

Project Report

On

Travel Desk Management System

For

Harbinger Systems Pvt. Ltd.

By

Shreyas Vivek Barhanpurkar

Seat No: 11587

MCA (2017-2020)

May 05, 2020

Certificate

This is to certify that **Mr. Shreyas Barhanpurkar**, who is pursuing his MCA from Institute of Management and Career Courses (IMCC), has successfully completed his project “**Travel Desk Management System**” with us.

The project duration is from January 2020 to June 2020.

Shreyas is a sincere and hardworking person and is committed to his work.

We wish him all the best in his future endeavors.


Regards,

For **Harbinger Systems Pvt. Ltd.**

DocuSigned by:
Premlata Mangudkar
28C0A92983A74F8...

Premlata Mangudkar
Associate General Manager – Talent Management

Registered Office Address:

 Harbinger Systems Pvt. Ltd.
139, 'Siddhant', Survey No. 97/6, Off. Paud Road,
Kothrud, Pune 411 038, Maharashtra, India

 **Phone No.:** +91-20-6626 0100 / 6720 3000 / +91-20-25284201/2/3
Fax No.: +91-20-25285411

 **Website:** www.harbinger-systems.com
E-mail: info@harbinger-systems.com
CIN: U72200PN1997PTC112423

Maharashtra Education Society's



INSTITUTE OF MANAGEMENT AND CAREER COURSES (IMCC)

(Recognized by Savitribai Phule Pune University & Approved by AICTE)

131, Mayur Colony, Kothrud, Pune 411 038.

Tel. +91-20-25466271, 25463453 • E-mail: director.imcc@mespune.in

DR. SANTOSH DESHPANDE

Director

Web Site: <https://imcc.mespune.in>

Ref. No: MCA/Project/023/2020-21

Date : 14/09/2020

CERTIFICATE

This is to certify that the Project Report entitled "***Travel Desk Management System***" is prepared by ***Shreyas Vivek Barhanpurkar***, a student of ***M.C.A.*** Course for the Academic Year 2019-20 at

M. E. Society's Institute of Management & Career Courses (IMCC), Pune - 411 038. M.C.A Course is affiliated to Savitribai Phule Pune University.

To the best of our knowledge, this is original study done by the said student and important sources used by him have been duly acknowledged in this report.

The report is submitted in partial fulfillment of M.C.A. Course for the Academic Year 2019-20 as per the rules & prescribed guidelines of Savitribai Phule Pune University.

Dr. Ravindra Vaidya

HOD, Department of MCA, IMCC

Dr. Santosh Deshpande

Director, IMCC

Acknowledgement

I am very glad to take this opportunity to acknowledge all those who helped me in designing, developing and successful execution of my Project “**Travel Desk Management System**”.

I would like to extend my thanks and gratitude to my project guide **Mrs. Darshana Yadav** (Assistant Professor, IMCC) – Internal Guide and **Mr. Abhay Damle** - External Guide for their valuable guidance and timely assistance throughout the development of this project.

I would also like to extend my thanks and gratitude to **Dr. Santosh Deshpande** (Director, IMCC), **Dr. Ravindra Vaidya** (HOD, IMCC), **Dr. Manasi Bhate** (Head – Training and Placement, IMCC) and **Ms. Apurwa Barwe** (Assistant Professor, IMCC and Class Coordinator - TYMCA) for their constant help and support.

Last but not the least, I would like to thank all the teaching and non-teaching faculties for their cooperation.

- Shreyas Barhanpurkar

INDEX

Sr. No.	Name of Topic	Page No.
1	Chapter 1 : Introduction	
	1.1 Company Profile	1
	1.2 Existing System and Need for System	5
	1.3 Scope of Work	11
	1.4 Operating Environment – Hardware and Software	15
	1.5 Detail Description of Technology Used	17
2	Chapter 2 : Proposed system	
	2.1 Proposed System	28
	2.2 Objectives of System	35
	2.3 User Requirements	40
3	Chapter 3 : Analysis & Design	
	3.1 Object Diagram	42
	3.2 Class Diagram	43
	3.3 Use Case Diagrams	44
	3.4 Activity Diagrams	48
	3.5 Sequence Diagrams	51
	3.6 Entity Relationship Diagram	54
	3.7 Module Hierarchy Diagram	55
	3.8 Component Diagram	56
	3.9 Deployment Diagram	57
	3.10 Module Specifications	58
	3.12 Web Site Map Diagram	68
	3.13 User Interface Design	69
	3.14 Data Dictionary	75
	3.15 Table specifications	84

	3.16 Test Procedures and Implementation	96
4	CHAPTER 4 : USER MANUAL	
	4.1 User Manual	128
	4.2 Operations Manual / Menu Explanation	134
	4.3 Program Specifications / Flow Charts	136
5	Drawbacks and Limitations	141
6	Proposed Enhancements	142
7	Conclusions	143
8	Bibliography	
9	ANNEXURES :	
	ANNEXURE 1 : USER INTERFACE SCREENS	
	ANNEXURE 2 : OUTPUT REPORTS WITH DATA	
	ANNEXURE 3 : SAMPLE PROGRAM CODE	

CHAPTER 1
INTRODUCTION

1.1 Company Profile

Harbinger Systems is a global company providing software technology services for independent software vendors and enterprises. Since its inception in 1990, Harbinger has developed a strong customer base of organizations worldwide that includes high-tech start-ups in Silicon Valley, multi-national product companies, and in-house IT teams of large organizations.

Harbinger Systems leverages the latest digital technologies to build software solutions in HR Tech, Health Tech, and Learn Tech domains, and helps solve complex business problems in these areas for organizations across industries.

Harbinger Systems is a part of Harbinger Group. Harbinger Group is a global provider of innovative software products and services to companies in over 60 countries. Harbinger Group's philosophy is to create value for its clients through a culture of continuous learning, respect for the individual and innovation.

We are a fast growing company with corporate headquarters and three development centers in Pune, in India. We have our sales and business development office in Redmond in the US. Harbinger is led by entrepreneurs who guide their teams towards creating excellence and absolute customer satisfaction; while achieving growth for everyone in the organization.

The top management at Harbinger has wealth of experience in product innovation, product development and cadre building. Learn more about the leadership team at the helm of Harbinger Systems.

Harbinger Systems has been recognized by distinguished research analysts, leading software industry organisations and global clients. Numerous awards and customer accolades are testimonials to this. In 2009, Deloitte named Harbinger Systems among the fastest growing technology companies in its 'Technology Fast 500 Asia Pacific' and 'Technology Fast 50 India' programs. Harbinger Systems was recognized as a 'Super Star' in 'Annual Mobile Star Awards for the years 2016, 2013 and 2011. Harbinger Systems is ranked in 'The

Global Outsourcing 100®' listing by International Association of Outsourcing Professionals for five years in a row.

Harbinger Group Values:

CUSTOMER CENTRICITY

We will strive to be a trusted partner to our customers instead of being just another vendor.

DEVELOPING PEOPLE

Our objective will be to develop Harbingers to become strong, positive, and vibrant leaders of the future.

MERITOCRACY

Harbinger will be a place where your performance, commitment, and output determine how far you will go.

AGILITY

We will encourage people to think out-of-the-box and act with speed and flexibility.

PROFESSIONALISM

We will honor commitments, be accountable to all our stakeholders, be respectful, and maintain high standards of corporate governance.

INCLUSIVITY

We will build an inclusive workplace that is diverse, sensitive, and committed to providing equal opportunity to all.

1.2 Existing System and Need for system

Existing System:

Many times employees have to travel to client location for some work, which may include travelling either domestically or internationally.

Many of the clients of Harbinger are from outside of India.

When developing solutions for them, there occur many instances where the company needs to interact with the client.

Various communications mediums are used for the same.

Sometimes it may require that some company representative needs to visit the client personally for face-to-face meeting.

In harbinger, currently if anyone has to travel on-site, either domestically or internationally, the following process is followed:

- 1) Employee provides the details like employee id, name, mobile number, address, city to travel, travel date, number

of days, passport and visa details(applicable for international travel), insurance nominee details, client name, address of client, etc. in an excel sheet and sends it along with the application to his/her Delivery Manager for approval.

- 2) The delivery manager verifies the details manually and then either approves or rejects the request.
- 3) If the request requires some changes in it, then the Delivery Manager personally calls or sends email to the requestor and then again the employee has to send the excel sheet with updated data.
- 4) If the request is rejected then the employee has to cancel his/her travel.
- 5) Once the delivery manager approves the request, he/she forwards the request to Admin head for his/her approval.
- 6) Before approving the request, the Delivery manager decides whom to charge the expense of the travel.

Sometimes the client takes all the expense of the travel, while sometimes Harbinger spares the expense itself.

- 7) The admin head checks the travel request details and approves or rejects the request considering the expense, the designation of the employee, etc.
- 8) If the request is approved, then the admin head first calls transportation booking vendor for flight/train/bus/taxi reservations. If the booking is not available for the desired dates, then the employee is asked to change the dates of his/her tour.
- 9) Then the admin head contacts the accommodation vendor for hotel bookings.
- 10) If the trip type is international, then the details of passport and visa of the traveler are verified and checked if they are valid. If the employee is not having passport/visa, then he/she is provided assistance for the same.
- 11) The admin head confirms with the employee whether he/she is having personal business cards. If not, then the

admin head contacts the printing vendor and sends the details of the employee and orders visiting cards for him/her.

12) Also, the admin head contacts the travel insurance agent and asks the employee for the nominee details and share to be paid to the nominee and accordingly takes the insurance for the employee travelling.

13) When the employee actually travels and if he/she wants to return early than desired return date, then he/she contacts the Delivery Manager and asks for approval. If approved, admin head checks for related travel bookings.

14) During all this process, the employee (requestor) needs to take update of his/her travel request from the concerned authorities personally.

15) After the entire tour is completed, the employee submits the hard copy of all the bills like hotel bills, transport bills, stationery bills.

Need for System

- 1) Employee needs to have the format of the excel sheet.
- 2) Filling a lot of details in excel sheet is not much convenient and easy as compared to an online form.
- 3) Employee needs to send the travel request to his/her delivery manager, and delivery manager has to forward it to admin head if the request is approved.
- 4) Admin needs to call the respective vendors for enquiry regarding the availability of bookings on desired dates.
- 5) The employee needs to contact the concerned authorities to track the status of his/her travel request till he/she gets the confirmation of either approval or rejection.
- 6) Once the employee sends the travel request and if he/she has to make changes in any of the details, then he/she has to restart the process of approval.
- 7) The validity of passport/visa expiry needs to be manually checked.

8) If the employee is not having his/her business cards, the admin team has to manually inform the vendor to get it printed before the date of travel.

1.3 Scope of work

- 1) This web application will use SSO for login, meaning that the login credentials will be the same as the employee's other harbinger web applications credentials. All the basic employee information will be fetched from an external API.
- 2) After login, the employee will have option to raise a new travel request. The travel request will take the basic employee details like company name, employee name, employee id, gender, date of birth, designation, etc. automatically from the database.
- 3) The employee will need to provide few details like project name, delivery head, destination city, departure date, return date, client address, mode of travel, meal preference, accommodation details.
- 4) If the employee is travelling abroad he/she will have to provide passport details like passport number, place of issue, date of issue, expiry date and Visa details like visa

number, visa country, visa category, visa type, visa valid from, visa expiry, type of entry, additional visa details, etc.

- 5) Also, for international travel, travel insurance nominee details need to be provided by the employee.
- 6) If the passport is expiring in less than 6 months, then request won't be taken ahead, as it is not allowed to travel in such situation. An email will be sent to the employee informing him/her to apply for renewal.
- 7) Once the details are filled by employee and he/she submits the form, all the details will be saved to the database and the request status will be set to "pending" and an email will be sent to Delivery Manager informing about the new request that is raised.
- 8) The employee will be able to view the status of his/her request on his dashboard. Also, option will be provided to view or edit the travel request details.

- 9) The delivery manager's and admin head's dashboard will have two tabs, one of his/her own travel requests and one of pending approvals.
- 10) Based on the details provided in the travel request, the Delivery Manager will either approve or reject the request. If the request is rejected, appropriate e-mail will be sent to the requestor along with the reason of rejection provided by the DM. If the request is approved, an email will be sent to admin head indicating that the request is forwarded to the admin head. Requestor will be informed about this too.
- 11) Option will be given to the Delivery Manager and the admin head to send the request back for correction.
- 12) The employee will be able to re-schedule the trip if required.
- 13) After the entire tour is done, option will be given to the employee to upload all the bills associated with his/her tour.

14) Admin head will be able to view all the bills of each travel that any employee has made.

Then after reading the details of the travel request and considering the approximate expense and designation of the employee, the admin head will either approve or reject the request. If the request is approved and the employee is not having business cards, then the details of the employee required for the business card will be automatically sent to the card printing vendor.

1.4 Operating Environment

HARDWARE REQUIREMENTS –

PROCESSOR: i3 3rd Gen Quad core

HARD DISK: 200GB

RAM: 16GB

OUTPUT DEVICES: LCD Monitor, Printer

INPUT DEVICES: Keyboard, Mouse

SOFTWARE REQUIREMENTS –

Server Side -

OPERATING SYSTEM- Any operating system

FRONT END- React JS, Bootstrap, CSS

MIDDLEWARE – Node.js

BACK END - Mongo DB

Client Side -

OPERATING SYSTEM- Any operating system

WEB BROWSER- Mozilla Firefox, Google Chrome or any
compatible web browser

1.5 Detail Description of technology used

Front End:

React JS:

1) Declarative-

React makes it painless to create interactive UIs. Design simple views for each state in your application, and React will efficiently update and render just the right components when your data changes. Declarative views make your code more predictable and easier to debug.

2) Component-Based

Build encapsulated components that manage their own state, then compose them to make complex UIs.

Since component logic is written in JavaScript instead of templates, you can easily pass rich data through your app and keep state out of the DOM.

3) Learn Once, Write Anywhere

We don't make assumptions about the rest of your technology stack, so you can develop new features in React without rewriting existing code.

React can also render on the server using Node and power mobile apps using React Native.

