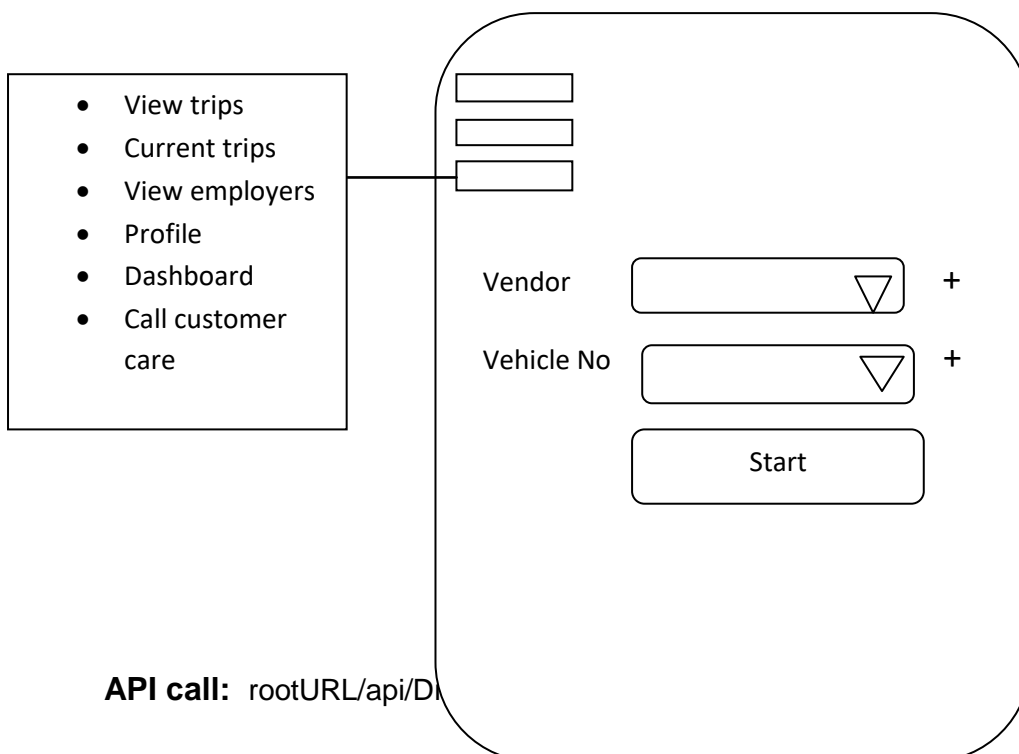
Method	URL
GET	rootURL/api/Driver/Validate?Otp=value&PhoneNo=value

Type	Params	Values
FromQuery	PhoneNo	String

Response

Status	Response
200	OTP(string)
401	{"error": "Unauthorized"}

- Once authentication process is completed by entering OTP an approval request will be sent to owner.
- Once owner approves the details and uploaded documents, driver can login into the mobile application where he get duty start page with menu bar.



API call:rootURL/api/Driver/ApprovedVehicles?cid=value

API call:rootURL/api/Driver/Duty?ActionType=Start&&NotificationId=value

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/ApprovedVendors

Response

Status	Response
200	List of OrganizationResource
401	{"error":"Unauthorized"}

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/ApprovedVehicles?cid=value

Type	Params	Values
FromQuery	Cid	Long

Response

Status	Response
200	List of VehicleResource
401	{"error":"Unauthorized"}

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/Duty?ActionType=Start&&NotificationId=value

Type	Params	Values
FromBody	VehicleResource <ul style="list-style-type: none"> RegNo Manufacturer VehicleTypeResource <ol style="list-style-type: none"> VTypeId ManufacturerName Name Occupance Icon ManufacturerIcon <ul style="list-style-type: none"> Color City ModelYear Photo Contract ContractResource <ol style="list-style-type: none"> ContractType StartContract EndContract <ul style="list-style-type: none"> ShortName 	String VehicleTypeResource Long String String Int String String String String String String Long String ContractResource String DateTime DateTime String
FromRoute	DriverId	Long
FromQuery	ContractType	String
FromQuery	EndContract	DateTime

Response

Status	Response
200	Boolean(True/False)
401	{"error":"Unauthorized"}

- Once driver clicks on start button he will get trip to start with duty end button.

