Distributed Storage Implementation in a Travel Bureau

Project Report
FOR THE DEGREE OF
BACHELOR OF TECHNOLOGY
IN
INFORMATION TECHNOLOGY

Submitted by:

Abhishek Vishwakarma-IIT2017091
Anuj Raj-IIT2017092
Kiran Kumar Datti-IIT2017094
Jai Khurana-ISM2017006
Nayan Agarwal-IWM2017501

Supervised By:

Dr. Anshu Anand
Assistant Professor

May 9, 2019

DECLARATION

We hereby declare that the work presented in this end semester project report of B.Tech (IT) 6th semester entitled “**Distributed Storage Implementation in a Travel Bureau**”, submitted by us at **Indian Institute of Information Technology, Allahabad,** is an authenticated record of our original work carried out from March 2020 to May 2020 under the guidance of **Dr. Anshu Anand.** Due acknowledgements have been made in the text to all other materials used. The project was done in full compliance with the requirements and constraints of the prescribed curriculum.

Place : Prayagraj
Date : 9th May,2020

Table Of Contents

ABSTRACT

This project implements an online travel agency as a web-based application that allows travelers to check for bus ticket availability, book a ticket ,check the running status of the bus and book a hotel. This website attracts service providers by allowing them to create and delete bus and hotel services.This web based travel bureau is implemented as a distributed storage. All data can be stored in multiple machines on a common network. The implementation also includes the notion of primary and secondary data servers, similar to GFS - Google File System. Location transparency and fault tolerance have been implemented properly to make good use of distributed storage. The application has a proper mechanism to handle read, write and updates.

1. Introduction

With the ever-growing technological expansion of the world, distributed systems are becoming more and more widespread. They are a vast and complex field of study in computer science.
A distributed system in its most simplest definition is a group of computers working together to appear as a single computer to the end-user.
These machines have a shared state, operate concurrently and can fail independently without affecting the whole system's uptime.
For us to distribute this database system, we'd need to have this database run on multiple machines at the same time.

With the quick headway of web based innovation, web based booking administrations has come as an incredible help to most travellers. Most travel offices just as different organizations that are occupied with travel business presently have on their site online transport ticketing booking gateways. From this website page one can undoubtedly book and pay for transport tickets on the web. Our venture is an online framework which uses the benefits of the web to give travel administrations extending from transport passes to movement and lodging reserving for clients from their homes. It gives a straightforward web interface to make the movement solicitation and administrations booking process simple.

2. Motivation

According to Claudia Leopold, there are eight main motivations for implementing a distributed system instead of simply utilizing the computing resources of a standard computer. These advantages are described briefly below:

(1) Distributed systems improve the "absolute performance" of the computing system

(2) The Price to Performance ratio for the system is more favorable for a distributed system

(3) Technological advantages

(4) Some applications are inherently distributed problems (they are solved most easily using the means of distributed computing

(5) Distributed computing allows the sharing of resources - both hardware and software

(6) Each piece of hardware is replaceable should it fail.

(7) Distributed Computing allows the system to grow incrementally as computers are added one by one.

(8) Distributed computing allows for “scavenging.” According to Leopold, “a lot of power is wasted, particularly during business hours. By integrating the computers into a distributed system, the excess computing power can be made available to other users or applications.

Our motivation was to learn how these features applies to a Online Travel Agency System. To experience the problem faced in implementing the concepts. And to refine the concepts of Distributed systems the we learned during the course.

3. Problem Definition

Implement Online Travel Agency as a distributed storage with following functionalities:

- Authentication - Register, Login, Recovery
- Book hotels and buses
- Be a service provider - create and delete hotel and bus services
- Manage services and bookings
- Be an admin - handle databases, user permissions and heartbeat rate
- Collaborate with users to manage services

4. Literature Review

The first Travel Agency of the world was established by Thomas Cook in 1845 in England. The use of the term travel trade dates back from the early years of the 19th century, but this should not obscure the fact that what we today describe as travel trade (travel agency and tour operation business) was taking place much earlier in history.
In 1841 a fortunate day came in the history of travel trade when Thomas Cook, as secretary of South Midland Temperance Association, organized a trip by a train for 570 members for his association to the distance of 22 miles. He bought railway tickets in bulk to sell them to people.

The experiment was successful and everybody was exultant. Mr. Cook had done his job on a no profit basis. But, incidentally, It gave him a new idea and turned it into a tour business.

Four years later in 1845, he set up a ‘World’s First Travel Agency‘ to organize excursions. Due to this innovative approach, Mr. Thomas Cook is known as the Father of Travel Agency Business. He co-ordinated railway and steamship excursions throughout England, Scotland and Europe.

However, the railways only gave him 5% commission which was not enough to meet his overheads, so he decided to diversify this business into tour operations.

In 1855 Mr. Cook started operating package tours. He conducted the world’s first international tour from England to Paris.

Travel agency is one of the most important organizations in the tourism private sector which plays a significant and crucial role in the entire process of developing and promoting tourism in the country or at a destination and helps in making arrangements for travel tickets and various travel related services.
This is a web based application for an online travel agency that helps travelers to check for bus ticket availability, book a ticket ,check the running status of the bus and book a hotel. Various services providers are attracted as they are allowed to create and delete bus and hotel services.

Distributed Database

A distributed database is a database in which data is stored across different physical locations.[1] It may be stored in multiple computers, located in the same physical location (e.g. a datacenter); or may be dispersed over a network of interconnected computers. Unlike parallel systems, in which the processors are tightly coupled and constitute a single database system, a distributed database system consists of loosely coupled sites that share no physical components.

