Ride Sharing System

Kartikey Dubey[1], Nikhil Kumar[2], Harsh Chaudhary[3], Nishant Singh[4]
[1,2,3,4]*Rajkumar Goel Institute of Technology, AKTU*

Abstract - **We define a method of Ride Sharing System consisting of a web portal from which one easily book and share the rides or vehicle in an efficient manner. To reduce the ill effect of the private vehicle this technology is very necessary now a days. In this there are new services and facilities by which the effect on the environment like pollution, congestion etc. can be reduced and to provide support to the needy at the earliest. Mass transit system is the best solution if provided efficiently, but many persons do not prefer it because of its lack of door to door service, longer and fixed route and less reliable schedule Ride sharing is one of the emerging technologies adopted all over the world, in which users with same origin destination and time of travel and they share the ride.**

So in this project there is a responsive website which uses the technology of front end and back end along with database management system. And to provide convenience to the users. Machine learning algorithm are also implied by which one can share or book ride in an efficient manner. Concept of data base management system like query optimization, joins are been used to save the data efficiently in the data base. The relational data base MYSQL is been used to maintain server side.

THE IDEA BEHIND IT

Ride Sharing is the popular mode of transportation which aims to increase vehicle usage with similar itineraries and time schedules. They do a very justified approximation. It is a service that arranges one-time shared rides on very short notice. This type of technology generally makes use of three technological: GPS navigation devices to determine a driver's, Smartphones for a traveler to request a ride , Social networks to establish trust and accountability between drivers and passengers. Real-time Ride Sharing promotes a better way to utilize the empty seats in most passenger cars, thus lowering fuel usage and transport costs. Ridesharing is also capable of serving one-time trips. Real-time ridesharing is especially suitable for daily commuting compared to driving alone. As it reduces air pollution which now a days is a major concern. Early Ride sharing idea comes in late 1990's but at that time due to lack of advance technologies like smartphones Gps this idea fails, Gradually this idea comes with the advancement shifted from telephone to internet, email, and smartphone.

THE PLANS

Ride-sharing frameworks should consolidate natural assurance (through a decrease of CO2, NO2 like exceptionally dirtied gases), socialization, and security. Urging individuals to utilize ridesharing frameworks by fulfilling their requests for wellbeing, security and comfort is testing.

With the progression of portable interpersonal interaction advancements, ride sharing framework turns out to be simple approach to make correspondence among traveler and the drivers in a most ideal way. It bring about inventive transportation services. The accommodation of mentioning a ride from your telephone, imparting a ride to an associate ridesharing is the cutting edge of the sharing economy.

ARCHITECTURAL DESIGN

Real time ride sharing will be beneficial if the matching is done instantly and on demand. For that, a model will be prepared using software in which riders with approximately similar origin-destination and travel time will be selected from the database generated from user interface.

The proposed methodology includes data collection regarding existing traffic conditions and inventory of areal conditions, providing a GIS technology-based ride sharing application. Inventory data like population, land use, vehicular composition as well as road network will be collected in study area. The proposed methodology involves following tasks:

1. Review of earlier study reports, existing and proposed development plans.
2. Site inspection and collection of data by carrying out surveys.

ALGORITHM

Given a graph G = (V, E), a matching M is a set of edges with the property that no two of the edges have an endpoint in common. We say that a vertex v V is matched if v is incident to an edge in the matching. Otherwise, the vertex is unmatched. A matching is maximum if there is no matching of greater cardinality. In particular, a maximum matching is called perfect if every vertex of G is matched.

```
The Matching Algorithm
{
1.      Start with any matching.
2.      Find an augmenting path with respect to the current matching.
3.      Augment the current matching.
4.      Repeat the above two steps as long as possible.
}
```

TECHNOLOGY USED

1. For UI (HTML5, CSS3) For Designing. And to provide validations Java Script and JQuery is used.
2. To make it responsive (User Friendly) Bootstrap is used.
3. Relational Data Base (MYSQL).
4. Apache Tomcat Server to handle the backend.
5. Google's Location APIs, Google Maps Android API.
6. In Back End Process is done through PHP (Server-side Scripting Language).
7. Concept of data base like (Stored Procedure), Triggers are used to make it safe and secure from sql injection.