Trip details

Duty end>>>

Trip Id:

Pax:

Pickup:

Escort:

>>>>>>

Centre:

EST start:

>>>>>>

API call: rootURL/api/Driver/AssignedTrip

API call: rootURL/api/Driver/Duty?ActionType=End&&NotificationId=value

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/AssignedTrip

Response

Status	Response
200	TripResource
401	{"error":"Unauthorized"}

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/Duty?ActionType=End&&NotificationId=value

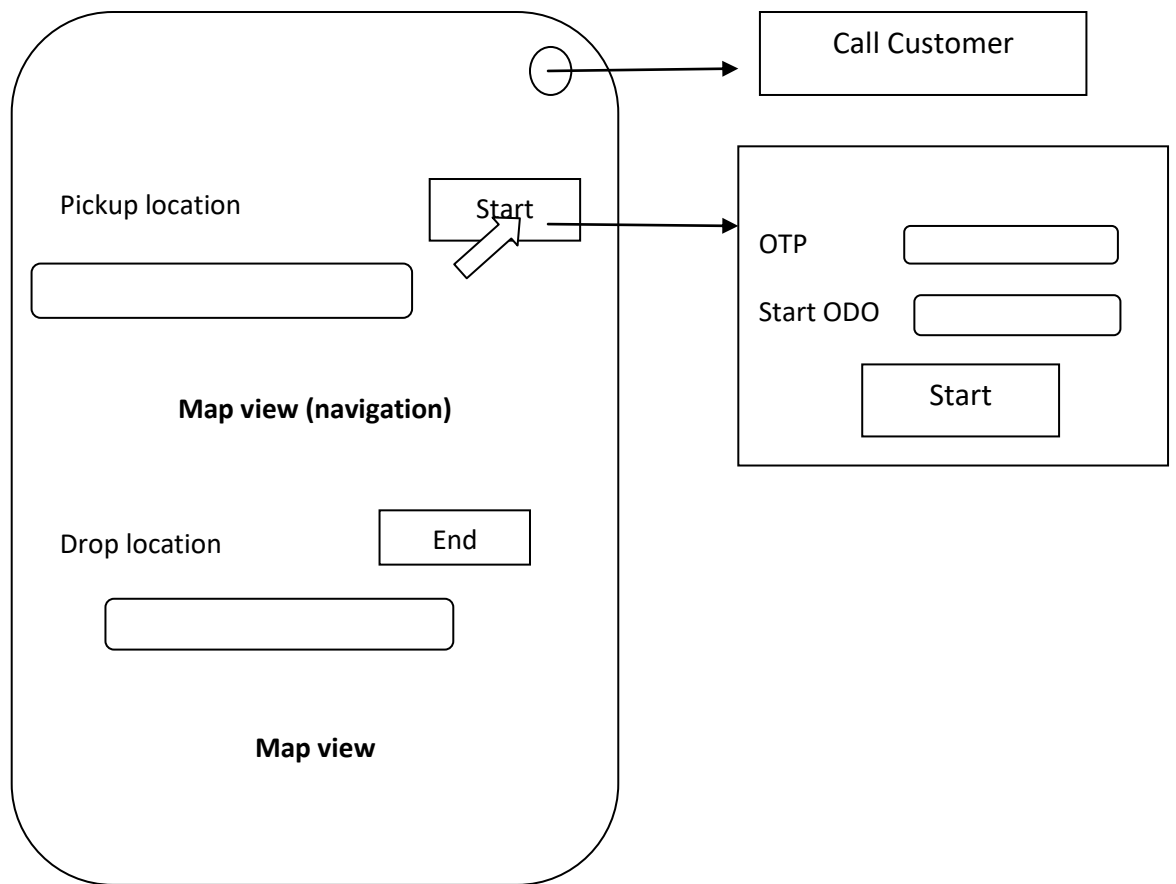
Type	Params	Values
------	--------	--------

FromBody	VehicleResource <ul style="list-style-type: none"> • RegNo • Manufacturer VehicleTypeResource <ul style="list-style-type: none"> 7. VTypeld 8. ManufacturerName 9. Name 10. Occupance 11. Icon 12. ManufacturerIcon • Color • City • ModelYear • Photo • Contract ContractResource <ul style="list-style-type: none"> 4. ContractType 5. StartContract 6. EndContract • ShortName 	String VehicleTypeResource Long String String Int String String String String Long String ContractResource String DateTime DateTime String
FromRoute	DriverId	Long
FromQuery	ContractType	String
FromQuery	EndContract	DateTime

Response

Status	Response
200	Boolean(True/False)
401	{"error": "Unauthorized"}

- Once driver clicks on trip, trip gets started and he will get a screen where driver need to enter source destination and end destination.
- Here Trip id will be hard coded for trip started/ended.
- Once driver start/end trip he will get a pop-up where he need to enter OTP which will be provided by customer, start ODO(when he starts trip) and OTP(provided by customer), end ODO(when he ends trip)
- Once driver end trip a new screen will bill details will be displayed.
- Once trips are completed driver will click on duty end button for that day.