System administrators can distribute collections of data (e.g. in a database) across multiple physical locations. A distributed database can reside on an organized network servers or decentralized independent computers on the Internet, on corporate intranets or extranets, or on other organization networks. Because distributed databases store data across multiple computers, distributed databases may improve performance at end-user worksites by allowing transactions to be processed on many machines, instead of being limited to one.[2]

Two processes ensure that the distributed databases remain up-to-date and current: replication and duplication.

Replication involves using specialized software that looks for changes in the distributive database. Once the changes have been identified, the replication process makes all the databases look the same. The replication process can be complex and time-consuming depending on the size and number of the distributed databases. This process can also require a lot of time and computer resources.

Duplication, on the other hand, has less complexity. It basically identifies one database as a master and then duplicates that database. The duplication process is normally done at a set time after hours. This is to ensure that each distributed location has the same data. In the duplication process, users may change only the master database. This ensures that local data will not be overwritten.

Both replication and duplication can keep the data current in all distributive locations.[2] Besides distributed database replication and fragmentation, there are many other distributed database design technologies. For example, local autonomy, synchronous and asynchronous distributed database technologies. These technologies' implementations can and do depend on the needs of the business and the sensitivity/confidentiality of the data stored in the database, and the price the business is willing to spend on ensuring data security, consistency and integrity.

Architecture

A database user accesses the distributed database through:

Local applications

applications which do not require data from other sites.

Global applications

applications which do require data from other sites.

A homogeneous distributed database has identical software and hardware running all databases instances, and may appear through a single interface as if it were a single database. A heterogeneous distributed database may have different hardware, operating systems, database management systems, and even data models for different databases.

Homogeneous Distributed Databases Management System

In a homogeneous distributed database, all sites have identical software and are aware of each other and agree to cooperate in processing user requests. Each site surrenders part of its autonomy in terms of right to change schema or software. A homogeneous DBMS appears to the user as a single system. The homogeneous system is much easier to design and manage. The following conditions must be satisfied for homogeneous database:

The data structures used at each location must be the same or compatible.
The database application (or DBMS) used at each location must be the same or compatible.

Heterogeneous DDBMS

In a heterogeneous distributed database, different sites may use different schema and software. Difference in schema is a major problem for query processing and transaction processing. Sites may not be aware of each other and may provide only limited facilities for cooperation in transaction processing. In heterogeneous systems, different nodes may have different hardware & software and data structures at various nodes or locations are also incompatible. Different computers and operating systems, database applications or data models may be used at each of the locations. For example, one location may have the latest relational database management technology, while another location may store data using conventional files or old version of database management system. Similarly, one location may have the Windows operating system, while another may have UNIX. Heterogeneous systems are usually used when individual sites use their own hardware and software. On heterogeneous system, translations are required to allow communication between different sites (or DBMS). In this system, the users must be able to make requests in a database language at their local sites. Usually the SQL database language is used for this purpose. If the hardware is different, then the translation is straightforward, in which computer codes and word-length is changed. The heterogeneous system is often not technically or economically feasible. In this system, a user at one location may be able to read but not update the data at another location.

JWT Application

A JSON Web Token (JWT) is a safe, compact, and self-contained way of transmitting information between multiple parties in the form of a JSON object.Say you want to log in to an app, like say Tinder. Tinder allows users to log in using their Gmail. So when the user selects the option to log in using Gmail, the app contacts google’s Authentication server with the user’s credentials (username and password). Once the Authentication server verifies the user’s credentials, it will create a JWT and sends it to the user. The app now gets this JWT and allows the user access to its data.