SYSTEM ANALYSIS

In this project we are using the technology of full Stack Web Development. We are fully exploiting the potential of these techniques. It is exciting to learn about how to create a responsive website and we think that this excitement is not unreasonable as this technology has shown its potential before as it is completely revolutionized computer vision.

Web portals has made its way into exciting compression methods. We have heard talks about method by which users can go and browse in the web portals to predict more decision to speed up the encodes. Their learned inter prediction method in loop filter and upscaling filters. But we are talking it to another level which is basically end- to-end. Even though the complexity may be high. We are interested in how the compression can be maximized even if we fully explore the potential of the techniques.

RIDESHARE DATABASE

A database is a set of tables, which contain fields. The tables and fields are related to one another in various logical ways. Entries can be stored in the tables as rows, with each row containing that entry's field data. The specific tables for each database will be determined during the implementation phase of the project. Each cylinder in the figure represents a set of tables, which in turn contain a set of fields, and a set of entries. We will utilize a MySQL database to create and maintain the rideshare database.

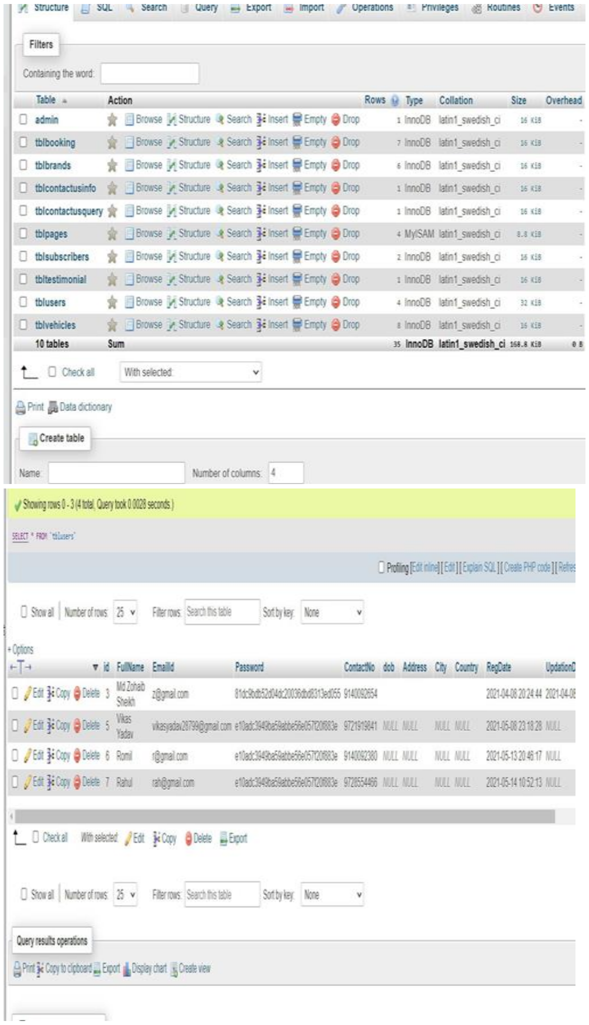


Table	Action	Rows	Type	Collation	Size	Overhead
admin	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16 KiB	-
tblbooking	Browse Structure Search Insert Empty Drop	7	InnoDB	latin1_swedish_ci	16 KiB	-
tblbrands	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16 KiB	-
tblcontactusinfo	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16 KiB	-
tblcontactusquery	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16 KiB	-
tblpages	Browse Structure Search Insert Empty Drop	4	MyISAM	latin1_swedish_ci	8.8 KiB	-
tblsubscribers	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
tbltestimonial	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16 KiB	-
tblusers	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	32 KiB	-
tblvehicles	Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	16 KiB	-
10 tables	Sum	35	InnoDB	latin1_swedish_ci	168.8 KiB	0 B

Showing rows 0 - 3 (4 total, Query took 0.0028 seconds.)