API call: rootURL/api/Driver/Start?OTP=value&Tripid=value&StartODO=value

API call:rootURL/api/Driver/End?OTP=value&Tripid=value&StartODO=value

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/Start?OTP=value&Tripid=value&StartODO=value

Response

Status	Response
200	Boolean (True/False)
401	{"error":"Unauthorized"}

Request

Method	URL
GET	rootURL/api/Driver/End?OTP=value&TripId=value&StartODO=value

Response

Status	Response
200	FinalBillResource (ApprovedKms,TravelledTime,Cost,Discount,FinalCost)
400	{"error":"Bad request"}
401	{"error":"Unauthorized"}

Travelled Time:

Kms:

Cost:

Discount:

Total amount:

- Driver can add new vendor by clicking on “+” button and entering the code given by the new vendor.
- If the entered code is valid driver will get a screen where he needs to upload his documents.

Document

Upload

Validity

Submit

API call: rootURL/api/Driver/VendorRequest?VendorId=value

API Documentation:

Request

Method	URL
POST	rootURL/api/Driver/VendorRequest?VendorId=value

Type	Params	Values
FromBody	UserId UserType DocumentName DocURL AlertDate DocNum Validity	Long String String String DateTime String DateTime

Response

Status	Response
200	Boolean
400	{"error":"Bad request"}
401	{"error":"Unauthorized"}

- Once driver clicks on submit button an approval request will be sent to vendor.
- If driver wants to change the contact number then he should change for whole profile but not for particular vendor.
- Driver can add new vehicle by clicking on “+” button and enter the details.

Manufacturer

Reg no

Colour

City

Model year

Occupancy

Contract

start date

End date

Short name

Save

API call: rootURL/api/Driver/Vehicle

API call: rootURL/api/Master/VehicleType (for manufacturer drop-down)

API Documentation:

Request

Method	URL
POST	rootURL/api/Driver/Vehicle

Type	Params	Values
FromBody	VehicleResource <ul style="list-style-type: none"> RegNo Manufacturer VehicleTypeResource <ul style="list-style-type: none"> 13. VTypeId 14. ManufacturerName 15. Name 16. Occupance 17. Icon 18. ManufacturerIcon Color City 	String VehicleTypeResource Long String String Int String String String String

	<ul style="list-style-type: none"> ModelYear Photo Contract ContractResource	Long String ContractResource
	7. ContractType 8. StartContract 9. EndContract	String DateTime DateTime
	<ul style="list-style-type: none"> ShortName 	String
FromRoute	DriverId	Long
FromQuery	ContractType	String
FromQuery	EndContract	DateTime

Response

Status	Response
200	VehicleResource
401	{"error":"Unauthorized"}

API Documentation:

Request

Method	URL
GET	rootURL/api/Master/VehicleType

Response

Status	Response
200	List of VehicleTypeResource
401	{"error":"Unauthorized"}

- Driver can view his total trips, total kms travelled, total amount earned till present.

Total number of trips	11
Total kilometres travelled	120
Total amount earned	Rs. 5000/-

API call: rootURL/api/Driver/DashBoard

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/DashBoard

Response

Status	Response
200	List of DriverDashboard(DriverId,OrganizationName,TotalTrips,TotalKms,TotalCo)
401	{"error":"Unauthorized"}

- Driver can view trips by simply entering datefrom which date to which date he wants to view the trips.

Date

View

API call: rootURL/api/Driver/Trips?StartDate=value

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/Trips?StartDate=value

Response

Status	Response
200	List of TripResource
401	{"error": "Unauthorized"}

- Once driver clicks on view button he will get the list of trips for particular date interval.

Date

Total Kms:

Trip ID

Pickup time:

Location:

- Driver can view employers' details once he clicks on "view employers".