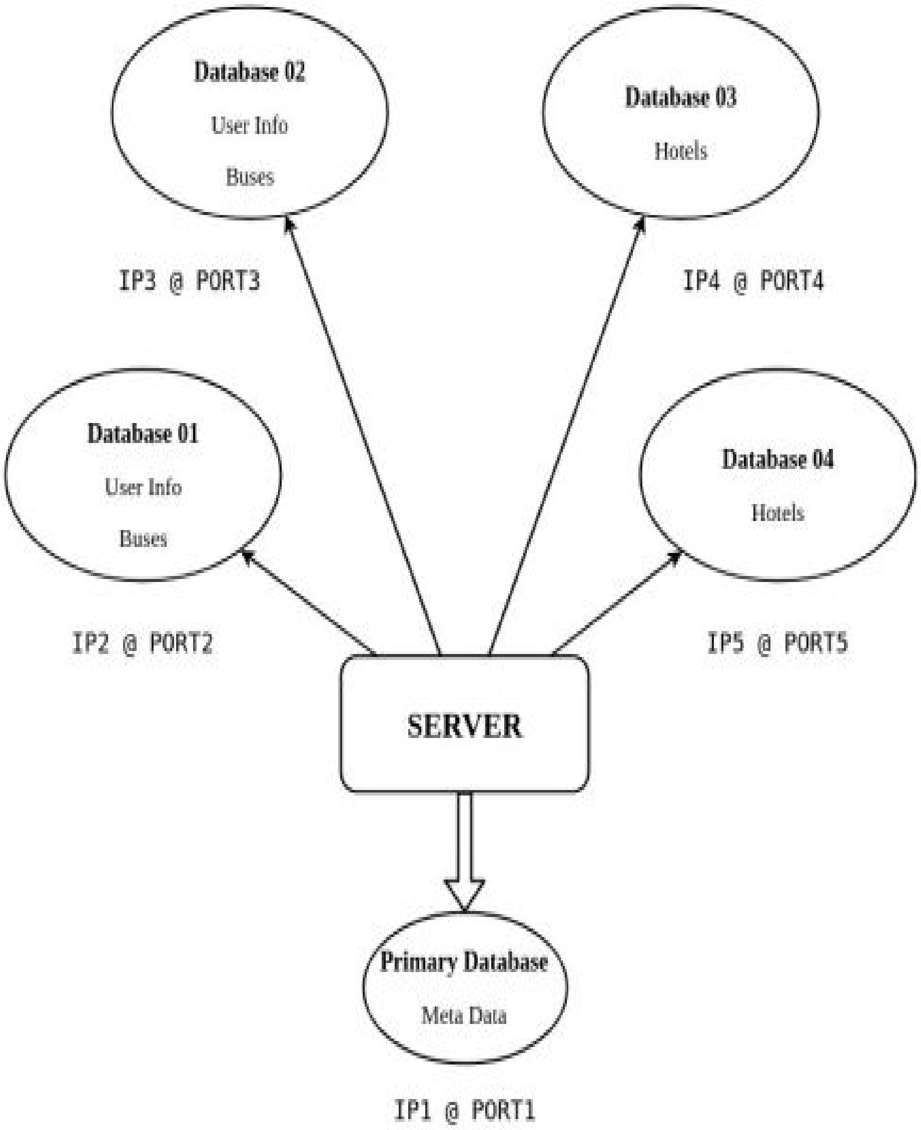
PostgreSQL

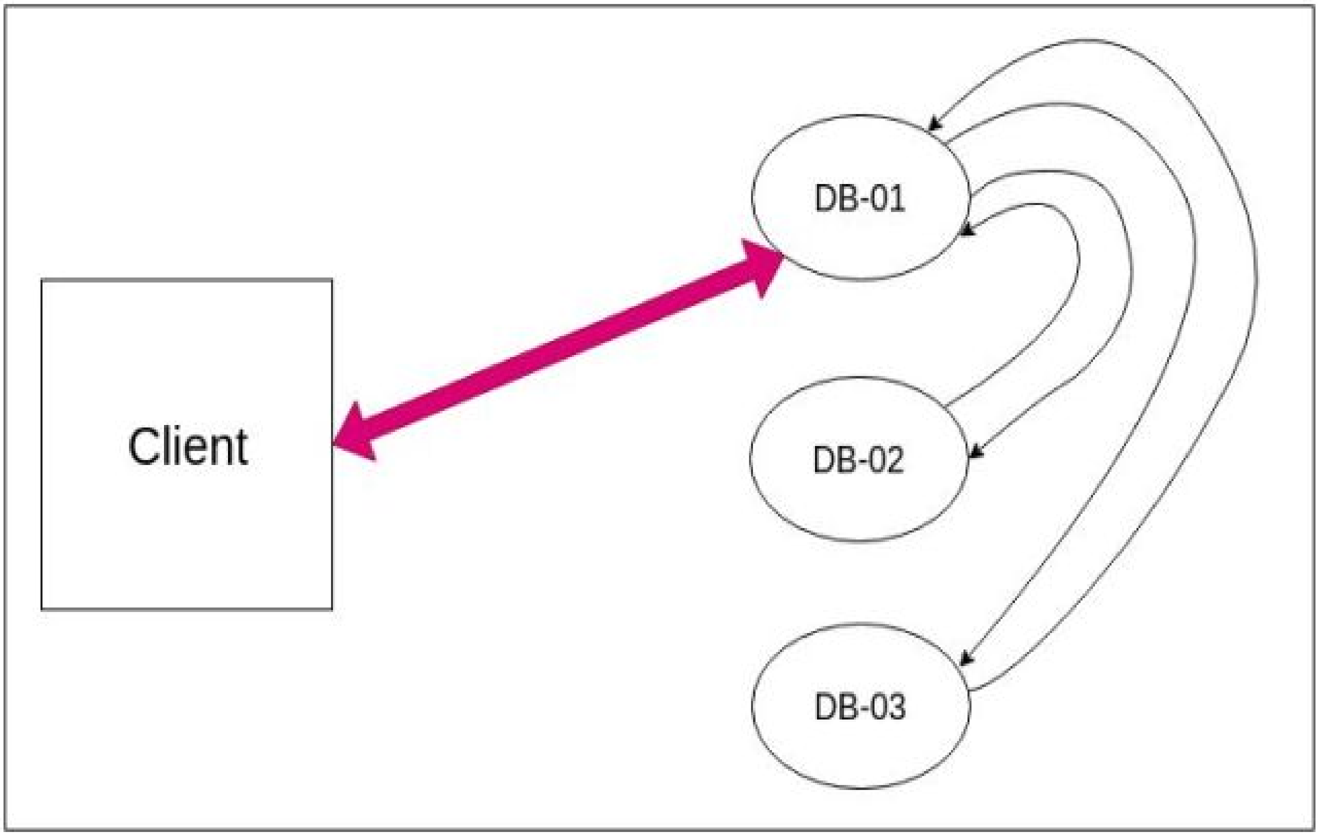
PostgreSQL is a general purpose and object-relational database management system, the most advanced open source database system. PostgreSQL was developed based on POSTGRES 4.2 at Berkeley Computer Science Department , University of California. PostgreSQL is free and open source software. Its source code is available under PostgreSQL license, a liberal open source license. You are free to use, modify and distribute PostgreSQL in any form. PostgreSQL requires very minimum maintained efforts because of its stability.

Therefore, if you develop applications based on PostgreSQL, the total cost of ownership is low in comparison with other database management systems.