4) A Simple Component

React components implement a `render ()` method that takes input data and returns what to display. This example uses an XML-like syntax called JSX. Input data that is passed into the component can be accessed by `render ()` via `this.props`.

JSX is optional and not required to use React. Try the Babel REPL to see the raw JavaScript code produced by the JSX compilation step.

5) A Stateful Component

In addition to taking input data (accessed via this. Props), a component can maintain internal state data (accessed via this. State). When a component's state data changes, the rendered markup will be updated by re-invoking render ().

6) An Application

Using props and state, we can put together a small Todo application. This example uses state to track the current list of items as well as the text that the user has entered. Although event handlers appear to be rendered inline, they will be collected and implemented using event delegation.

7) A Component Using External Plugins

React allows you to interface with other libraries and frameworks. This example uses **remarkable**, an external Markdown library, to convert the <textarea>'s value in real time.

Middleware:

Node JS:

As an asynchronous event-driven JavaScript runtime, Node.js is designed to build scalable network applications. In the following "hello world" example, many connections can be handled concurrently. Upon each connection, the callback is fired, but if there is no work to be done, Node.js will sleep.

This is in contrast to today's more common concurrency model, in which OS threads are employed. Thread-based networking is relatively inefficient and very difficult to use. Furthermore, users of Node.js are free from worries of dead-locking the process, since there are no locks. Almost no function in Node.js directly performs I/O, so the process never blocks. Because nothing blocks, scalable systems are very reasonable to develop in Node.js.

If some of this language is unfamiliar, there is a full article on Blocking vs. Non-Blocking.

Node.js is similar in design to, and influenced by, systems like Ruby's Event Machine and Python's Twisted. Node.js takes the event model a bit further. It presents an event loop as a runtime construct instead of as a library. In other systems, there is always a blocking call to start the event-loop. Typically, behavior is defined through callbacks at the beginning of a script, and at the end a server is started through a blocking call like `EventMachine::run()`. In Node.js, there is no such start-the-event-loop call. Node.js simply enters the event loop after executing the input script. Node.js exits the event loop when there are no more callbacks to perform. This behavior is like browser JavaScript — the event loop is hidden from the user.

HTTP is a first-class citizen in Node.js, designed with streaming and low latency in mind. This makes Node.js well suited for the foundation of a web library or framework.

Node.js being designed without threads doesn't mean you can't take advantage of multiple cores in your environment. Child processes can be spawned by using our `child_process.fork()` API, and are designed to be easy to communicate with. Built upon that same interface is the cluster module, which allows you to share sockets between processes to enable load balancing over your cores.

Backend :

MongoDB :

MongoDB is a document database with the scalability and flexibility that you want with the querying and indexing that you need.

MongoDB is cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with schema. MongoDB

is developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL).

Main features

Ad hoc queries

MongoDB supports field, range query, and regular expression searches.^[9] Queries can return specific fields of documents and also include user-defined JavaScript functions. Queries can also be configured to return a random sample of results of a given size.

Indexing

Fields in a MongoDB document can be indexed with primary and secondary indices.

Replication

MongoDB provides high availability with replica sets. A replica set consists of two or more copies of the data. Each replica set member may act in the role of primary or secondary replica at any time. All writes and reads are done on the primary replica by default. Secondary replicas maintain a copy of the data of the primary using built-in replication. When a primary replica fails, the replica set automatically conducts an election process to determine which secondary should become the primary. Secondaries can optionally serve read operations, but that data is only eventually consistent by default.

Load balancing

MongoDB scales horizontally using sharding. The user chooses a shard key, which determines how the data in a collection will be distributed. The data is split into ranges (based on the shard key) and distributed across multiple shards. (A shard is a master with one or

more replicas.). Alternatively, the shard key can be hashed to map to a shard – enabling an even data distribution.

MongoDB can run over multiple servers, balancing the load or duplicating data to keep the system up and running in case of hardware failure.

File storage

MongoDB can be used as a file system, called GridFS, with load balancing and data replication features over multiple machines for storing files.

This function, called grid file system, is included with MongoDB drivers. MongoDB exposes functions for file manipulation and content to developers. GridFS can be accessed using mongofiles utility or plugins for Nginx and lighttpd. GridFS divides a file into parts, or chunks, and stores each of those chunks as a separate document.

Aggregation

MongoDB provides three ways to perform aggregation: the aggregation pipeline, the map-reduce function, and single-purpose aggregation methods.

Map-reduce can be used for batch processing of data and aggregation operations. But according to MongoDB's documentation, the Aggregation Pipeline provides better performance for most aggregation operations.

The aggregation framework enables users to obtain the kind of results for which the SQL GROUP BY clause is used. Aggregation operators can be strung together to form a pipeline – analogous to Unix pipes. The aggregation framework includes the \$lookup operator which can join documents from multiple collections, as well as statistical operators such as standard deviation.

Server-side JavaScript execution

JavaScript can be used in queries, aggregation functions (such as MapReduce), and sent directly to the database to be executed.

Capped collections

MongoDB supports fixed-size collections called capped collections. This type of collection maintains insertion order and, once the specified size has been reached, behaves like a circular queue.

Transactions

Support for multi-document ACID transactions was added to MongoDB with the General Availability of the 4.0 release in June 2018.

CHAPTER 2
PROPOSED SYSTEM

2.1 Proposed System

The Travel Desk Management System is divided into functionalities based on the login of the user and the functions available after login will be based on the type of the user.

1) Login –

For Login purpose, the web application will use SSO(Single Sign-On) feature, meaning that the employee can use the same login credentials as he/she uses for logging in to other applications of the same company.

For e.g. Using the same credentials of one Google account, we can login to youtube, google playstore, duo, google maps, and other google applications.

Based on the user id returned from the database, the employee login will be categorized as Normal Employee login, Delivery Manager login, or Admin Head Login.

2) Employee –

When the employee logs in, he/she will be able to see a dashboard of his/her travel requests. The dashboard will be blank if the employee hasn't made any travel request in the past.

Employee will have an option to raise a new travel request. In the travel request he/she will have to fill the following details

- **Travel Details –**
 - Trip type
 - Travel type
 - Source city
 - Destination city
 - Departure date
 - Return date
 - Mode of travel
 - Meal Preference
 - Frequent Flyer Number

- Accommodation requirement,etc
- **Passport Details (Applicable for International Travel) –**
 - Passport Number
 - Place of issue
 - Date of issue
 - Date of expiry
 - Upload scanned copy
- **Visa Details (Applicable for International Travel) –**
 - Visa number
 - Visa country
 - Visa category
 - Visa type
 - Valid from
 - Valid till
 - Type of entry
 - Additional visa
- **Insurance Details –**

- Nominee name
- Relationship with employee
- Date of birth
- Share to be paid
- Address

After filling these details and submitting the request, all the details will be stored in the database and an acknowledgement email will be sent to the employee.

Also, email will be sent to his/her Delivery Manager informing about the new request and ask to approve or reject the request.

The employee dashboard will now have the request details that he/she has made along with its approval status.

Options will be given to view or edit the details he/she has filled.

Employee will be able to reschedule his/her trip.

Also, after the trip he/she can upload all the bills associated with his/her tour.

3) Delivery Manager –

The delivery manager dashboard will have two tabs – one to display his/her own travel requests, and another to display the pending approvals that he/she has.

In the pending approvals tab, the details of the basic travel request details will be displayed of each request that is received. Option will be given to view the full travel request details that an employee has filled.

After reviewing the details, the Delivery Manager can either approve or reject the request. Before approving, an option will be given to select whom to charge the travel expense, as sometimes the company spares the expense, while sometimes client does it.

If the request is to be rejected, then the DM can specify the reason for rejection. If the request is approved then an email is sent to the

requestor informing him/her the request status and another email to admin head informing him/her about the request and ask for approval.

The Delivery Manager can also send the travel request back to the employee for corrections.

4) Admin Head –

After viewing the travel request details and the approximate expense required for the travel, the admin head will either approve or reject the travel request.

If the request is approved then an email is sent to the requestor informing him/her the request status.

After the request is approved by admin head, if the employee is not having his/her own business cards, then request email will be sent to the card printing vendor along with the required details of the employee. Also, appropriate emails will be sent

to transport, accommodation and other vendors for booking requests.

The admin head will be able to view the bills that an employee has submitted after his/her trip is completed.

2.2 Objectives of System

The Travel Management System will be having different users and the objectives of the system are categorized based on the type of the user.

1) Employee –

- The employee should be able to log-in to the system using his/her common login credentials used for all other harbinger applications.
- Immediately after the login, the employee dashboard should display a list (in a tabular format) of all the travel requests that he/she has made.
- The employee should be able to view all the details of the travel request that he/she has made.
- The employee should be able to edit the changes in the travel request.
- The employee should be able to view the approval status of his/her request on the dashboard itself.
- The employee should be able cancel his/her travel request.

- After the trip is completed, the employee should be able to upload the bills associated with his/her trip.
- The employee should receive an email if his/her passport is expiring in 6 months.

2) Delivery Manager –

- The Delivery Manager should be able to log-in to the system using his/her common login credentials used for all other harbinger applications.
- Immediately after the login, the Delivery Manager dashboard should display a list (in a tabular format) of all the travel requests that he/she has made.
- The Delivery Manager should be able to view all the details of the travel request that he/she has made.
- The Delivery Manager should be able to edit the changes in the travel request.

- The Delivery Manager should be able to view the approval status of his/her request on the dashboard itself.
- The Delivery Manager should be able to cancel his/her travel request.
- After the trip is completed, the Delivery Manager should be able to upload the bills associated with his/her trip.
- In another tab on the dashboard itself, the list of pending approval requests should be displayed along with an option to approve, reject or send request back for correction.
- Before approving any request, the Delivery Manager should be able to select whom to charge the travel expense.
- If a travel request is to be rejected, then the delivery manager should be able to specify the reason for rejection.

- The Delivery Manager should be able to send the request back to the employee for correction.

3) Admin Head -

- The Admin head should be able to log-in to the system using his/her common login credentials used for all other harbinger applications.
- The admin head dashboard should display a list of travel requests that are approved by the Delivery Manager and are pending for approval from admin head side.
- If a travel request is to be rejected, then the admin head should be able to specify the reason for rejection.
- The admin head should be able to send the request back to the employee for correction.

- Once the admin head approves the request, then automatically an email should be sent to the respective vendors for the bookings required for that travel.
- If the employee is not having personal business cards, then on approval of the request, an email should be sent to the printing vendor along with the details of the employee required for business cards, and the quantity of cards to be printed.

2.3 User Requirements

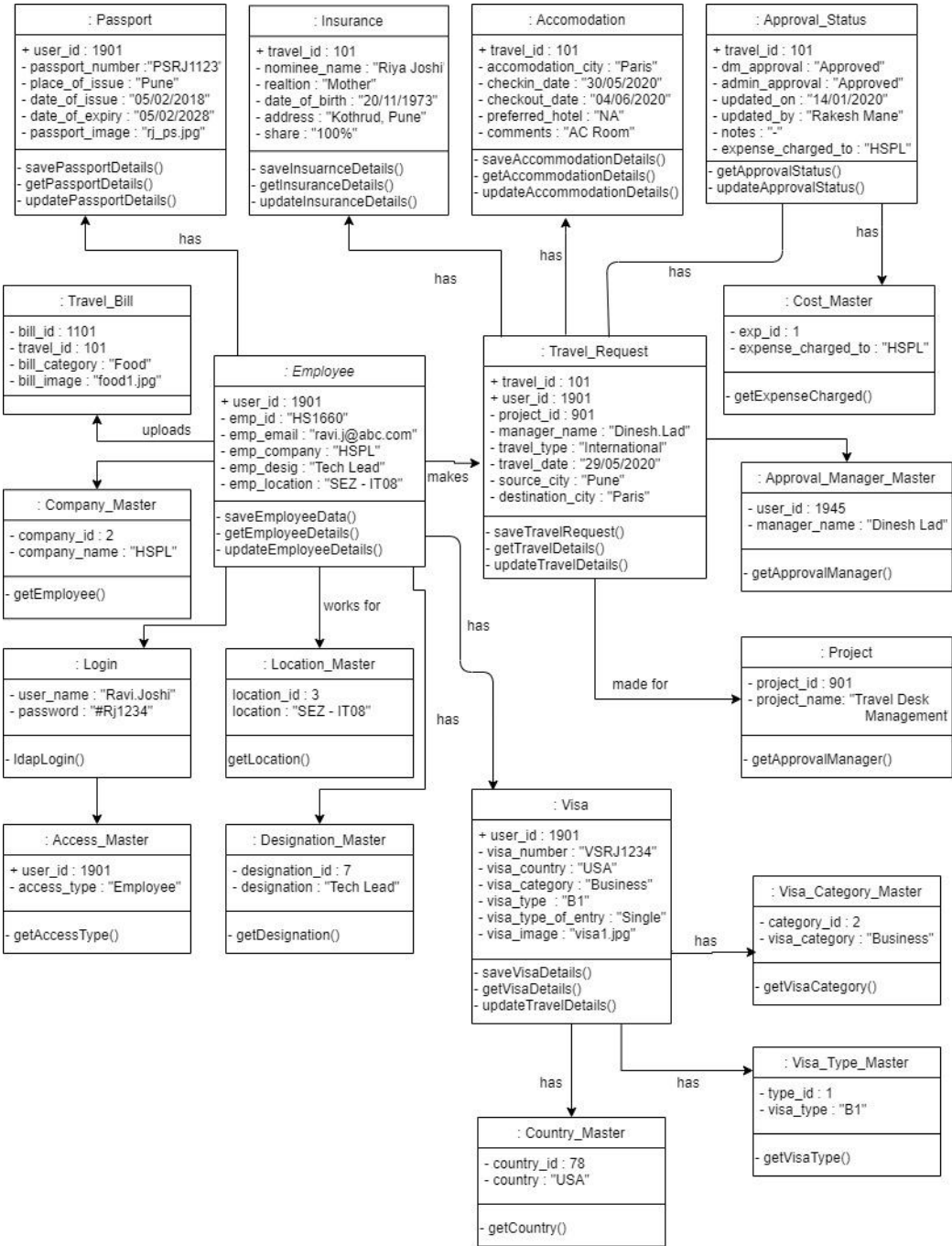
The client wants to have the following functionalities from the system:

- Any user should be able to log-in to the system using his/her common login credentials used for all other harbinger applications.
- The travel request form should be easy to understand and the user should not be required to fill the common personal details, as they should be retrieved from the employee database.
- Employee should be able to upload the scanned copy of his/her passport and visa.
- Send email to requestor and DM after a request is submitted.
- Send email to requestor and admin head when the DM approves/rejects the request.
- Send email to respective vendors if the admin head approves the request.