List of employers

Vendor name:

Vendor id:

API call: rootURL/api/Driver/ApprovedVendors

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/ApprovedVendors

Response

Status	Response
200	List of OrganizationResource
401	{"error": "Unauthorized"}

- Driver can edit/view his profile by selecting profile option from the menu. Here driver can edit only his email id and profile picture.

Driver id
First name
Last name
Cid
Driver License no
Driver License doc
Phone no
Email id
Photo

API call: rootURL/api/Driver/Update

API Documentation:

Request

Method	URL
PUT	rootURL/api/Driver/Update

Type	Params	Values
FromBody	DriverResource <ul style="list-style-type: none">• DriverId• FirstName• LastName• Cid• DriverLicenseNo• DriverLicenseDoc• PhoneNo• NotificationID	Long String String Long String String String String

Response

Status	Response
200	DriverResource
401	{"error": "Unauthorized"}

- Whenever there is a call for OTP in driver app below OTP API call should takes place.

API call: rootURL/api/Driver/Otp?PhoneNo=value

API Documentation:

Request

Method	URL
GET	rootURL/api/Driver/Otp?PhoneNo=value


Type	Params	Values
FromQuery	PhoneNo	String

Response

Status	Response
200	String
401	{"error": "Unauthorized"}

- Once trip is completed driver need to give feedback regarding trip.

Feedback



Remarks

submit

API call: root/URL/api/Driver/Feedback?TripId=value

API documentation:

Request

Method	URL
POST	root/URL/api/Driver/Feedback?TripId=value

Type	Params	Values
FromBody	CheckListResource <ul style="list-style-type: none">TripIdCommentsRatingTagUserId	Long String Int String Long

Response

Status	Response
200	Boolean
401	{"error":"Unauthorized."}

- Location API need to be called for every 30 seconds.

API call: root/URL/api/Locations?Locations=value

API documentation:

Request

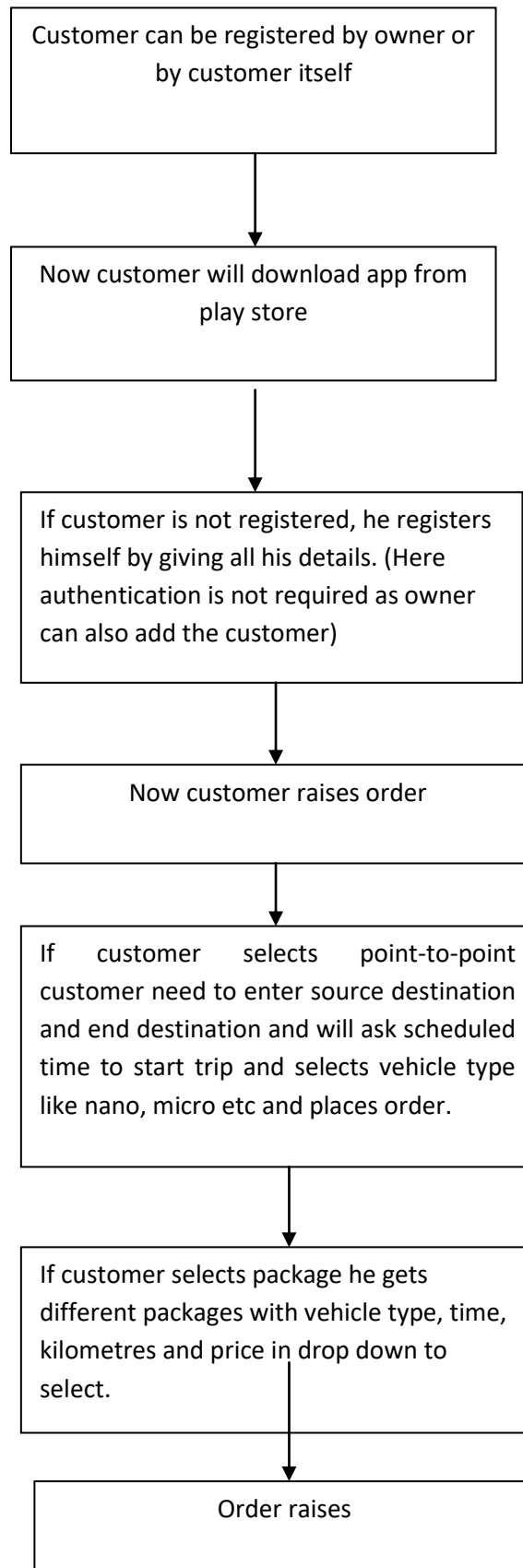
Method	URL
GET	root/URL/api/Locations?Locations=value