SELECT * FROM `tblusers`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refres
Show all
Number of rows: 25
Filter rows: Search this table
Sort by key: None
+ Options

	id	FullName	EmailId	Password	ContactNo	dob	Address	City	Country	RegDate	UpdationD
Edit Copy Delete	3	Md Zohaib Sheikh	z@gmail.com	81dc9bdb52d04dc20036dbd8313ed055	9140092654					2021-04-08 20:24:44	2021-04-0
Edit Copy Delete	5	Vikas Yadav	vikasyadav28799@gmail.com	e10adc3949ba59abbe56e057f20f883e	9721919841	NULL	NULL	NULL	NULL	2021-05-08 23:18:28	NULL
Edit Copy Delete	6	Romil	r@gmail.com	e10adc3949ba59abbe56e057f20f883e	9140092380	NULL	NULL	NULL	NULL	2021-05-13 20:46:17	NULL
Edit Copy Delete	7	Rahul	rah@gmail.com	e10adc3949ba59abbe56e057f20f883e	9728554466	NULL	NULL	NULL	NULL	2021-05-14 10:52:13	NULL

Database Table

METHODOLOGY

How to take a Rideshare and rider and ascertain whether they are compatible. Checking for age and gender preferences is simple enough. But schedule and route compatibility are a different story. It turns out that the first can be done if we apply a clever observation about the Driver"s worst case leave window, and the second is yet another instance of our good friend the Vehicle Routing Problem.

To do this we need to find out the best way of ordering the riders, so as to minimize route length. Since the capacity of a vehicle is fairly small, we find the best route by simply trying all combinations.

SYSTEM DESIGN

The system model uses remote server to store all the important information and share it through client-server model which overcome the drawback of storing on the mobile device. User preferences are given more importance in the mobile device. A modular approach technique is used to make this software simple to use for the customer. In a time when energy and fuel efficiency are pressuring people to leave their car at home and seek alternatives, our application can help people coordinate ride share programs and plan trips efficiently and effectively. People who share rides regularly will significantly reduce their ecological footprint and save on travel expenses. In order to accomplish this, our project must meet certain basic criteria. These criteria will ensure its proper functionality, reliability, and maintainability.

MATCHING

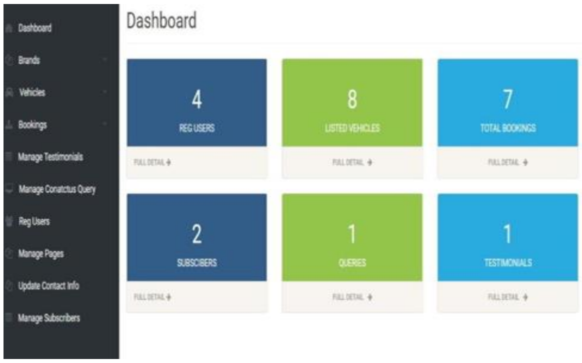
When matching riders and drivers, we must implement an algorithm (a logical set of steps which reach a solution to the problem), which generates the most efficient results. Efficiency is a measure of a number of parameters, foremost of which is maximal matching, i.e. the highest number of customers receiving rides. Maximal matching is crucial for the results to be effective and useful for our customers. Route lengths and times must also be minimized by the algorithm, and suggested rideshares should be viable, intelligent solutions. Timeliness of results is also important, since an effective result which does not complete in a timely manner is useless. This last requirement can be quantified as an algorithm which runs in polynomial-time, as opposed to exponential-time. This means it will complete some time tolerable by humans. Exponential-time algorithms can potentially take longer to complete than the history of the universe. The algorithm should also utilize external inputs efficiently. This can be quantified as an algorithm which minimizes route requests to Google-Maps by reducing the search-space, i.e. the number of possible valid matches. For example, if a driver is going from Austin to Houston, the algorithm would refrain from evaluating routes for which the riders' beginning and end points aren't in Texas. Those that are in Texas can further be reduced to riders' who are in central Texas, etc. Once this pre-processing has been completed, we would evaluate only those combinations within the limited search space which has been generated.

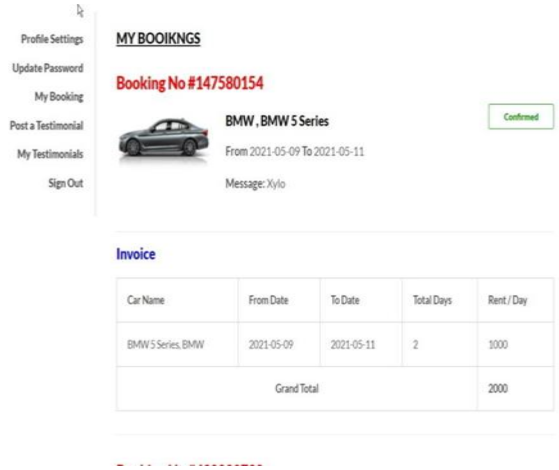
SERVER

Our final product will need a server that can support the projected bandwidth and storage requirements. The server must have an application server, preferably Apache, installed on it. Apache can run our implementation, which uses an SQL database and the programming language Ruby.