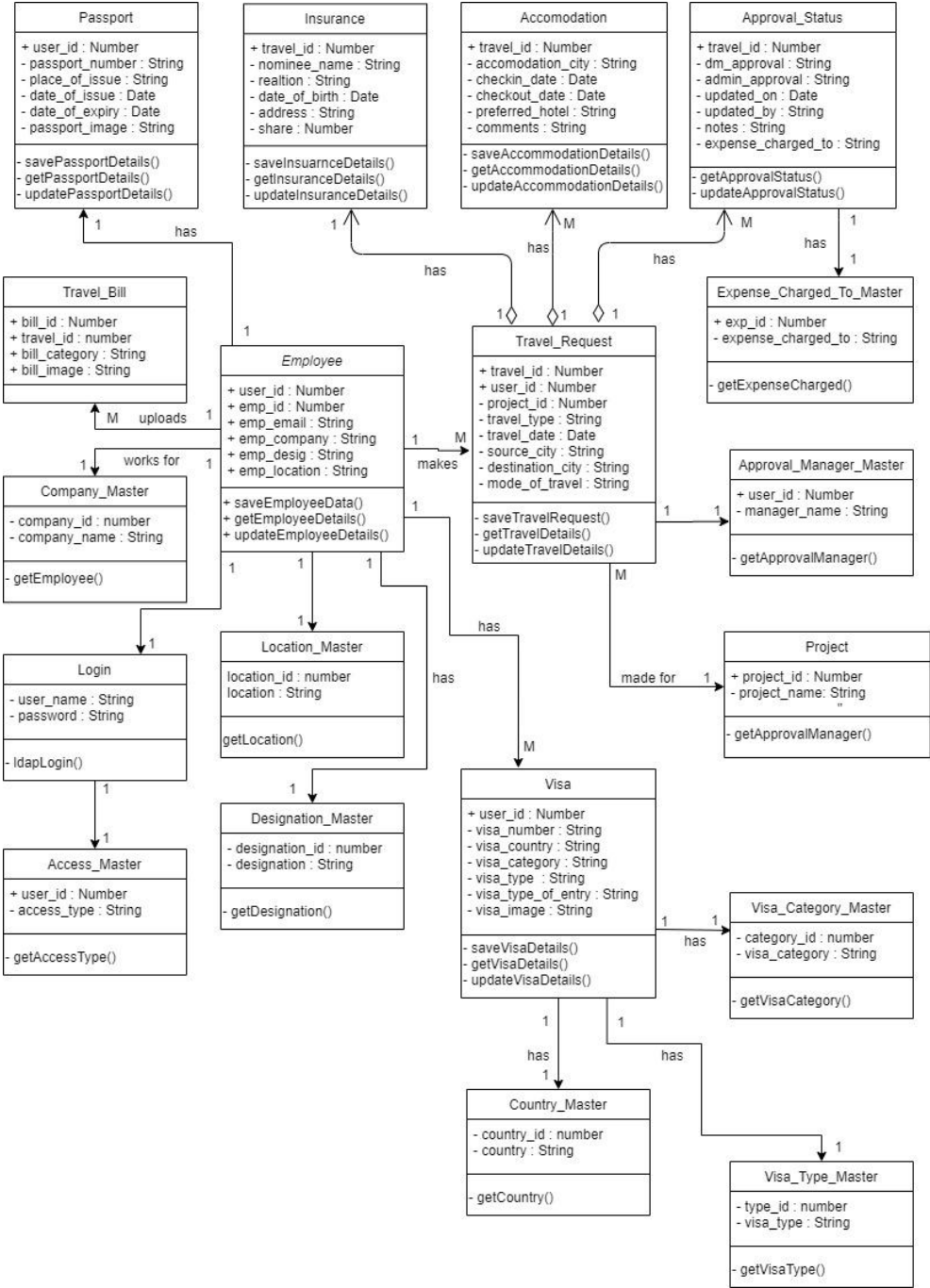
- Send email to employee 6 months before the expiry of his/her passport.
- Employee should be able to track the status of his/her travel request.
- Employee should be able upload bills associated with his/her trip after the trip is completed.
- The Delivery Manager and Admin head should be able to approve, reject, and send the travel request back to the employee for correction.