Type	Params	Values
FromQuery	Locations	String

Response

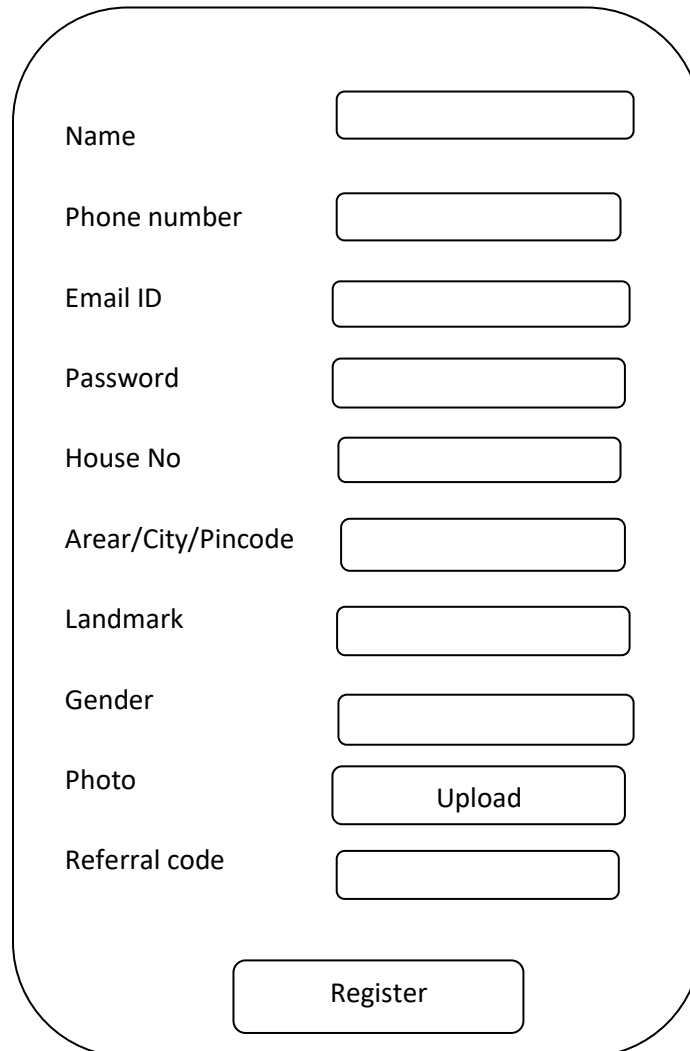
Status	Response
200	Boolean

Customer app workflow:



Customer APP:

- Customer can be registered by himself or by owner.
- Customer need to download app from play store.
- Once app download is completed customer will get registration page where he needs to enter details.



A registration form UI mockup with a rounded rectangular border. It contains the following fields and buttons:

- Name:
- Phone number:
- Email ID:
- Password:
- House No:
- Arear/City/Pincode:
- Landmark:
- Gender:
- Photo:
- Referral code:
-

API call: rootURL/api/Customer/Register/Cid?ReferredBy=value&&ReferralCode=value

API Documentation:

Request

Method	URL
POST	rootURL/api/Customer/Register/Cid?ReferredBy=value&&ReferralCode=value

API call: rootURL/api/Auth/CustomerLogin

returns: jwt token

- If sign in fails should display pop-up as “enter valid credentials”.

API Documentation:

Authenticate the user with the system and obtain the jwt_token.

- Here token will be in base64.
- If the token is decoded, then we get list of roles.