5. Methodology

5.1 Architecture





5.2 ASSUMPTIONS and DETAILS

The master server never crashes (single point failure)

- ➢ More than 2 servers are never down at any point of time (replication = 3)
- ➢ Transaction handling is out of scope for this project
- ➢ There may be a finite delay in message requests
- ➢ There are no permanent crashes
- ➢ There are at least 3 databases available (active) at any point of time
- ➢ Primary server may not crash while waiting for response
- ➢ There is a single master server and 8 data servers
- ➢ Heartbeat rate can be changed by the admin
- ➢ New databases can be added only by the superuser
- ➢ Replication factor is set to 3, i.e 3 copies of any data are stored.

5.3 Handling Writes

These are possibilities of write request:

- Request received at master
- Request received at active data server

Handling Write Request received at master:

- Master forwards the request until 3 data servers respond positive
- Assign “primary” to one of the 3 servers and “secondary” to the other 2
- Returns status = 200 OK (positive) to the client

Handling Write Request received at active data server:

- Request forwarded by master
- Write data
- Response status = 200 OK to the master, if successful
- Response status = 404 BAD REQUEST (or any other relevant status), if unsuccessful

To create a new record, we need to select 3 active machines and insert data into them. Also, we name one of them as primary.

5.4 Handling Updates

Updates need to be consistent, meaning all the data servers should have the same updated record. As soon as the master receives the request for update, it forwards it to the primary data server of that record. It is the primary data servers' duty to update the remaining 2 secondary servers. This technique would help to reduce the load on the master. If it happens that any of the secondary data servers is down, then the primary informs the master regarding the pending update. The master later updates the record when the secondary server becomes active again. If it happens that the primary itself is down, then the master simple chooses a new primary and makes the current one a secondary.

These are possibilities of Update request:

- Request received at master
- Request received at primary server

Handling Update Request received at master:

- Figure out the primary database and secondary databases
- If primary is not active, make any of the active secondary server primary
- Forward update request to the primary and wait for response

Handling Update Request received primary server:

- Write data
- Forward the request to the secondary servers
- Return relevant response
    - All OK
    - Any DB not updated?

5.5 Handling Reads

To fetch any record, we iterate through the active data servers and try to get the required record. We return it to the user as soon as we get it from the database.

These are possibilities of read request:

- Request received at master server
- Request received at Data Server

Request received at master server:

- Forward request one after another to all active databases (or just 3, depends on the situation)
- Return as soon as any one responds

Request received at Data Server:

- Return status along with requested data

5.6 HeartBeat

- The master pings all data servers periodically to check which ones are crashed and which ones are active. The time period can be altered and set in the application by the admin**.**
- A parallel thread manages heartbeat requests along with the running masterserver
- Sends request to all registered data servers and marks their status
- Recommended to have heartbeat below 10 seconds, else user has to wait long to update data (receives Internal Error)

5.7 Handling Failures

As already mentioned in assumptions, the primary data server may not crash while waiting for a response.

Consider the following cases:

Case1:At Master

1. Make sure primary is active
2. Forward update request to primary

In case 1 if the primary server crashed in between steps 1 and 2 then the user receives a message - “Internal Error”.

Case2: At Master with Heartbeat rate = 30 seconds

1. T = 0, heartbeat detects primary data server alive
2. T = 5, primary data server fails
3. T = 10, update request for a record where primary is the failed server
4. Next heartbeat at t = 30

In this case also the user receives a message - “Internal Error”.

The replication factor is fixed to 3, meaning data would be inserted in 3 different machines (called data servers). Replication factor of 3 ensures that even if any two systems are down, the application would still run smoothly.And one among them will be the primary server and we assume that primary server never fails.