CHAPTER 3
ANALYSIS & DESIGN

3.1 Object Diagram

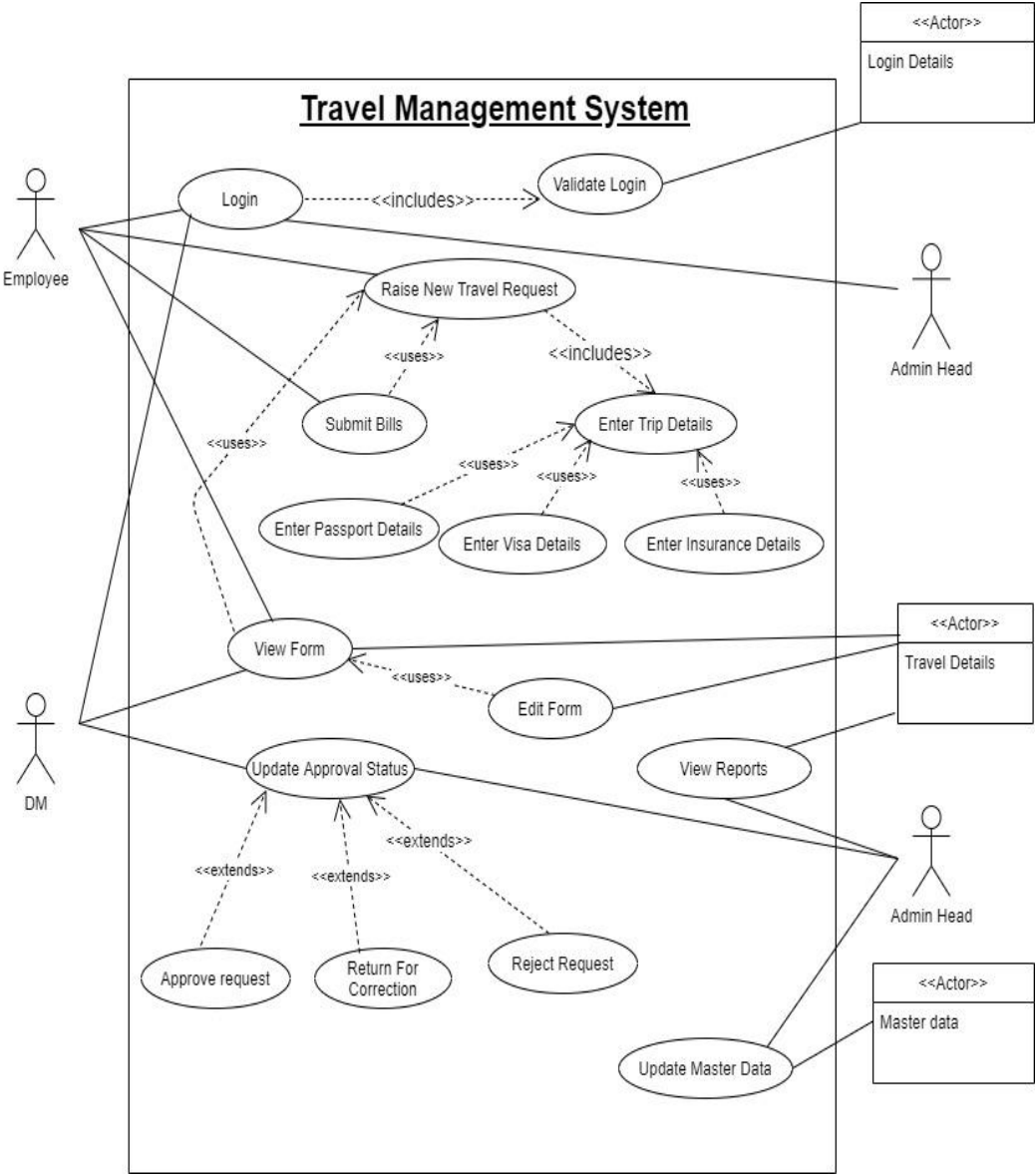


3.2 Class Diagram

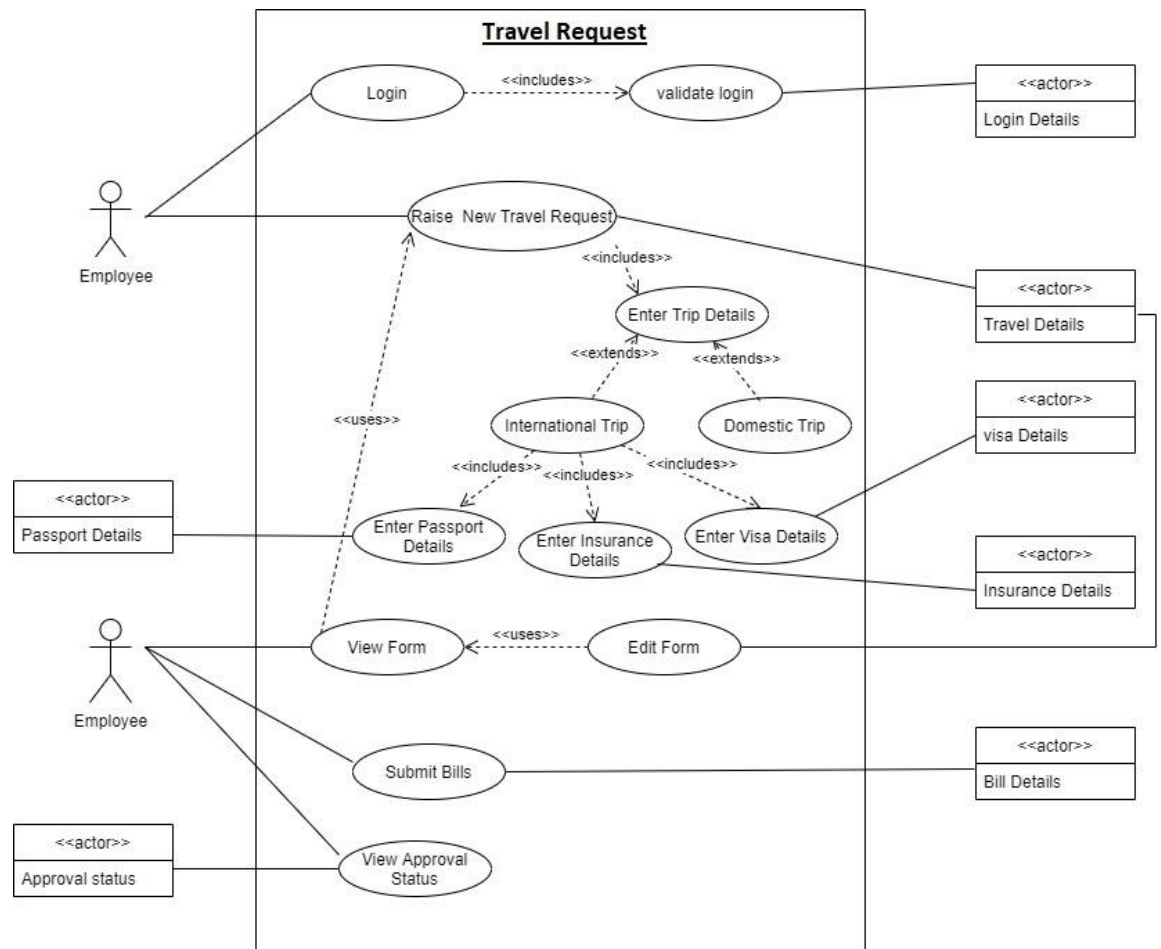


3.3 Use Case Diagrams

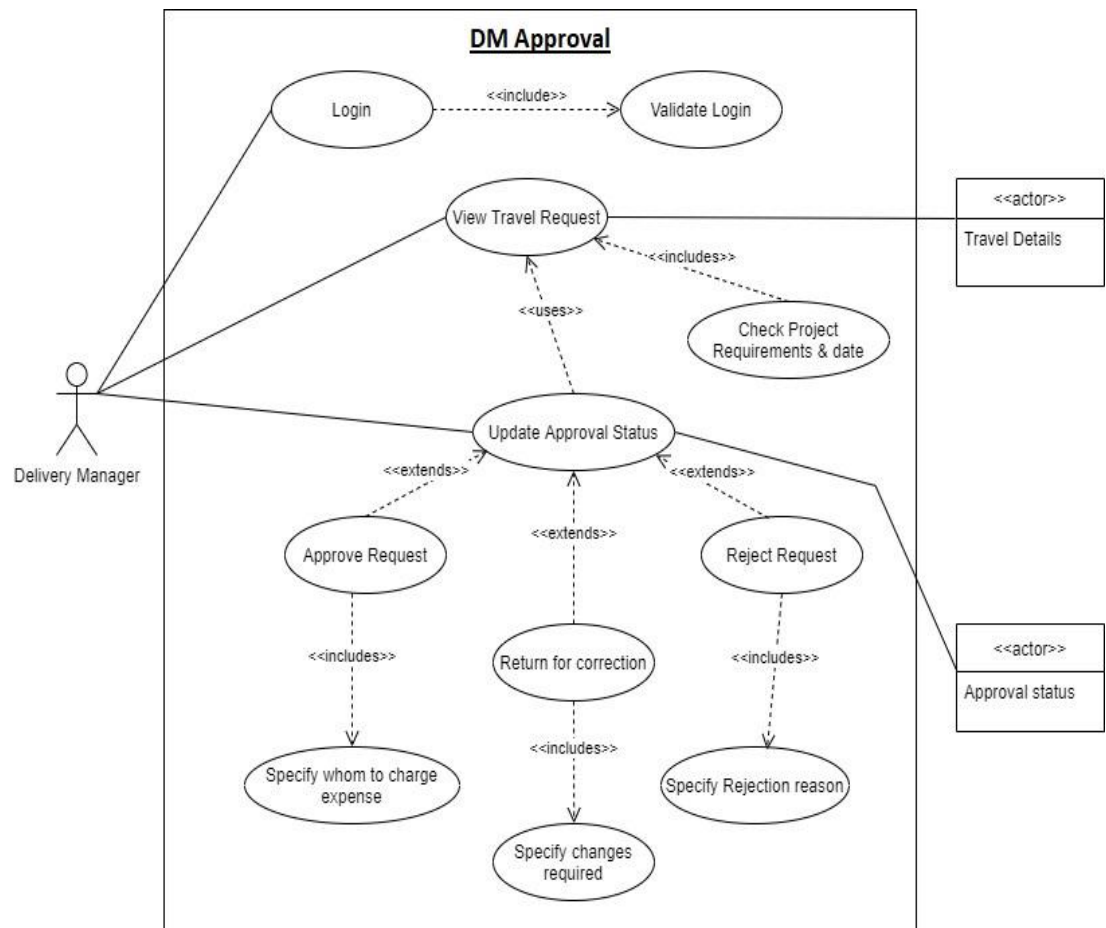
1) Main use-case diagram



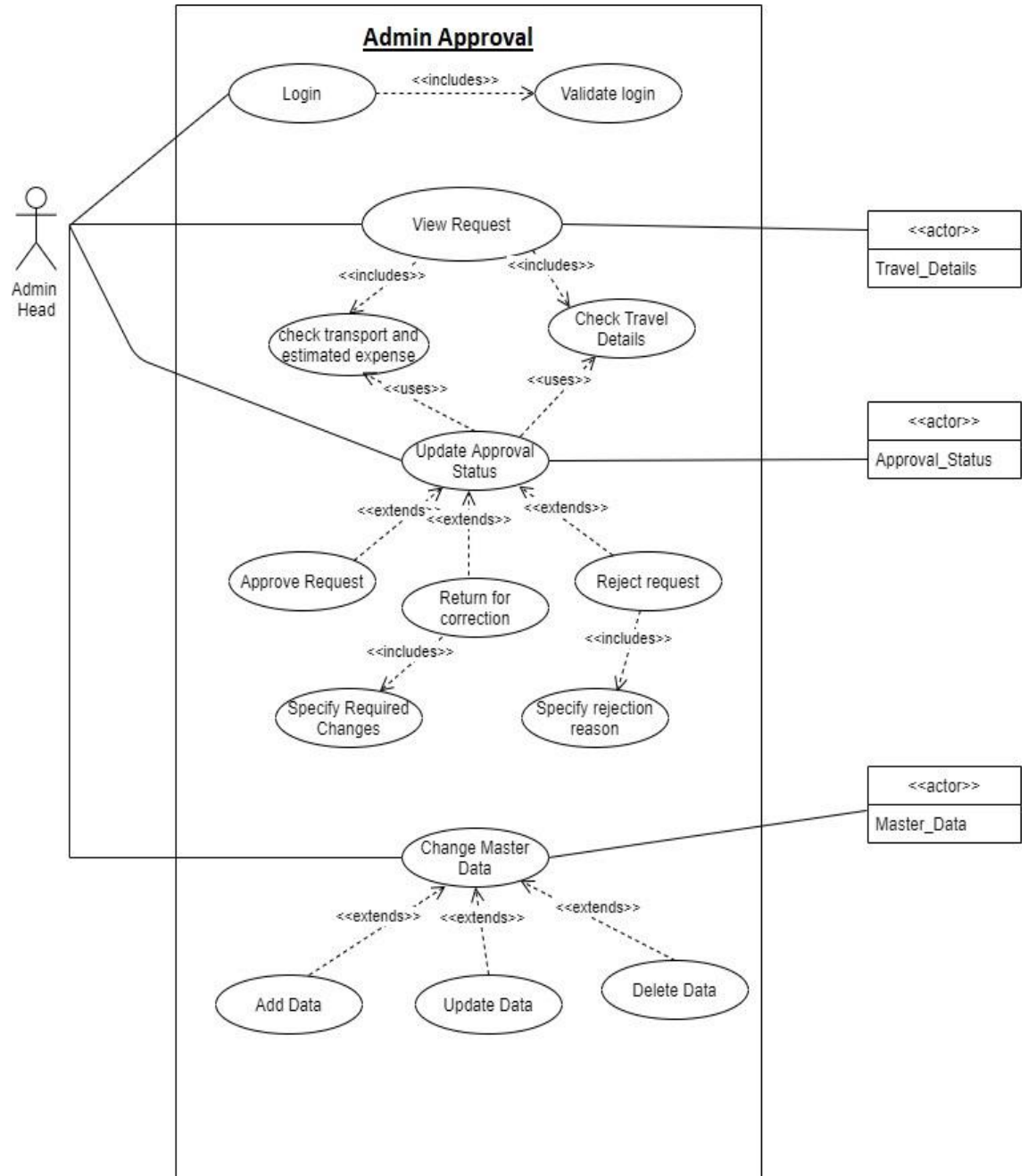
2) Travel Request



3) DM Approval

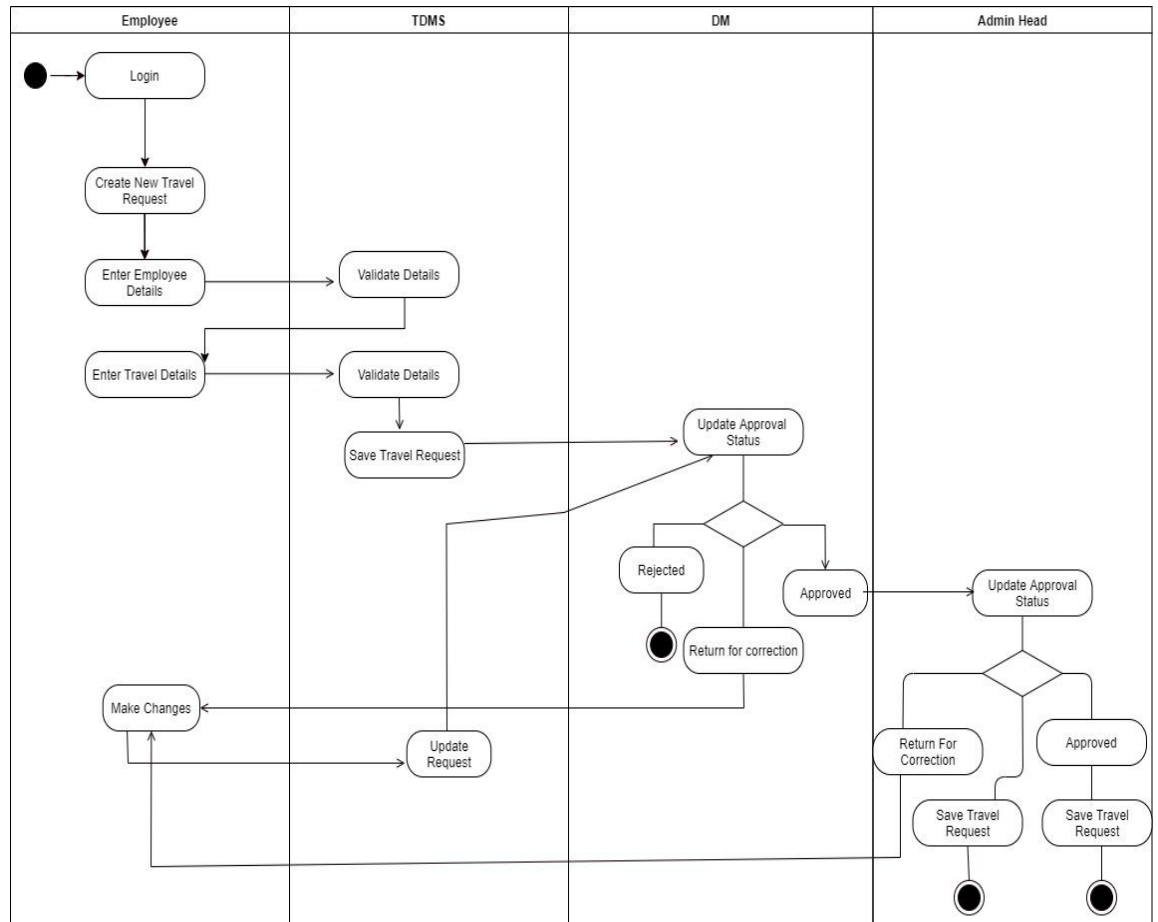


4) Admin Head Approval

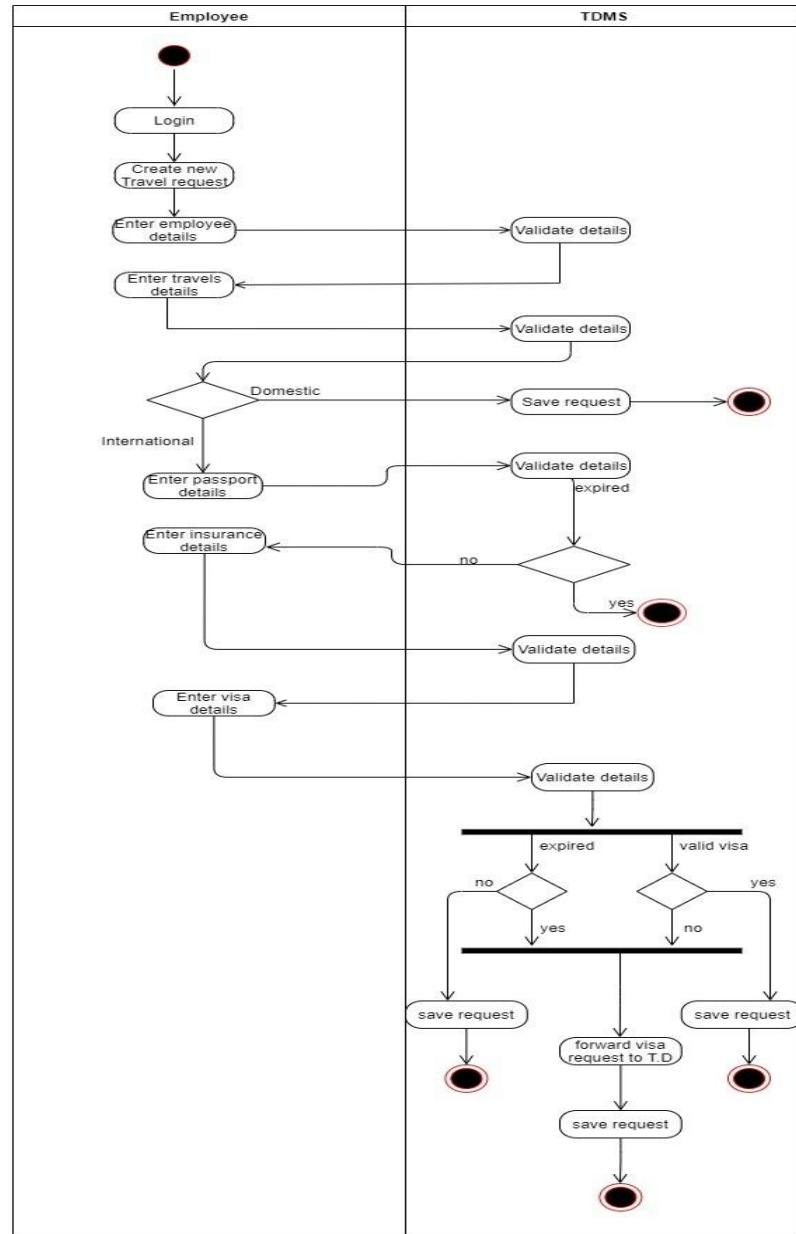


3.4 Activity Diagram

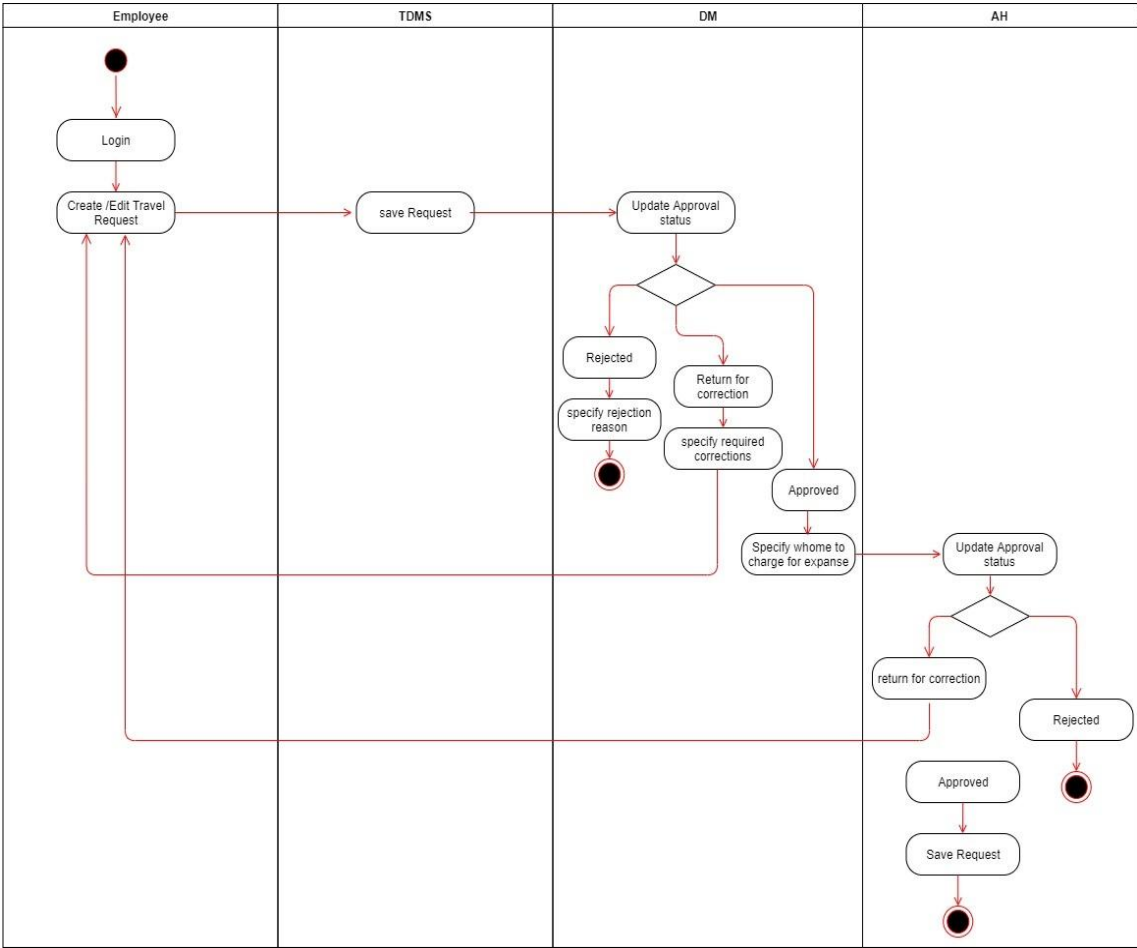
1) Main



2) Travel Request

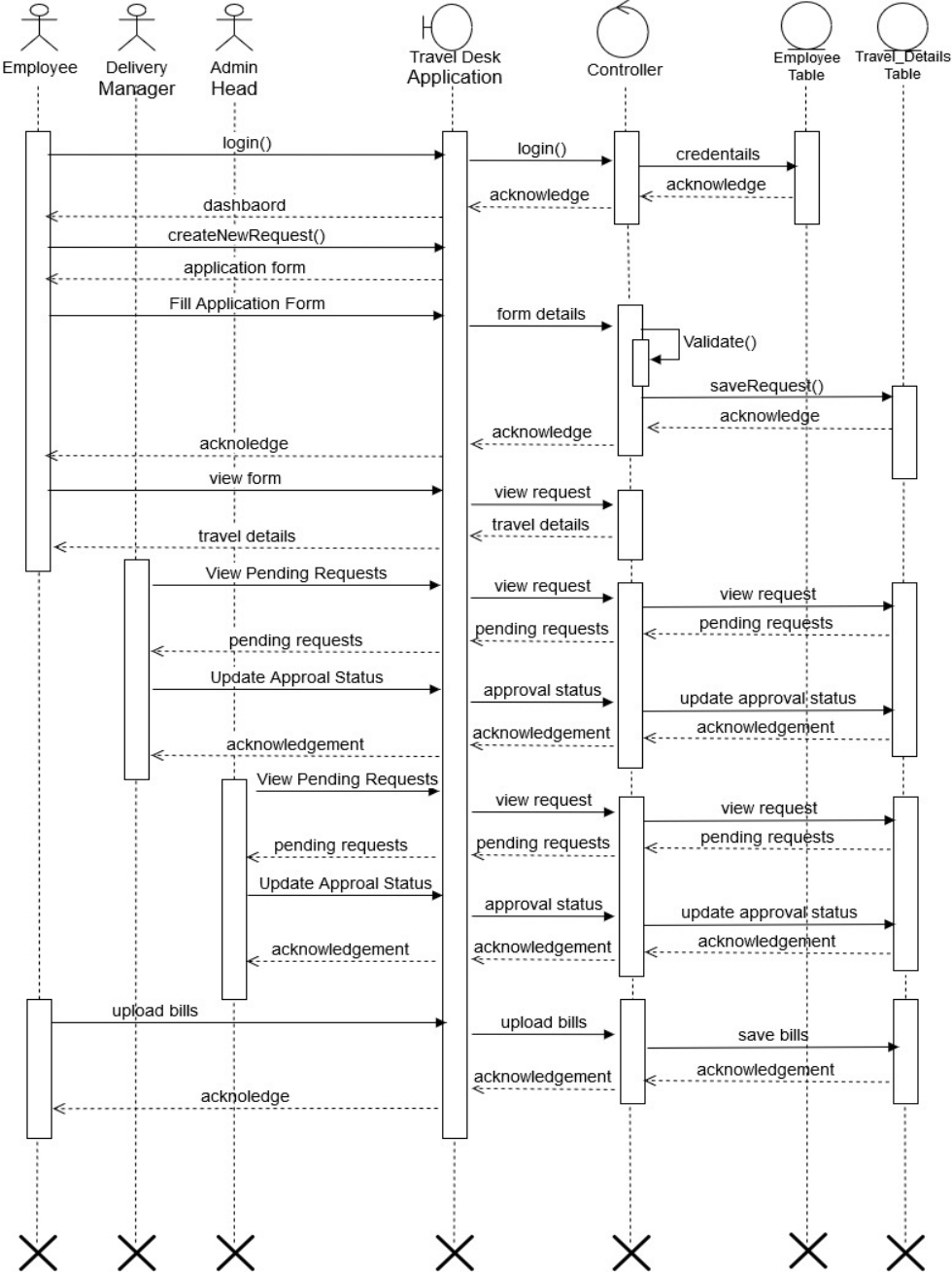


3) Request Approval

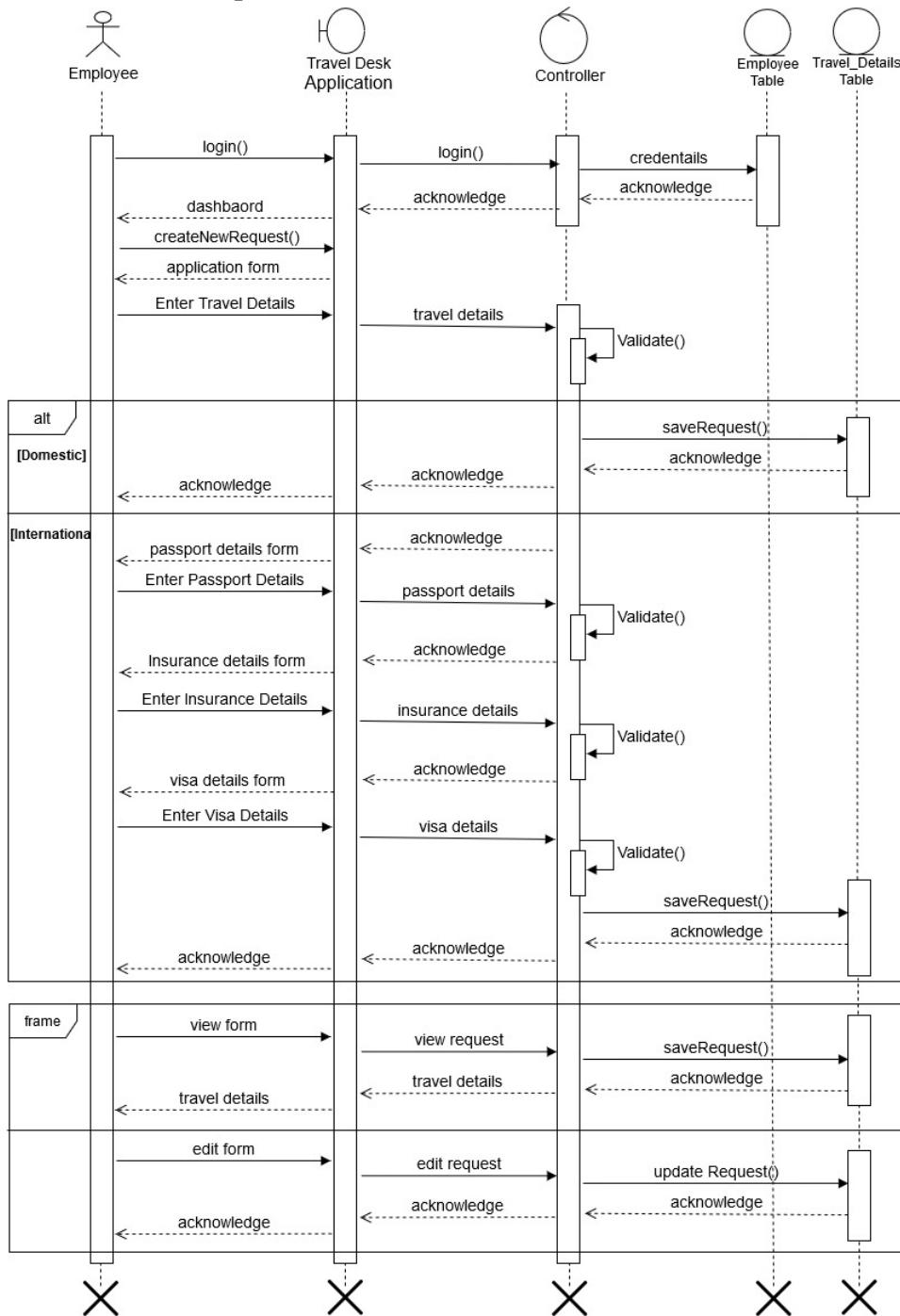


3.5 Sequence Diagrams

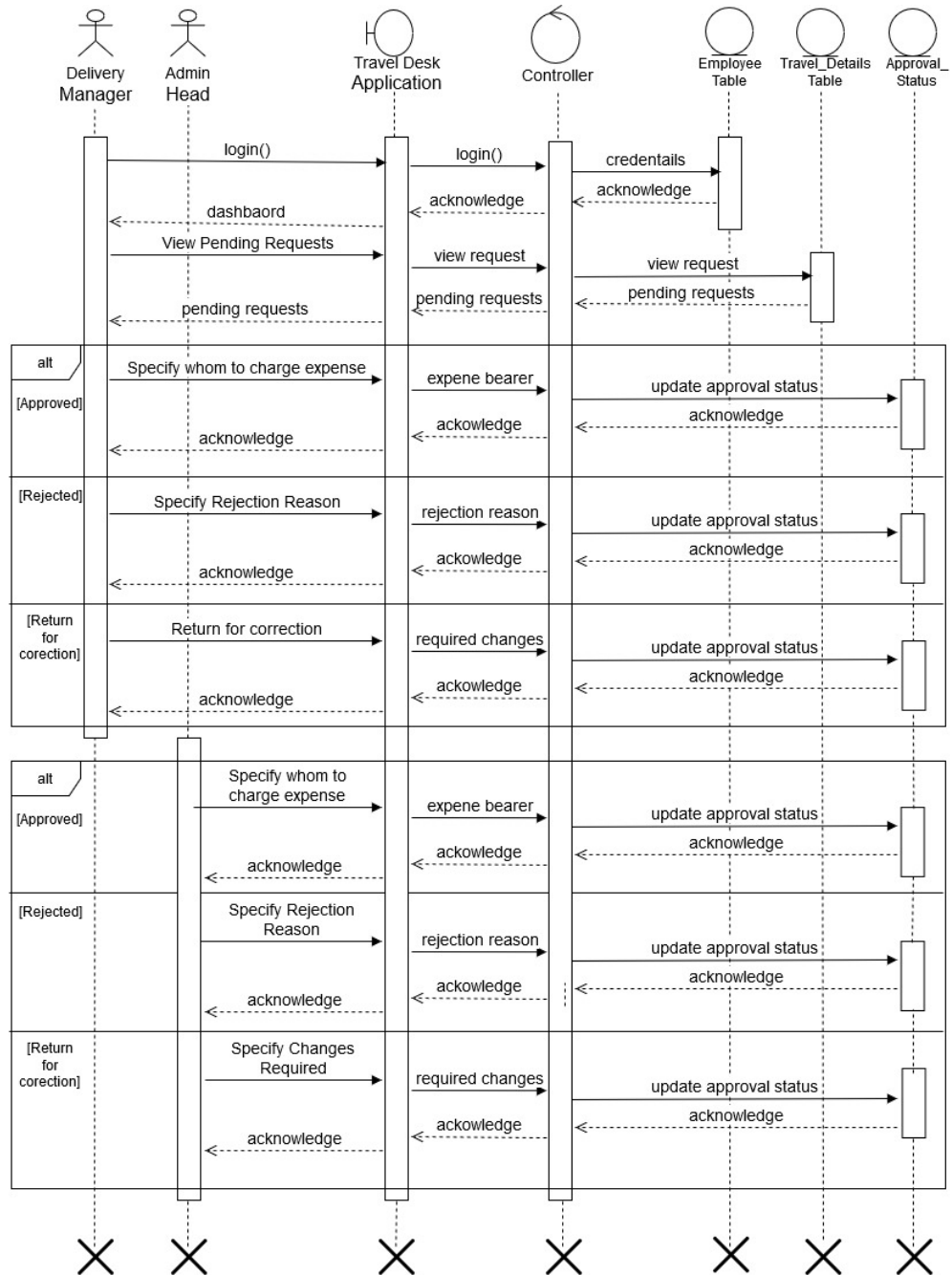
1) Main



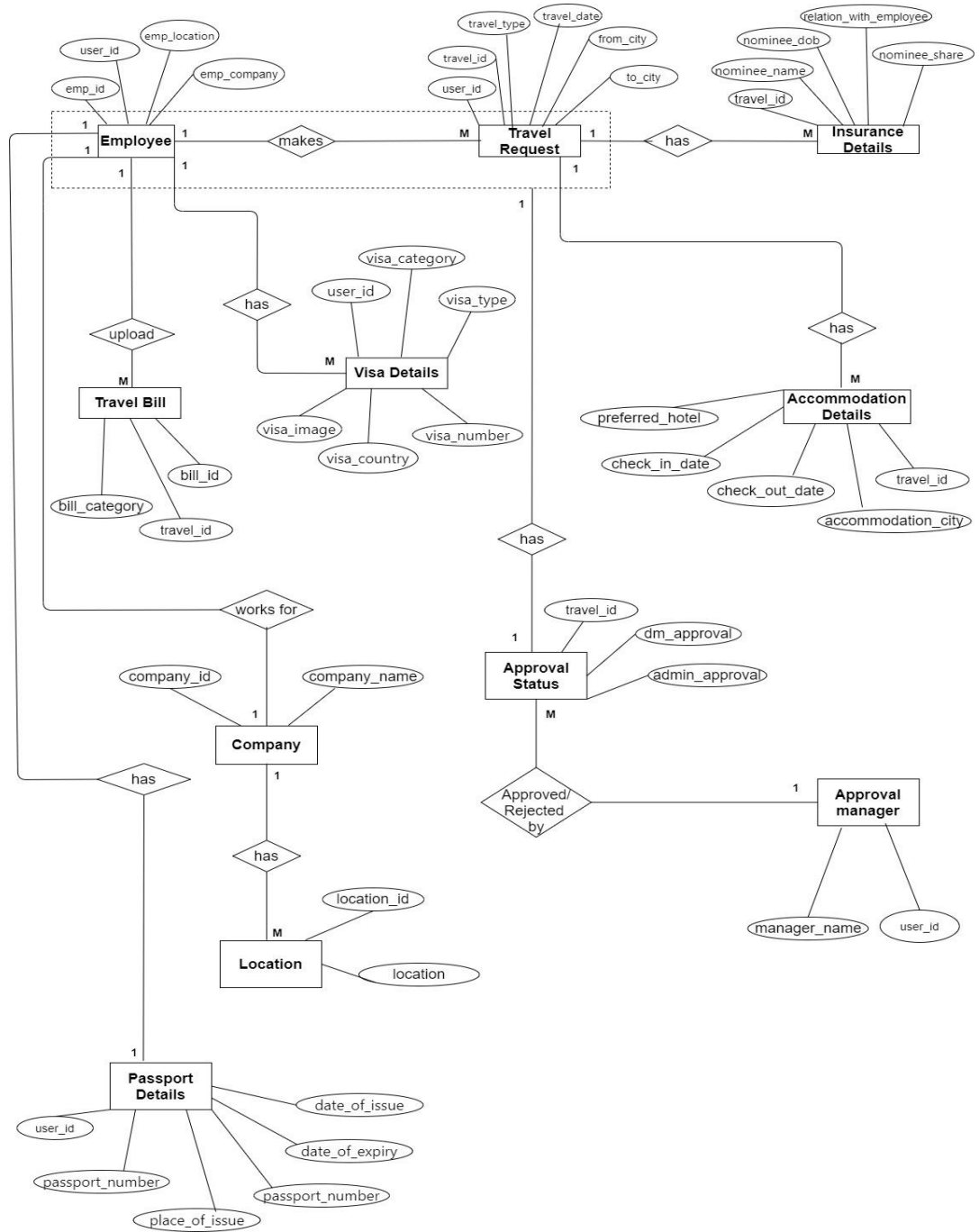
2) Travel Request



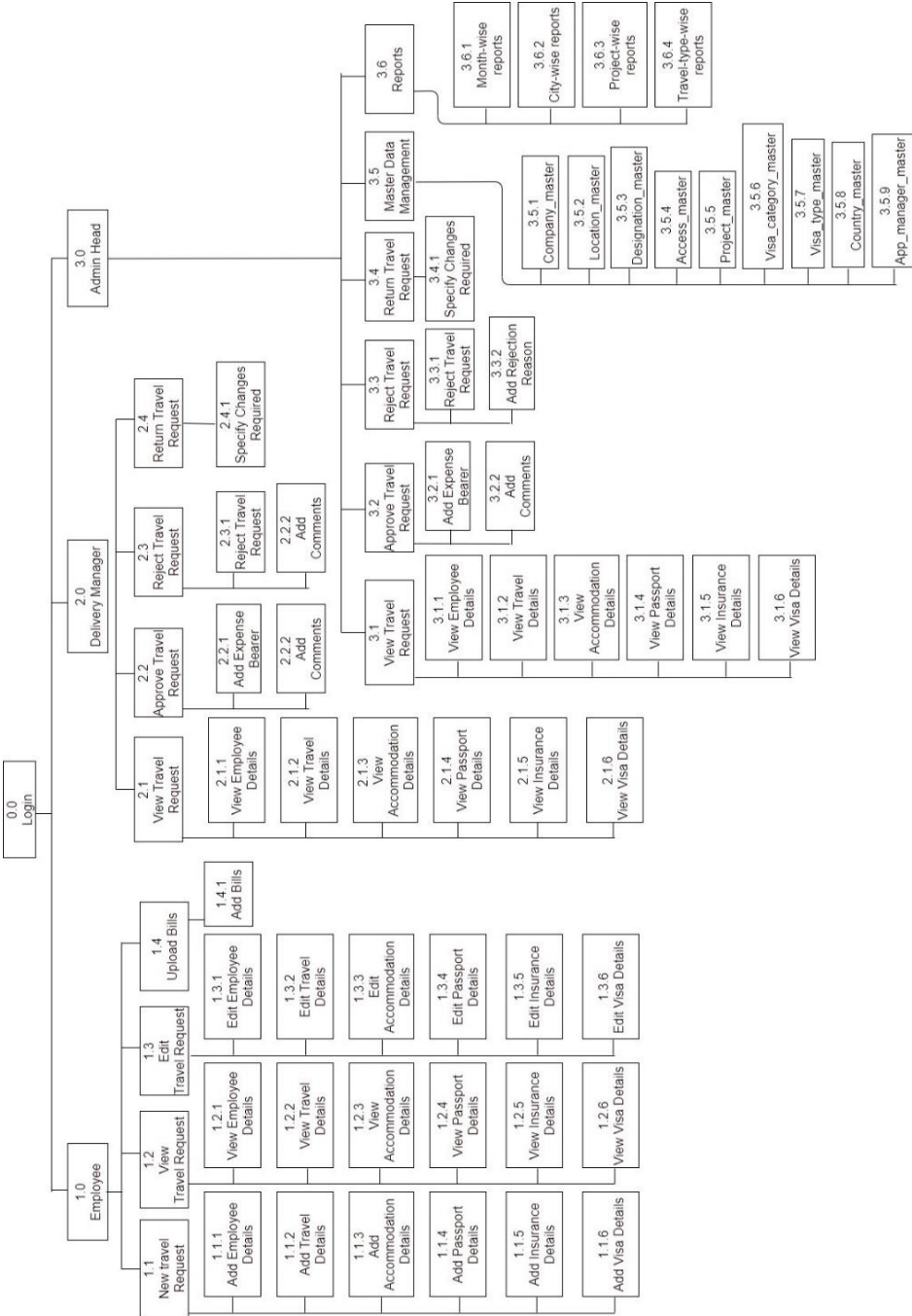
3) Request Approval



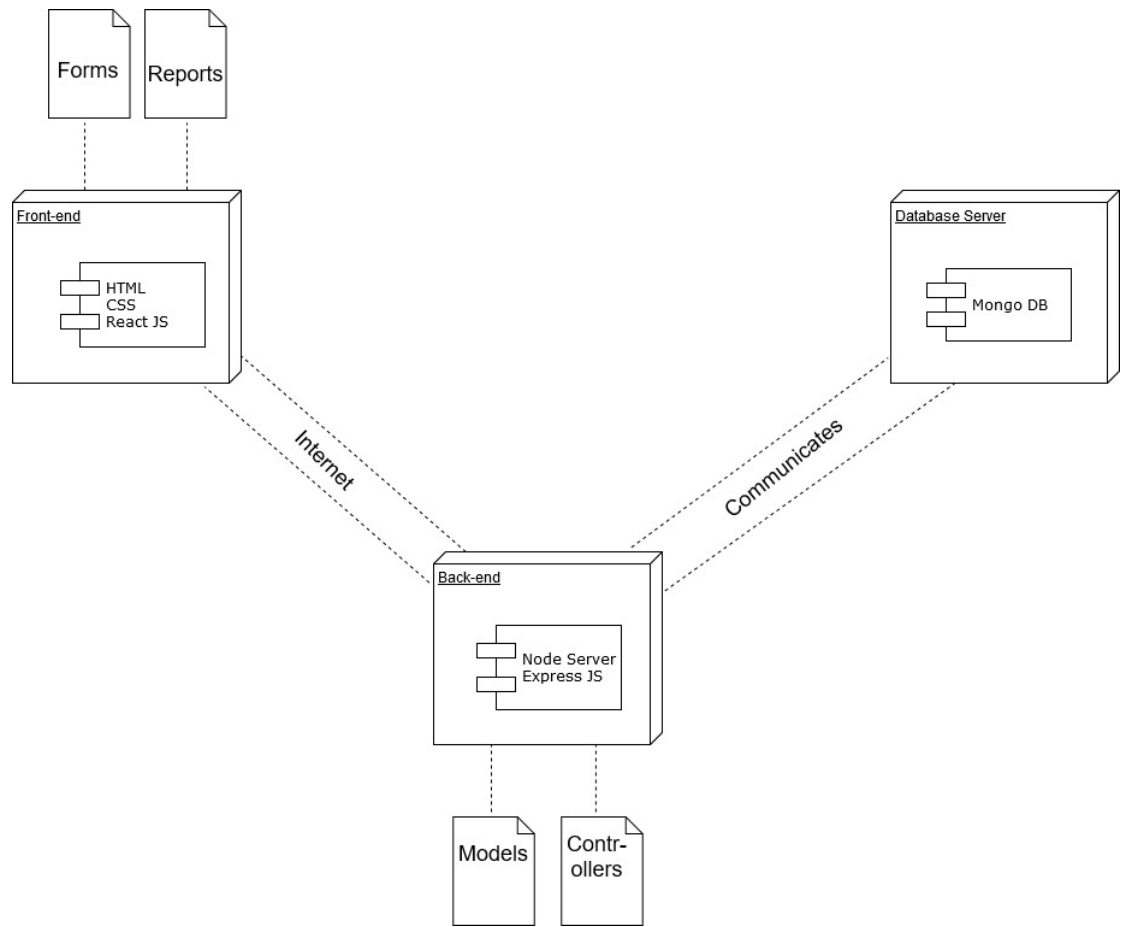
3.6 Entity Relationship Diagram



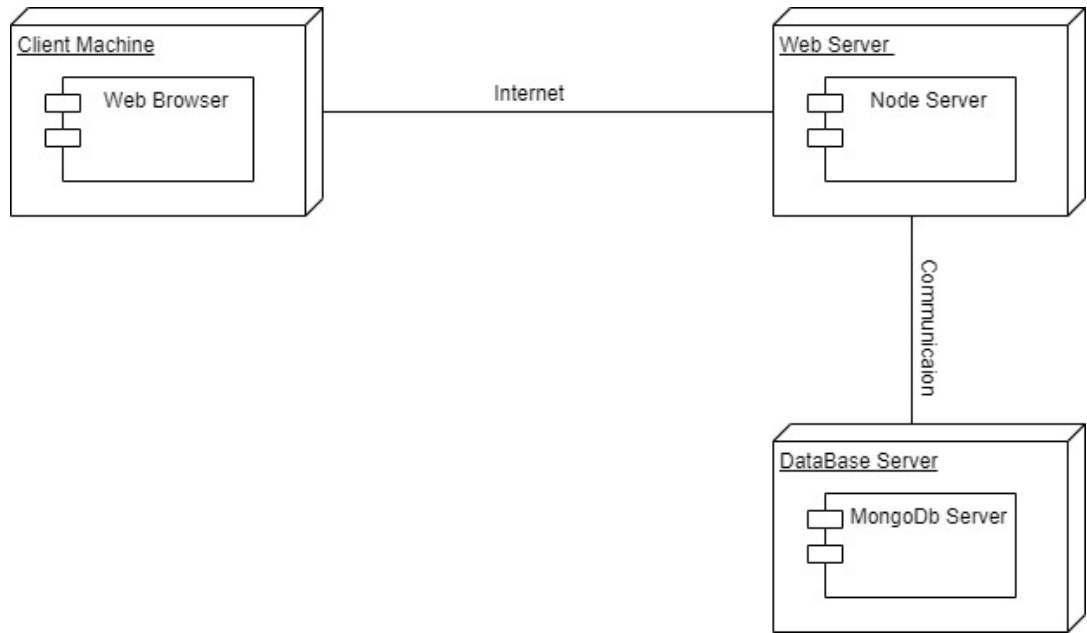
3.7 Module Hierarchy Diagram



3.8 Component Diagram



3.9 Deployment Diagram



3.10 Module Specifications

The Travel Management System is divided into following modules :

1) User Authentication

For authentication(login) purpose, the web application will use LDAP - SSO(Single Sign-On) feature, meaning that the employee can use the same login credentials as he/she uses for logging in to other applications of the same company.