Request

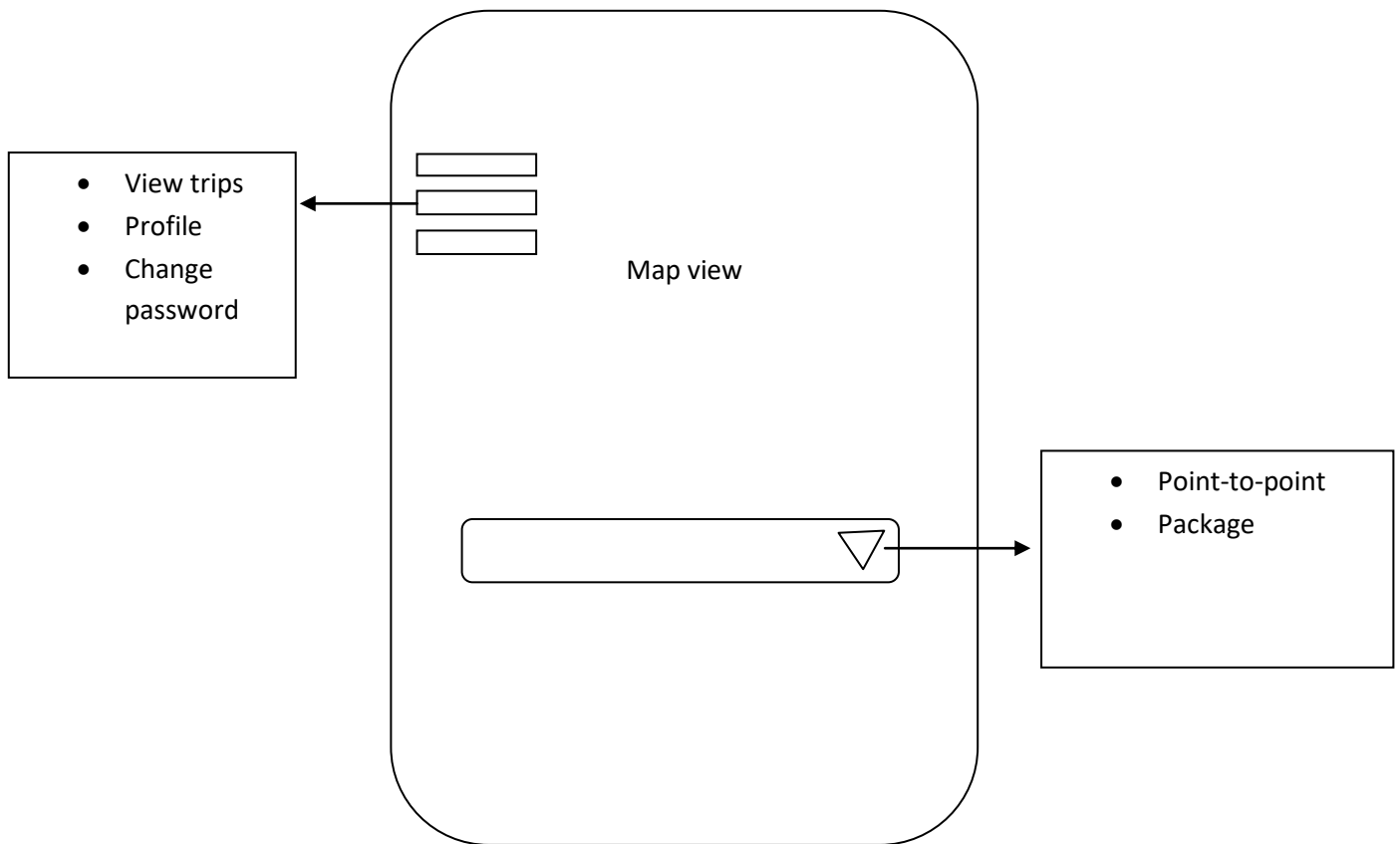
Method	URL
POST	rootURL/api/Auth/CustomerLogin

Type	Params	Values
POST	LoginModel Username Password	string string

Response

Status	Response
200	<pre>{ "auth_key": <auth_key> }</pre> <p>auth_key(string) - all further API calls must have this key in header</p>
401	<pre>{"error": "Enter valid credentials."}</pre>

- Once customer login into the application a map view with order request page will be displayed.
- In this page customer will get two options to book cab.
 - 1) Point-to-Point
 - 2) Package



API call: rootURL/api/User/Packages

API Documentation:

Request

Method	URL
POST	rootURL/api/User/Packages

Response

Status	Response
200	List of PackageResource
401	{"error":"Unauthorized"}

- If customer selects point-to-point option from drop down he will get a screen where he need to select type of vehicle by entering source and destination points.
- Once customer selects vehicle app will automatically directs to next screen where customer will get journey prices with different kilometres and time limits.
- Once customer selects required package he will pace order.