6.Software & Hardware Requirements

6.1 Tools used

1) Front End : HTML, CSS (Bulma UI), Javascript

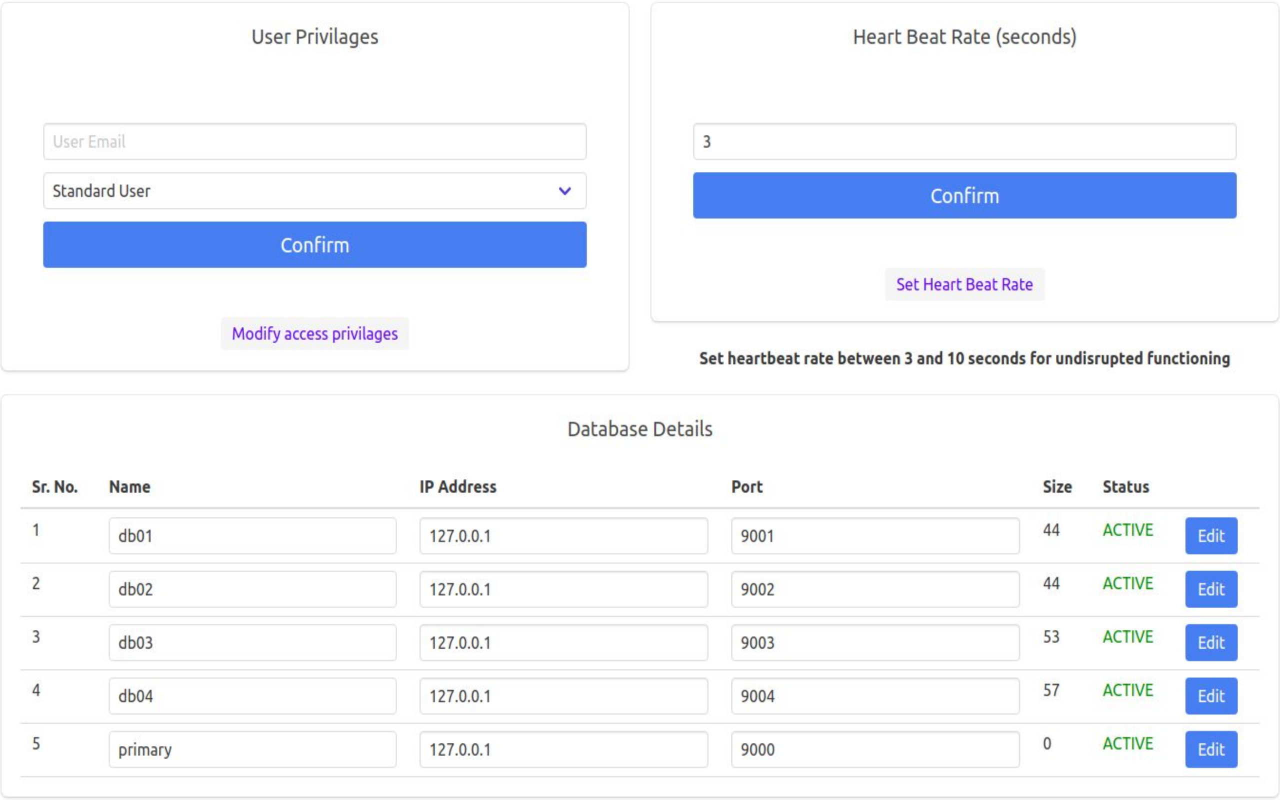
2) Back -End : Python Django

3) Database : PostgreSQL

7. Results

7.1 Database details , access Privileges and Heartbeat Rate

Here in Database details we can observe that there are 4 secondary databases at different ports which have the same data with a primary database which checks the status of all the secondary databases according to the heartbeat rate as discussed in the earlier sections.



User Privilages

User Email

Standard User

Heart Beat Rate (seconds)

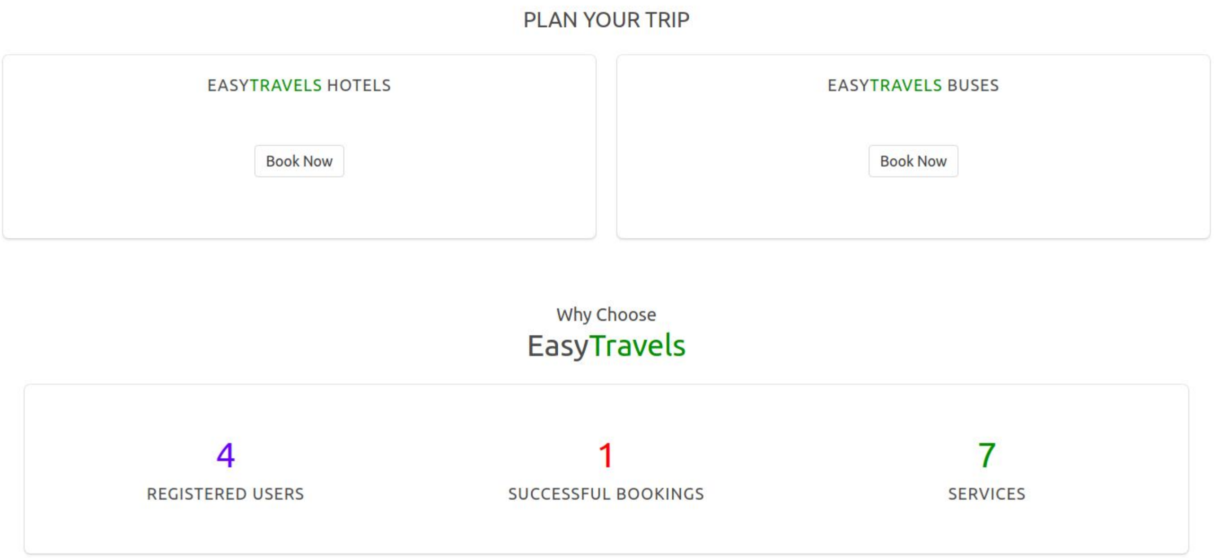
3

Set heartbeat rate between 3 and 10 seconds for undisrupted functioning

Database Details

Sr. No.	Name	IP Address	Port	Size	Status	
1	db01	127.0.0.1	9001	44	ACTIVE	Edit
2	db02	127.0.0.1	9002	44	ACTIVE	Edit
3	db03	127.0.0.1	9003	53	ACTIVE	Edit
4	db04	127.0.0.1	9004	57	ACTIVE	Edit
5	primary	127.0.0.1	9000	0	ACTIVE	Edit