For e.g. Using the same credentials of one Google account, we can login to youtube, google playstore, duo, google maps, and other google applications.

Based on the user id returned from the database, the employee login will be categorized as Normal Employee login, Delivery Manager login, or Admin Head Login.

2) Travel Request Management

When an employee logs in, he/she will be able to see a dashboard of his/her travel requests. The dashboard will be blank if the employee hasn't made any travel request in the past.

Employee will have an option to raise a new travel request. In the travel request he/she will have to fill the following details :

Travel Details –

- o Trip type
- o Travel type
- o Source city
- o Destination city
- o Departure date
- o Return date
- o Mode of travel

- o Meal Preference
- o Frequent Flyer Number
- o Accommodation requirement,etc

Passport Details (Applicable for International Travel) –

- o Passport Number
- o Place of issue
- o Date of issue
- o Date of expiry
- o Upload scanned copy

Visa Details (Applicable for International Travel) –

- o Visa number
- o Visa country
- o Visa category

- o Visa type
- o Valid from
- o Valid till
- o Type of entry
- o Additional visa

Insurance Details –

- o Nominee name
- o Relationship with employee
- o Date of birth
- o Share to be paid
- o Address

After filling these details and submitting the request, all the details will be stored in

the database and an acknowledgement email will be sent to the employee.

Also, email will be sent to his/her Delivery Manager informing about the new

request and ask to approve or reject the request.

The employee dashboard will now have the request details that he/she has made

along with its approval status.

Options will be given to view or edit the details he/she has filled.

Employee will be able to reschedule his/her trip.

Also, after the trip he/she can upload all the bills associated with his/her tour.

3) User Profile Management -

The user profile will be linked to a SAP application of the company. However, the basic profile details can be updated through Travel Desk Management System too.

The user can view and update the following details :

- 1) User name
- 2) Profile picture
- 3) Date of birth
- 4) Email ID
- 5) Company extension
- 6) Address
- 7) Account Password

If the user makes changes in any of the above fields, then those changes will be reflected in the SAP application's database too.