Map view

Source location

Drop location

MiniSedanSUV

Map view

Est timeEst rate

Promo code

Place order

API call: rootURL/api/Trip/Order?PackageId=value

API Documentation:

Request

Method	URL
POST	rootURL/api/Trip/Order?PackageId=value

Type	Params	Values
FromBody	OrderResource <ul style="list-style-type: none"> • Cid • OrderType • EstOccupancy • Comments • SchedulePickup • ScheduleDrop • EstKms • PromoCode • Customer <ol style="list-style-type: none"> 1. Email 2. Password 3. Name 4. PhoneNo 5. DisplayPicture 6. Address <ol style="list-style-type: none"> I. HouseNo II. Landmark III. Area IV. City V. Country VI. Lat VII. Lng VIII. Tag IX. Pincode • Source <ol style="list-style-type: none"> I. Location II. Lat III. Lng • Destination <ol style="list-style-type: none"> I. Location II. Lat III. Lng 	Long String Int String DateTime DateTime Double String String String String String String String String String String double double String String String Double Double String Double Double

	<ul style="list-style-type: none"> Package <ol style="list-style-type: none"> I. PackageId Long II. PackageType String III. PackageName String IV. EstCost Double V. EstKm Double VI. EstTime Long VII. EstModel String VIII. ExtCost Double IX. ExtKm Double X. ExtTime Long XI. Description String 	
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

Response

Status	Response
200	OrderResource
401	{"error": "Unauthorized."}

- If customer selects package option, he will get a screen where he needs to enter source and destination locations and he will get different packages options in drop down to select.

Source location

Map view

Drop location

Map view

Indica (4 seater) 4hrs 40kms

Place order

Est cost 1500/-

Extra cost Rs.10/km

API call: rootURL/api/Trip/Order?PackageId=value

API Documentation:

Request

Method	URL
POST	rootURL/api/Trip/Order?PackageId=value

Type	Params	Values
FromBody	OrderResource <ul style="list-style-type: none"> • Cid • OrderType • EstOccupancy • Comments • SchedulePickup • ScheduleDrop • EstKms • PromoCode • Customer <ol style="list-style-type: none"> 1. Email 2. Password 3. Name 4. PhoneNo 5. DisplayPicture 6. Address <ol style="list-style-type: none"> I. HouseNo II. Landmark III. Area IV. City V. Country VI. Lat VII. Lng VIII. Tag IX. Pincode • Source <ol style="list-style-type: none"> I. Location II. Lat III. Lng • Destination <ol style="list-style-type: none"> I. Location 	Long String Int String DateTime DateTime Double String String String String String String String String String String String double double String String String Double Double String Double

	II. Lat III. Lng • Package I. Packageld II. PackageType III. PackageName IV. EstCost V. EstKm VI. EstTime VII. EstModel VIII. ExtCost IX. ExtKm X. ExtTime XI. Description	Double Long String String Double Double Long String Double Double Long String
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------

Response

Status	Response
200	OrderResource
401	{"error": "Unauthorized."}

- Both point-to-point and package will be calling same API.
- Once customer place orders an order request will be sent to owner and he will allocate vehicle and driver to that trip.
- Customer can edit his personal details by selecting profile option from the menu.
- Customers can view trips by selecting “view trips” option from menu.
- Initially Last four trips will be viewed.

Day, Date, time
Vehicle type
Pickup location
Drop location

Day, Date, time
Vehicle type
Pickup location
Drop location

Day, Date, time
Vehicle type

API call: rootURL/api/Customer/PastTrips?StartDate=value&&EndDate=value

API Documentation:

Request


Method	URL
POST	rootURL/api/Customer/PastTrips?StartDate=value&&EndDate=value

Response

Status	Response
200	List of TripResource
401	{"error": "Enter valid credentials."}

- Once trip is completed customer need to give feedback for driver.

Feedback



Remarks

API call: root/URL/api/Customer/Feedback?TripId=value

API documentation:

Request

Method	URL
POST	root/URL/api/Customer/Feedback?TripId=value

Type	Params	Values
FromBody	CheckListResource <ul style="list-style-type: none">• TripId• Comments• Rating• Tag• UserId	Long String Int String Long

Response

Status	Response
200	Boolean
401	{"error": "Unauthorized."}

- Customer selects “change password” from menu list if he wants to change password.

New password

Confirm password

Save

API call:root/URL/api/Customer/Update?Password=value

API documentation:

Request

Method	URL
PUT	root/URL/api/Customer/Update?Password=value

Type	Params	Values
FromQuery	Password	String

Response

Status	Response
200	True
401	{"error":"Unauthorized."}