7.2 Other User Interface Functionalities



PLAN YOUR TRIP

EASYTRAVELS HOTELS

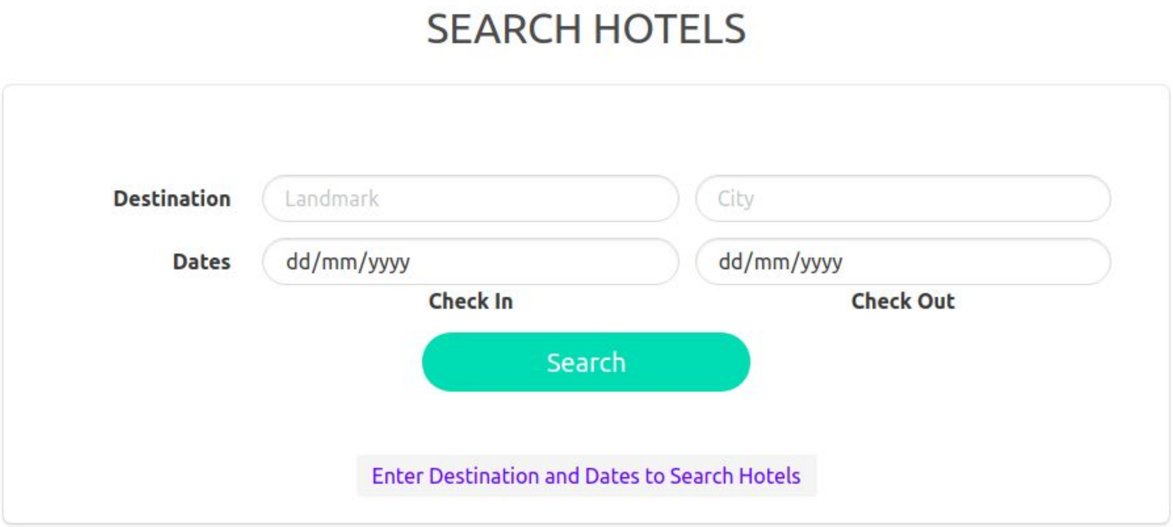
EASYTRAVELS BUSES

Why Choose
EasyTravels

4
REGISTERED USERS

1
SUCCESSFUL BOOKINGS

7
SERVICES



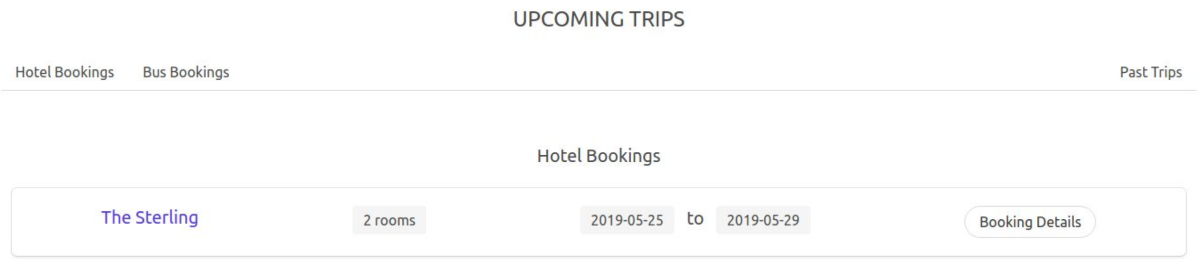
SEARCH HOTELS

Destination Landmark City

Dates dd/mm/yyyy dd/mm/yyyy

Check In Check Out

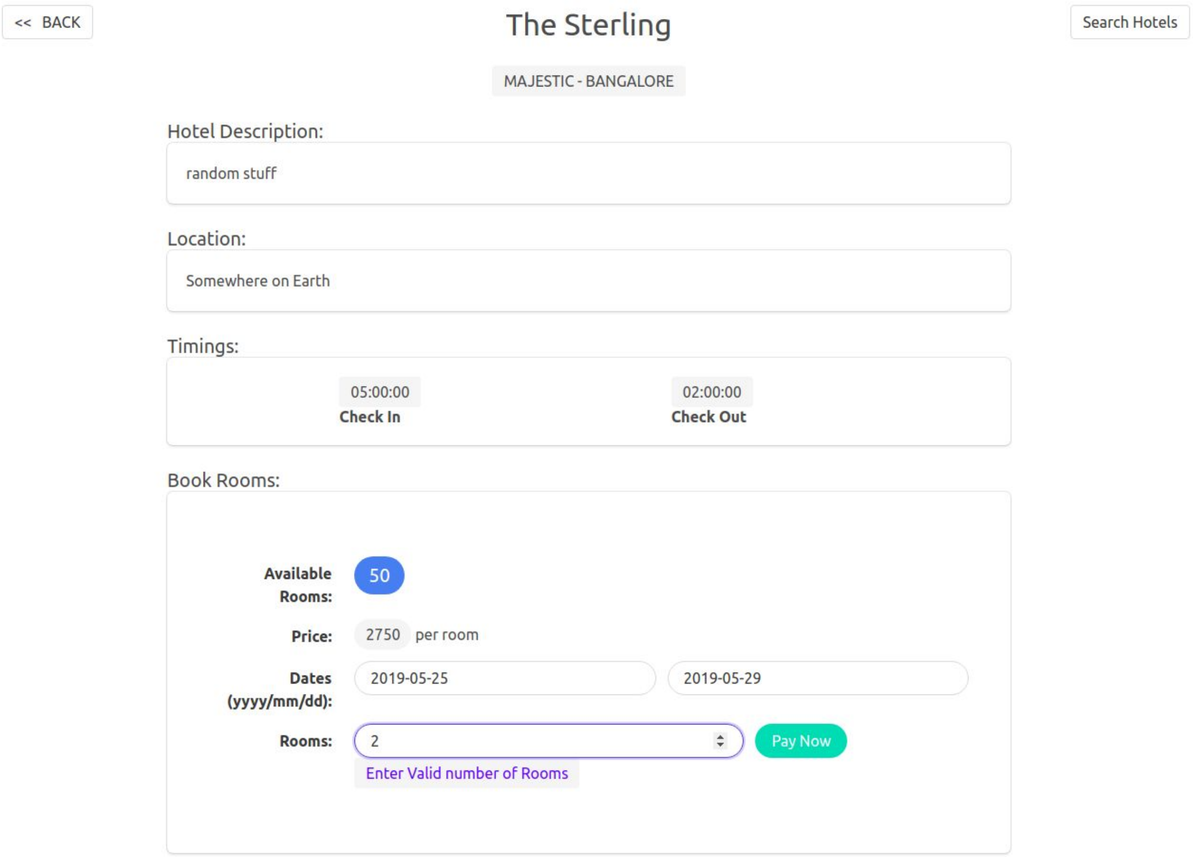
Enter Destination and Dates to Search Hotels