We will need an application development environment such as Eclipse, which supports web application development, including Ruby, SQL, and JavaScript. Our choice must be evaluated on its reliability, effectiveness, usability, and cost. Within this environment, we will need to enforce good programming practices, including a secure wiki and file server for collaboration, naming conventions, revision control, and modularization of programming tasks.

Output





MY BOOIKNGS

Booking No #147580154

BMW , BMW 5 Series

From 2021-05-09 To 2021-05-11

Message: Xylo

Invoice

Car Name	From Date	To Date	Total Days	Rent / Day
BMW 5 Series, BMW	2021-05-09	2021-05-11	2	1000
Grand Total				2000

ADVANTAGES

1. Traffic congestion.
2. Ridesharing helps those with limited mobility.
3. Ridesharing offers less discriminatory practices.
4. Ridesharing is quick, affordable, and safe.
5. Ridesharing has created millions of jobs.
6. Ride-sharing services offer door-to-door convenience, safety, and reliable quality.
7. Convenient.

CONCLUSION

We completed most of the tasks and features introduced in our project proposal. We created a web application for matching users who want to participate in carpools and solved a design problem that involved several complex challenges, including confronting an intractable optimization problem, designing an interactive, map-based user interface, and communicating results to users using email and RSS feeds. Moreover, we needed these components to work together harmoniously and utilize a common database. Confronting and overcoming these challenges provided us all with a major learning experience and a great feeling of accomplishment. Our approach began with setting up a collaborative work environment which would enable us to discuss how to solve each of the challenges, utilize a ticket system, and keep a complete revision history of all of the code. We designed the software with modularity in mind, in order to allow for simultaneous development by all team members.

This modularity ensured that we could easily add or replace features in the future. In addition, we used the Test-Driven Development paradigm which increased the reliability of code during each stage of development. There were four major components to our design for Rideshare: user interface, database, controller, and the optimizers. The idea behind this design was to decouple the user interaction from the optimization process, making the database an interface between these two processes. In addition, we developed a testing framework for unit-testing each individual component of the software, and also created a set of larger test cases for assessing the functional and time performance of the optimizers. Each of the major components had its challenges but in the end we completed all but two of the original features we proposed.If we were to launch this product, we would implement some of the features we had initially wanted to use, like friends lists. We would try to launch from within existing trust networks like Facebook and would try to add additional safety features.

We have each learned a lot from this experience in terms of the theoretical knowledge, technological tools, development environment, tea work, and communication skills. We feel that.

REFERENCES

[1] Blerim Cici et al., “Quantifying the Potential of Ride- Sharing using Call Description Records”, February - 2013.

[2] Milica Šelmić, “Ride Matching Using K-means Method: Case Study of Gazela Bridge in Belgrade, Serbia”, Journal of transportation engineering, January – 2012, 138, 132- 140.

[3] Ali Haghani, “Real-Time Rideshare Matching Problem”, January - 2011.

[4] BC Transit, “Transit Feasibility study for the Hazeltons”, September - 1996.

[5] Prasuna Reddy et al., “Advanced interfaces for a multi- option real-time ride share system”.

[6] D. J. Dailey et al., “Seattle Smart Traveler: Dynamic Ridematching on the World Wide Web”, Transportation Reseach Part C 7, March 1999.

[7] Gauresh Pandit et al., “Dynamic ridesharing using social media”, International Journal on AdHoc Networking Systems (IJANS) Vol. 2, No. 4, October - 2012.

[8] Swati.R.Tare et al., “Review Paper On CarPooling Using Android Operating System-A Step Towards Green Environment”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 4, April - 2013.

[9] Yan Huang et al., “Noah: A Dynamic Ridesharing System”, June - 2013.

[10] Dynamic Ridesharing Practice and Future Directions”, EURO Journal on Transportation and Logistics manuscript.

[11] Niels Agatz et al., “Optimization for dynamic ride-sharing: A review”, European Journal of Operational Research, 2012, 223, 295– 303.

[12] Sasha dos Santos, “A Geographical Information System for Dynamic Ride matching”, Graduate School Theses and Dissertations, March - 2005.