

**LNext Cabs!**

**Release 0.0**

**Purpose**

LNext cabs allow customers to automatically book the nearest available cab. In the release 0.0 we are providing a simple functionality where the user will enter her/ his name along with location details. The nearby driver will be auto assigned.

**Database Details**

* URL: http://www.phpmyadmin.co/
* Server: sql2.freemysqlhosting.net
* Name: sql12350415
* Username: sql12350415
* Password: jRAaefXBXm

**Table Structure**

1). Driver table will contain the driver information (driver name, current latitude, current longitude and current status). Initially the status will be 'AVAILABLE", once the cab is assigned status will be updated to 'BUSY'.

*create table driver\_details (did bigint not null, driver\_name varchar(255), latitude double precision, longitude double precision, status varchar(255), primary key (did))*

2). Order/Booking Table will contain the order number, customer name, customer latitude, longitude, driver id and the time at which booking is initiated.

*create table booking\_details (order\_number integer not null, booking\_date TIMESTAMP DEFAULT CURRENT\_TIMESTAMP, customer\_latitude double precision, customer\_longitude double precision, customer\_name varchar(255), driver\_did bigint, primary key (order\_number))*

alter table booking\_details add constraint FK6n1luvxu5j9cqmoidfbx0u3pv foreign key (driver\_did) references driver\_details (did)

**Insert Queries**

For the testing purpose, five drives are added in the database.

*#Mahesh, ICICI Bank*

*INSERT INTO `driver\_details` (`did`, `driver\_name`, `latitude`, `longitude`, `status`) VALUES ('0', 'Mahesh', '19.1204671', '72.8869815', 'AVAILABLE');*

*#Nandu, Mangala Hospital*

*INSERT INTO `driver\_details` (`did`, `driver\_name`, `latitude`, `longitude`, `status`) VALUES ('1', 'Nandu', '19.1226761', '72.8869962', 'AVAILABLE');*

*#Ramesh, Seven Hills*

*INSERT INTO `driver\_details` (`did`, `driver\_name`, `latitude`, `longitude`, `status`) VALUES ('2', 'Ramesh', '19.1175964', '72.8770851', 'AVAILABLE');*

*#Suresh, JB Nagar Metro station*

*INSERT INTO `driver\_details` (`did`, `driver\_name`, `latitude`, `longitude`, `status`) VALUES ('3', 'Suresh', '19.1204671', '72.8869815', 'AVAILABLE');*

*#Dinesh, Ganesh Nagar*

*INSERT INTO `driver\_details` (`did`, `driver\_name`, `latitude`, `longitude`, `status`) VALUES ('4', 'Dinesh', '19.110580', '72.886738', 'AVAILABLE');*

**Test Customers**

Below customers can be used for booking the cab.

1). Himanshu Shukla: Lok Sarita:19.1205232, 72.8823594, nearest driver is Mahesh.

2). Divyang Bhimani: Peninsula Grand Hotel, Saki Naka:19.103938, 72.888382, nearest driver is Dinesh.

**Query to fetch nearest driver**

Used Vincenty formula is used to find-out the nearest driver.

**Reference**: https://www.plumislandmedia.net/mysql/vicenty-great-circle-distance-formula/

Here is the query to fetch the nearby driver for Divyang:

*SELECT \*, (ATAN2(SQRT(*

*POW(COS(RADIANS(d.latitude))\*SIN(RADIANS(d.longitude-72.888382)),2) +*

*POW(COS(RADIANS(19.103938))\*SIN(RADIANS(d.latitude)) -*

*(SIN(RADIANS(19.103938))\*COS(RADIANS(d.latitude)) \**

*COS(RADIANS(d.longitude-72.888382))) ,2)),*

*SIN(RADIANS(19.103938))\*SIN(RADIANS(d.latitude)) +*

*COS(RADIANS(19.103938))\*COS(RADIANS(d.latitude))\*COS(RADIANS(d.longitude-72.888382)))) as distance*

*from driver\_details d HAVING d.status='AVAILABLE' ORDER BY distance LIMIT 1;*

**Query to update driver status**

As of now we are not changing the driver status from BUSY to AVAILABLE, in the future or during testing we might need to do it. Here is the query to update the status:

*update driver\_details set status='AVAILABLE' where did=?;*

**Web-Services Details**

As of now four web-services are added.

Here is the swagger URL: http://localhost:8082/swagger-ui.html#/

1). *getDriverName*

**Description**: This web-service will return the nearest driver name.

**URL**: http://localhost:8082/getDriverName

**Input**: {"customerLatitude":19.103938,"customerLongitude":72.888382,"customerName":"Divyang Bhimani"}

**Output**: {"applicationCode":"200","applicationResponse":"Success","driverName":"Dinesh","customerName":"Divyang Bhimani"}

2). *bookcab*

**Description**: This web-service takes Customer Name, Customer Latitude, Customer Longitude. It first fetchs the nearest available driver from 'driver\_details' table. Insert the details in 'booking\_details' table and then update the driver status to BUSY in 'driver\_details'. And return the name of assigned driver.

**URL**: http://localhost:8082/bookcab

**Input**: {"customerLatitude":19.103938,"customerLongitude":72.888382,"customerName":"Divyang Bhimani"}

**Output**: {"applicationCode":"200","applicationResponse":"Success","driverName":"Dinesh","customerName":"Divyang Bhimani"}

3). *getAllBookings*

**Description**: This web-service will return all the bookings till date.

**URL**: http://localhost:8082/getAllBookings

**Input**: NA

**Output**: {"applicationCode":"200","applicationResponse":"Success","bookingDetails":[{"orderNumber":1,"customerName":"Divyang Bhimani","customerLatitude":19.103938,"customerLongitude":72.888382,"driver":{"did":4,"driverName":"Dinesh","latitude":19.11058,"longitude":72.886738,"status":"BUSY"},"bookingDate":"2020-06-24T11:26:25.000+00:00"}]}

4). *getAllDrivers*

**Description**: This web-service will return a list of all the drivers along with the customer name (if the cab is booked)

**URL**: http://localhost:8082/getAllDrivers

**Input**: NA

**Output**: {"applicationCode":"200","applicationResponse":"Success","drivers":[{"driverName":"Mahesh","status":"AVAILABLE","customerName":null},{"driverName":"Nandu","status":"AVAILABLE","customerName":null},{"driverName":"Ramesh","status":"AVAILABLE","customerName":null},{"driverName":"Suresh","status":"AVAILABLE","customerName":null},{"driverName":"Dinesh","status":"BUSY","customerName":"Divyang Bhimani"}]}

**Future Enhancements**

Here is a list of couple of enhancements which we can incorporate in a future release

* Field validations need to be added.
* API's to add and delete the driver.
* API to update the live location of Driver.
* Once the ride is finished Driver status should be changed to AVAILABLE.
* Option for a driver to refuse the ride. They must be allowed to take a break, for that we need to expose a web-service which will allow them to update their status.
* Payment gateway.
* Better locking mechanism while doing the booking.