UPCOMING TRIPS

Hotel Bookings

The Sterling 2 rooms 2019-05-25 to 2019-05-29



The Sterling

MAJESTIC - BANGALORE

Hotel Description:

random stuff

Location:

Somewhere on Earth

Timings:

05:00:00 **Check In**

02:00:00 **Check Out**

Book Rooms:

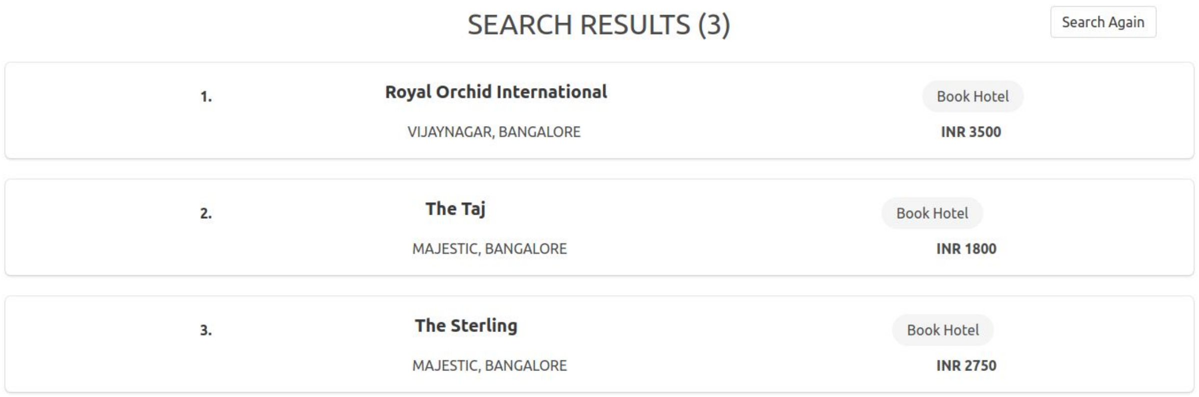
Available Rooms: 50

Price: 2750 per room

Dates (yyyy/mm/dd): 2019-05-25 2019-05-29

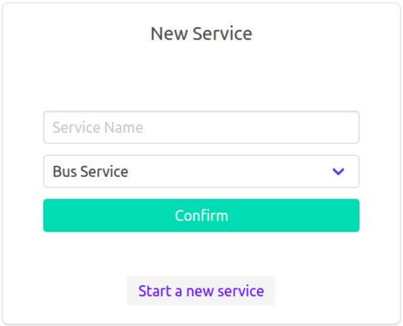
Rooms: 2

Enter Valid number of Rooms

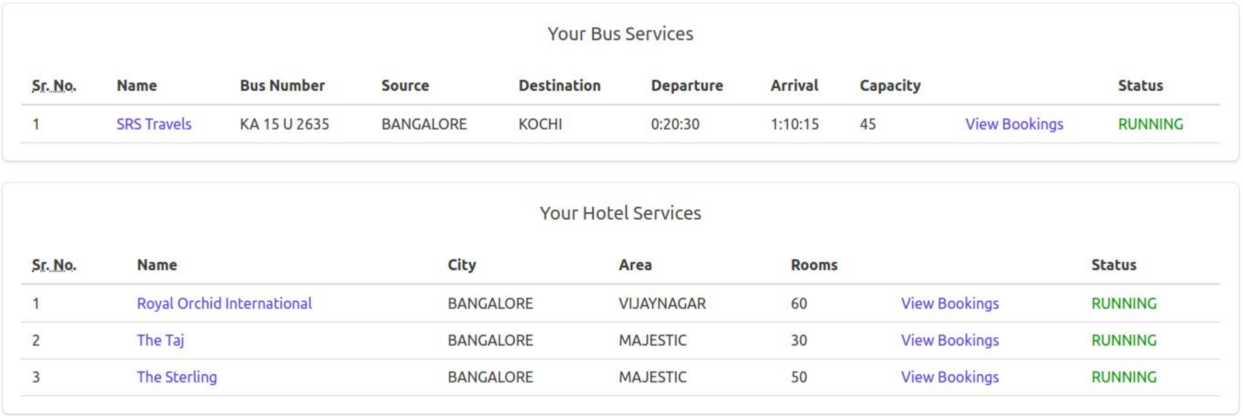


SEARCH RESULTS (3)

1.	**Royal Orchid International** VIJAYNAGAR, BANGALORE	Book Hotel	**INR 3500**
2.	**The Taj** MAJESTIC, BANGALORE	Book Hotel	**INR 1800**
3.	**The Sterling** MAJESTIC, BANGALORE	Book Hotel	**INR 2750**