4) Approval Status -

The status of a travel request, when created, is by default set to “pending”. This status can be updated by Delivery Manager and Admin Head.

Both, Delivery Manager and Admin Head, can view the travel requests in the “pending requests” tab of their dashboard.

In the pending requests tab, the details of the basic travel request will be displayed of each request that is received. Option will be given to view the full travel request details that an employee has filled.

After reviewing the details, the Delivery Manager can either approve or reject the request. Before approving, an option will be given to select whom to charge the travel expense, as sometimes the company spares the expense, while sometimes client does it.

If the request is to be rejected, then the DM can specify the reason for rejection. If the request is approved then an email is sent to the

requestor informing him/her the request status and another email to admin head informing him/her about the request and ask for approval.

The Delivery Manager can also send the travel request back to the employee for corrections.

Only after the request is approved by the Delivery Manger, the Admin head can approve, reject or return request for correction.

If the request is approved then an email is sent to the requestor informing him/her the request status.

After the request is approved by admin head, if the employee is not having his/her own business cards, then request email will be sent to the card printing vendor along with the required details of the employee. Also, appropriate emails will be sent to transport, accommodation and other vendors for booking requests.

5) Master Data Management -

The master data in the database is mainly used for populating the components on the travel request form.

Only the admin head will have the access to view the master data table and view, add, update and delete data in them.

6) Reports -

Based on the data stored in the database related to travel requests, reports will be generated based on various categories.

The reports can only be viewed by the Admin Head.

The reports are of two types -

1) Table-format Reports

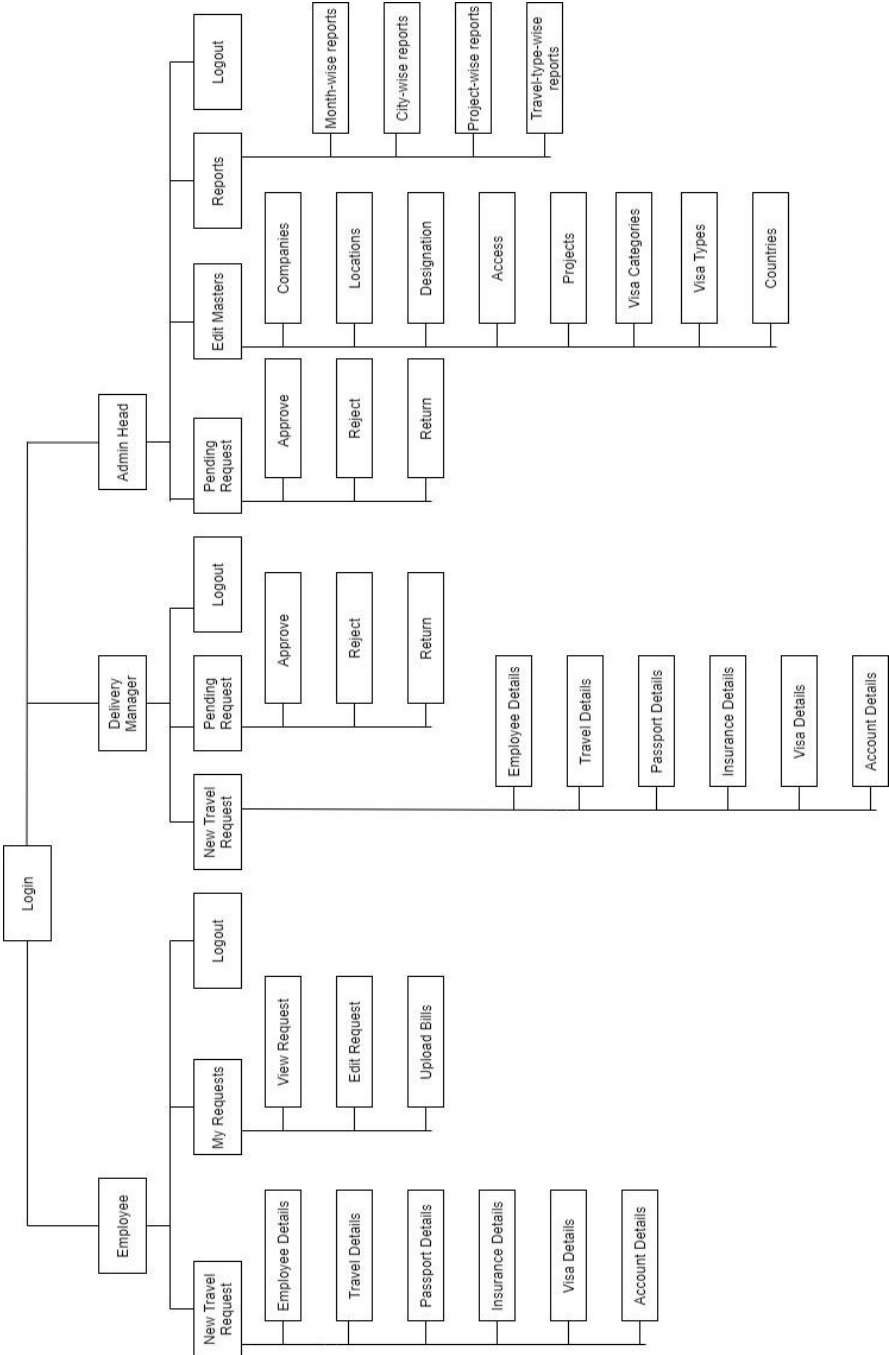
2) Graph-format Reports

The reports can be viewed based on the following categories :

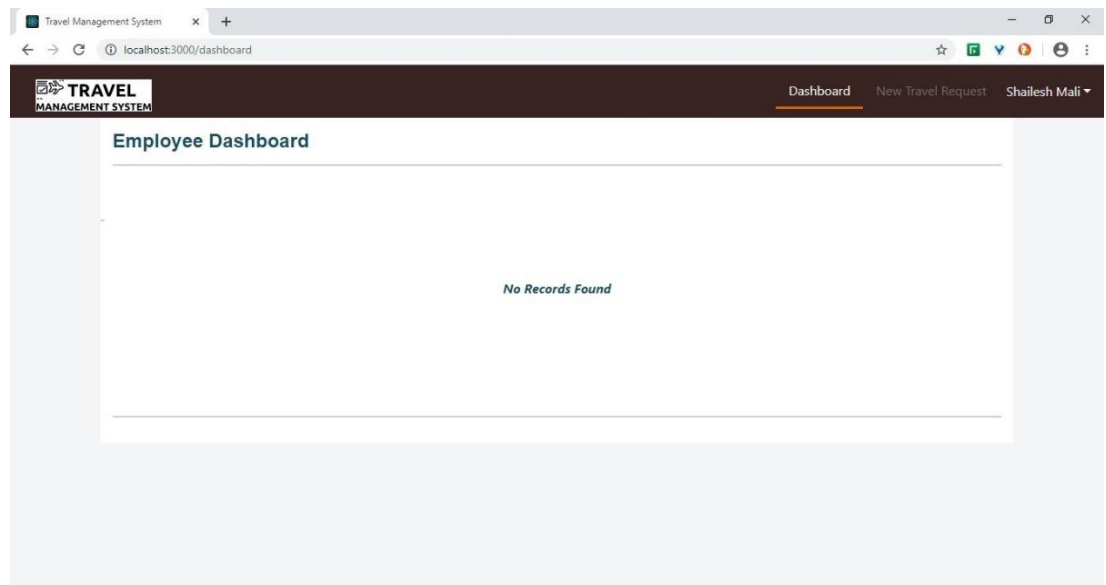
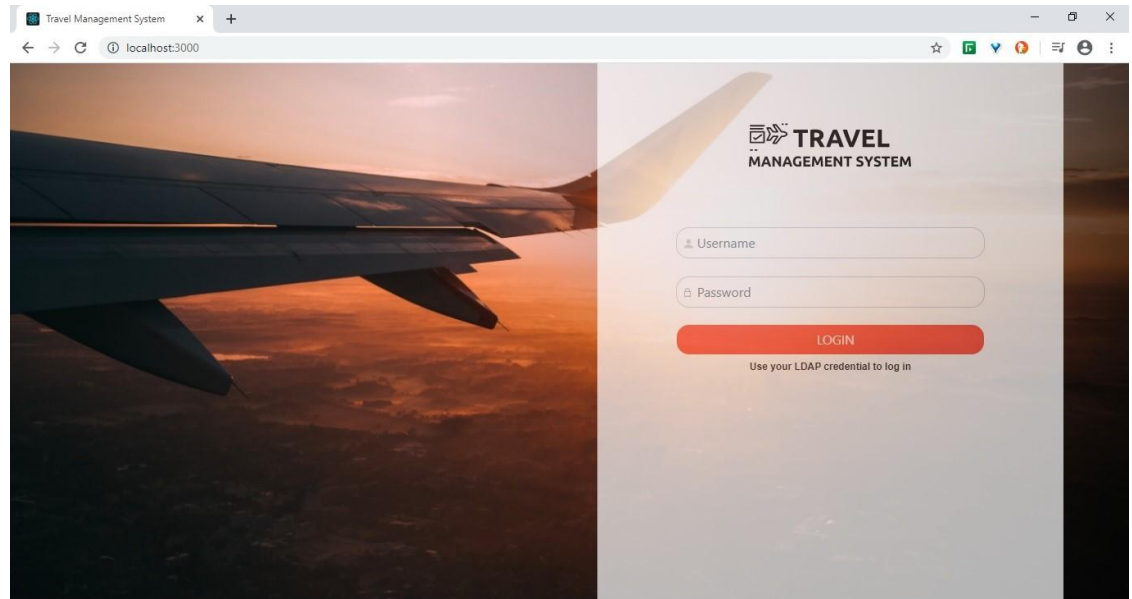
- 1) Month-wise Travel Requests report
- 2) City-wise Travel Requests report
- 3) Project-wise Travel Requests report
- 4) Travel-type-wise Travel Requests report

All these reports can be viewed either in table format or graph format.

3.12 Web Site Map Diagram



3.13 User Interface Design



Travel Management System x +

localhost:3000/travelrequest

Dashboard **New Travel Request** Shailesh Mali

New Travel Request

Employee Master Travel Details Passport Details Insurance Details Visa Details

EMPLOYEE MASTER

*Company Name

*Employee Name

*Employee ID

*Gender Male Female

*Mobile Number

*Designation

*Employee's Location

Specify Location

*Project Name

*Approval Manager

*Date of Birth

*Do you have Business Cards?

Next

Travel Management System x +

localhost:3000/travelrequest

Dashboard **New Travel Request** Shailesh Mali

New Travel Request

Employee Master Travel Details Passport Details Insurance Details Visa Details

TRAVEL DETAILS

Round Trip Multi City

*Travel Type

*Mode of Travel

*Booking Through

*Project name

*Client Address

*Travel Date

OTHER DETAILS

*Seat Preference

*Meal Preference

*Frequent Flyer Number

*Preferred Time

ACCOMMODATION

Travel Management System x +
localhost:3000/travelrequest

Select [] Select []

*Project name: Travel Desk Management
*Client Address: []
*Travel Date: dd/mm/yyyy

OTHER DETAILS

*Seat Preference: [] *Meal Preference: Select [] *Frequent Flyer Number: [] *Preferred Time: []

ACCOMMODATION

*Accommodation Needed: Yes No
Place to Visit: Enter Place to Visit []
Any Preferred Hotel? Provide Details: []

Check In Date: dd/mm/yyyy
Check out Date: dd/mm/yyyy
Comments If Any: []

Back Next

Travel Management System x +
localhost:3000/travelrequest

TRAVEL MANAGEMENT SYSTEM Dashboard New Travel Request Shailesh Mali

New Travel Request

Employee Master Travel Details **Passport Details** Insurance Details Visa Details

PASSPORT DETAILS

*Has a Passport: Yes []
*Passport Number: A87364872
*Date of Expiry: 24/12/2020

Upload Files

Upload Passport Front Image
Choose file [] Browse []

Upload Passport Back Image
Choose file [] Browse []

Cancel Upload

Back Next

Please refer following links for Harbinger Travel Policy Domestic | International

Travel Management System x +

localhost:3000/travelrequest

New Travel Request

Employee Master | Travel Details | Passport Details | Insurance Details | **Visa Details**

VISA DETAILS (Applicable only for International Travel)

*Valid Visa Yes No

*Upload Visa

*Visa Number

*Country for which visa required

*Visa Category

*Visa Type

*Valid From

*Valid Till

*Intended Date Of Travel

*Type of Entry

*Mention If Additional Visa Available

*Total Work Experience

*Total Harbinger Experience

*Current Project Total Experience

Please refer following links for Harbinger Travel Policy [Domestic](#) | [International](#)