New Service

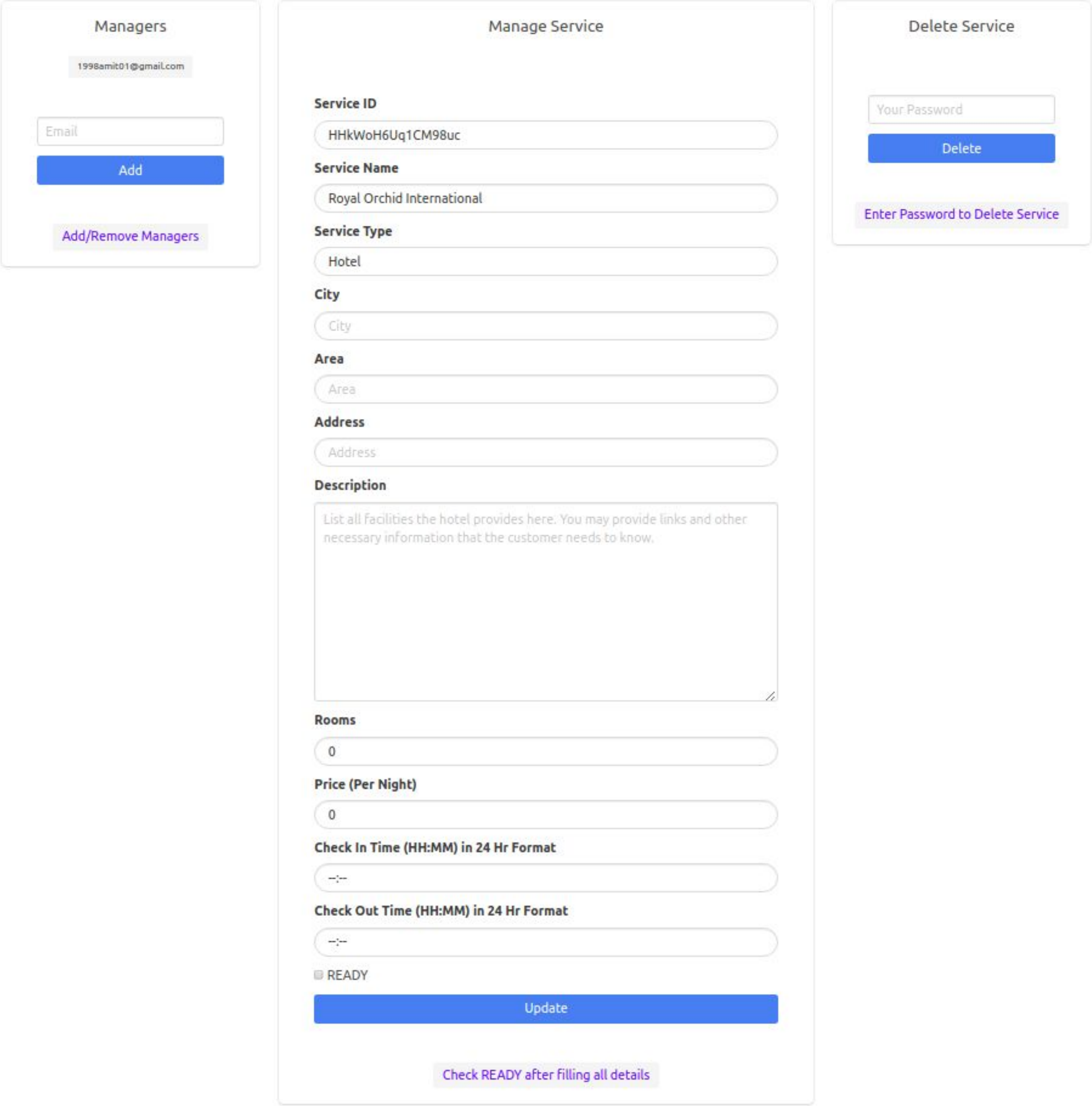


Your Bus Services

Sr. No.	Name	Bus Number	Source	Destination	Departure	Arrival	Capacity		Status
1	SRS Travels	KA 15 U 2635	BANGALORE	KOCHI	0:20:30	1:10:15	45	View Bookings	RUNNING

Your Hotel Services

Sr. No.	Name	City	Area	Rooms		Status
1	Royal Orchid International	BANGALORE	VIJAYNAGAR	60	View Bookings	RUNNING
2	The Taj	BANGALORE	MAJESTIC	30	View Bookings	RUNNING
3	The Sterling	BANGALORE	MAJESTIC	50	View Bookings	RUNNING



Managers

1998amit01@gmail.com

Email

Add/Remove Managers

Manage Service

Service ID

HHkWoH6Uq1CM98uc

Service Name

Royal Orchid International

Service Type

Hotel

City

City

Area

Area

Address

Address

Description

List all facilities the hotel provides here. You may provide links and other necessary information that the customer needs to know.

Rooms

0

Price (Per Night)

0

Check In Time (HH:MM) in 24 Hr Format

--:--

Check Out Time (HH:MM) in 24 Hr Format

--:--

□ READY

Check READY after filling all details

Delete Service

Your Password

Enter Password to Delete Service

8. Future Work

As a future scope of this project, we can add a recommendation section, where we can build an ML engine to provide necessary travel information to tourists. It must give up-to-date, accurate and timely information regarding destinations mode of travel, accommodation, sight-seeing, shopping, immigration, passport, visa, customs clearance and procedure, health and security rules and about various permits required to travel in particular areas etc.

9. References

[1] Xu, Lihao & Bruck, Jehoshua. (1998). Highly Available Distributed Storage Systems.. 307-330. 10.1007/BFb0110096.

[2]https://en.wikipedia.org/wiki/Distributed_database#cite_note-obrien-2

[3] F. E. Oggier and A. Datta, "Maintenance of distributed storage systems,” CoRR, vol. abs/1107.3129, 2011.