Travel Management System x +

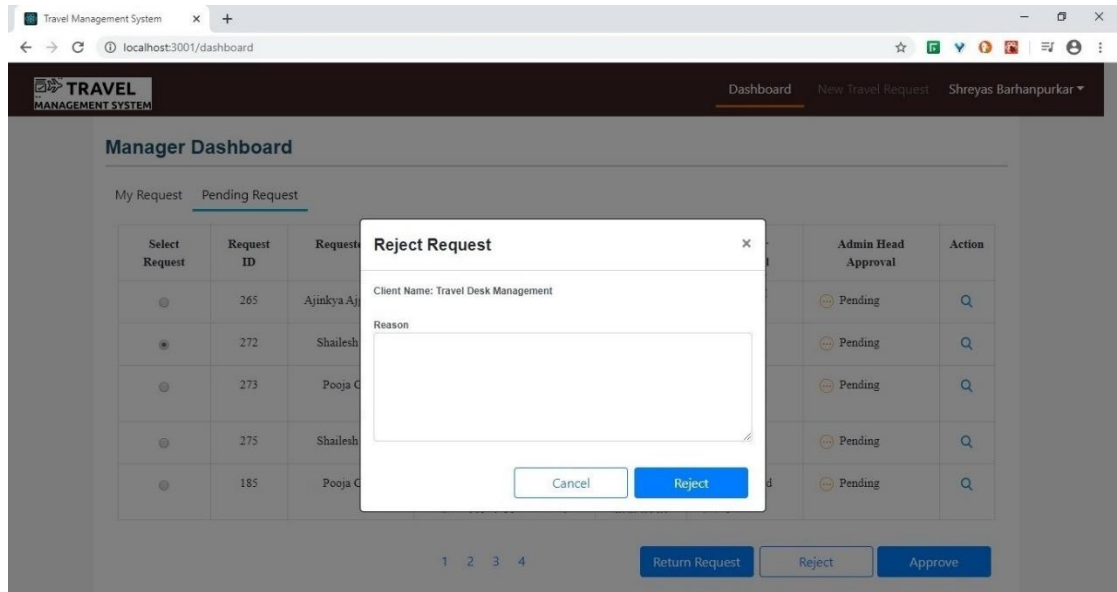
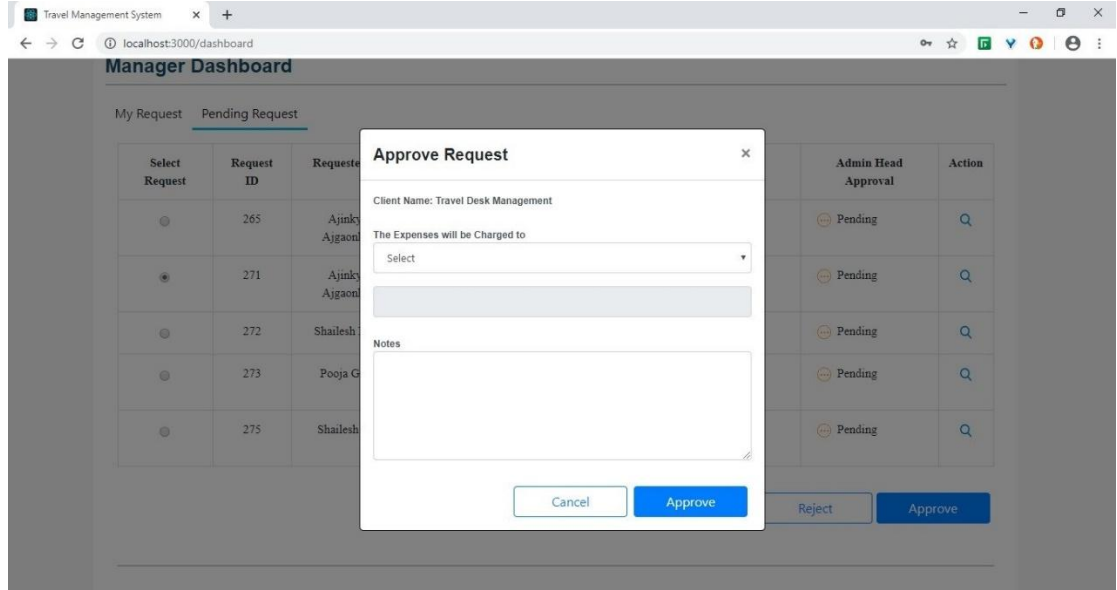
localhost:3000/uploadBills

TRAVEL MANAGEMENT SYSTEM Dashboard | New Travel Request | Shreyas Barhanpurkar

Upload Bills

Travel ID : 173

From	To	Departure
<input type="text" value="Pune"/>	<input type="text" value="Mumbai"/>	<input type="text" value="02/11/2020"/>
<input type="text" value="Mumbai"/>	<input type="text" value="Bangalore"/>	<input type="text" value="02/18/2020"/>



Travel Management System x +

localhost:3001/dashboard

Dashboard New Travel Request Shreyas Barhanpurkar

Manager Dashboard

My Request Pending Request

Select Request	Request ID	Request	Admin Head Approval	Action
<input type="radio"/>	265	Ajinkya A	Pending	Q
<input checked="" type="radio"/>	272	Shailesh	Pending	Q
<input type="radio"/>	273	Pooja G	Pending	Q
<input type="radio"/>	275	Shailesh	Pending	Q
<input type="radio"/>	185	Pooja G	Pending	Q

Return Request for Correction

Client Name: Travel Desk Management

Changes to be made

1 2 3 4

Travel Management System x +

localhost:3000/changeMasterData/edit/add

Dashboard New Travel Request Edit Masters Reports Dinesh Lad

Designation_Master

ADD DATA

*Enter Designation to Add

3.14 Data Dictionary

Sr. No	Field Name	Data-type	Width	Description	Table Name
1	access_type	String	15	access type	Access_Master
2	accommodation_comments	String	100	accommodation comments	Accommodation
3	accommodation_needed	String	3	yes/no	Accommodation
4	checkin_date	Date	-	checkin date	Accommodation
5	checkout_date	Date	-	checkout date	Accommodation
6	place_to_visit	String	20	place to visit	Accommodation
7	preferred_hotel	String	20	preferred hotel	Accommodation
8	approval_manager	String	50	approval manager	Approval_Manager_Master

9	admin_approval_status	String	20	admin approval status	Approval_Status
10	dm_approval_status	String	20	dm approval status	Approval_Status
11	notes	String	100	notes	Approval_Status
12	rejection_reason	String	50	rejection reason	Approval_Status
13	request_created_by	String	20	request created by	Approval_Status
14	request_created_on	Date	-	request created on	Approval_Status
15	request_updated_by	String	20	request updated by	Approval_Status
16	request_updated_on	Date	-	request updated on	Approval_Status
17	bill_category	String	20	bill category	Travel_Bill
18	bill_id	Number	5	bill id	Travel_Bill

19	bill_image	String	50	bill image	Travel_Bill
20	company_id	Number	5	company id	Company_Mas ter
21	country	String	20	country	Country_Mast er
22	country_id	Number	5	country id	Country_Mast er
23	designation_id	Number	5	designation id	Employee, Designation_ Master
24	business_card	String	3	yes/no	Employee
25	company_name	String	50	company name	Employee
26	designation	String	20	designation	Employee
27	dob	Date	-	date of birth	Employee
28	emp_location	String	50	employee location	Employee
29	employee_email	String	50	employee email	Employee

30	employee_id	String	10	employee id	Employee
31	employee_name	String	50	employee name	Employee
32	gender	String	10	gender	Employee
33	mobile	Number	10	mobile	Employee
34	user_id	Number	5	user identification number	Employee, Access_Master, Passport, Visa, Travel_Request, Approval_Manager_Master
35	nominee_address	String	100	nominee address	Insurance
36	nominee_dob	Date	-	nominee dob	Insurance
37	nominee_name	String	20	nominee name	Insurance
38	nominee_share	Number	3	nominee share	Insurance

39	relation_with_employee	String	20	relation with employee	Insurance
40	location_id	Number	5	location id	Location_Master
41	password	String	20	password	Login
42	user_name	String	50	user name	Login
43	has_passport	String	3	yes/no	Passport
44	passport_backimg	String	50	passport back image	Passport
45	passport_date_of_expiry	Date	-	passport date of expiry	Passport
46	passport_date_of_issue	Date	-	passport date of issue	Passport
47	passport_frontimg	String	50	passport front image	Passport
48	passport_number	String	9	passport number	Passport

49	passport_place_of_issue	String	20	passport place of issue	Passport
50	project_id	Number	5	project id	Project_Master
51	project_name	String	50	project name	Project_Master
52	approval_manager	String	50	approval manager	Travel_Request
53	booking_through	String	15	booking through	Travel_Request
54	client_address	String	50	client address	Travel_Request
55	departure	Date	-	departure date	Travel_Request
56	expense_charged_to	String	50	expense charged to	Travel_Request
57	frequent_flyer_number	String	10	frequent flyer number	Travel_Request
58	from_city	String	20	source city	Travel_Request

59	meal_preference	String	20	meal preference	Travel_Request
60	mode_of_travel	String	20	mode of travel	Travel_Request
61	preferred_time	String	10	preferred time	Travel_Request
62	return	Date	-	return date	Travel_Request
63	seat_preference	String	20	seat preference	Travel_Request
64	to_city	String	20	destination city	Travel_Request
65	travel_date	Date	-	travel date	Travel_Request
66	travel_type	String	15	type of travel	Travel_Request
67	trip_type	String	15	trip type	Travel_Request
68	travel_id	Number	5	travel id	Travel_Request, Insurance, Accommodation, Approval_S

					tatus
69	additional_visa_available	String	3	addition visa available	Visa
70	intended_date_of_travel	Date	-	intended date of travel	Visa
71	total_harbinger_experience	String	10	total harbinger experience	Visa
72	total_project_experience	String	10	total project experience	Visa
73	total_work_experience	String	10	total work experience	Visa
74	type_of_entry	String	10	type of entry	Visa
75	valid_from	Date	-	valid from	Visa
76	valid_till	Date	-	valid till	Visa
77	valid_visa	String	3	yes/no	Visa
78	visa_category	String	20	visa	Visa

				category	
79	visa_country	String	20	visa country	Visa
80	visa_image	String	50	visa image	Visa
81	visa_number	String	10	visa number	Visa
82	visa_type	String	20	visa type	Visa
83	visa_category_id	Number	5	visa category id	Visa_Category_Master
84	visa_type_id	Number	5	visa type id	Visa_Type_Master
85	email_id	String	50	email ids for sending email	Email_Master
86	category	String	20	email category	Email_Master

3.15 Table specifications

1) Passport

Sr. No.	Field Name	Data Type	Width	Constraint
1	user_id	Number	5	Foreign Key
2	has_passport	String	3	Not Null
3	passport_number	String	9	-
4	passport_place_of_issue	String	20	-
5	passport_date_of_issue	Date	-	-
6	passport_date_of_expiry	Date	-	-
7	passport_frontimg	String	50	-
8	passport_backimg	String	50	-

2) Insurance

Sr. No.	Field Name	Data Type	Width	Constraint
1	travel_id	Number	5	Foreign Key
2	nominee_name	String	20	Not Null
3	relation_with_employee	String	20	Not Null
4	nominee_dob	Date	-	Not Null
5	nominee_address	String	100	Not Null
6	nominee_share	Number	3	Not Null

3) Accommodation

Sr. No.	Field Name	Data Type	Width	Constraint
1	travel_id	Number	5	Foreign Key
2	checkin_date	Date	-	-
3	checkout_date	Date	-	-

4	preferred_hotel	String	20	-
5	accommodation_comments	String	100	Not Null
6	accommodation_needed	String	3	Not Null
7	place_to_visit	String	20	-

4) Approval_Status

Sr. No.	Field Name	Data Type	Width	Constraint
1	travel_id	Number	5	Foreign Key
2	dm_approval_status	String	20	Not Null
3	admin_approval_status	String	20	Not Null
4	request_updated_on	Date	-	Not Null
5	request_updated_by	String	20	Not Null
6	request_created_by	String	20	Not Null

7	request_created_on	Date	-	Not Null
8	rejection_reason	String	50	-
9	notes	String	100	-

5) Employee

Sr. No.	Field Name	Data Type	Width	Constraint
1	user_id	Number	5	Primary Key
2	employee_id	String	10	Primary Key
3	employee_name	String	50	Not Null
4	employee_email	String	50	Not Null
5	company_name	String	50	Not Null
6	designation_id	Number	5	Foreign Key
7	Location_id	Number	5	Foreign Key
8	Gender	String	10	Not Null

9	mobile	Number	10	Not Null
10	business_card	String	3	Not Null
11	dob	Date	-	Not Null

6) Travel_Request

Sr. No.	Field Name	Data Type	Width	Constraint
1	travel_id	Number	5	Primary Key
2	user_id	Number	5	Foreign Key
3	trip_type	String	15	Not Null
4	travel_type	String	15	Not Null
5	travel_date	Date	-	Not Null
6	from_city	String	20	-
7	to_city	String	20	-
8	mode_of_travel	String	20	Not Null

9	booking_through	String	15	Not Null
10	departure	Date	-	-
11	client_address	String	50	Not Null
12	expense_charged_to	String	50	Not Null
13	frequent_flyer_number	String	10	-
14	meal_preference	String	20	Not Null
15	preferred_time	String	10	Not Null
16	return	Date	-	-
17	seat_preference	String	20	-
18	approval_manager	String	50	Not Null

7) Travel_Bill

Sr. No.	Field Name	Data Type	Width	Constraint
1	bill_id	Number	5	Primary Key

2	travel_id	Number	5	Foreign Key
3	bill_category	String	20	Not Null
4	bill_image	String	50	Not Null

8) Expense_Charged_To_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	exp_id	Number	5	Primary Key
2	expense_charged_to	String	50	Not Null

9) Approval_Manger_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	user_id	Number	5	Foreign Key
2	approval_manager	String	50	Not Null

10) Company_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	company_id	Number	5	Primary Key
2	company_name	String	50	Not Null

11)Login

Sr. No.	Field Name	Data Type	Width	Constraint
1	user_name	String	50	Not Null
2	password	String	20	Not Null

12)Location_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	location_id	Number	5	Primary Key
2	location	String	50	Not Null

13) Access_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	user_id	Number	5	Foreign Key
2	access_type	String	10	Not Null

14) Designation_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	designation_id	Number	5	Primary Key
2	designation	String	20	Not Null

15) Visa

Sr. No.	Field Name	Data Type	Width	Constraint
1	user_id	Number	5	Foreign Key
2	valid_visa	String	3	Not Null

3	visa_number	String	10	-
4	visa_country	String	20	Not Null
5	visa_category	String	20	-
6	visa_type	String	20	-
7	type_of_entry	String	10	-
8	valid_from	Date	-	Not Null
9	valid_till	Date	-	Not Null
10	visa_image	String	50	Not Null
11	additional_visa_available	String	3	Not Null
12	intended_date_of_travel	Date	-	Not Null
13	total_harbinger_experience	String	10	-
14	total_project_experience	String	10	-
15	total_work_experience	String	10	-

16) Visa_Category_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	visa_category_id	Number	5	Primary Key
2	visa_category	String	20	Not Null

17) Visa_Type_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	visa_type_id	Number	5	Primary Key
2	visa_type	String	20	Not Null

18)Country_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	country_id	Number	5	Primary Key
2	country	String	20	Not Null

19)Email_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	email_id	String	50	Primary Key
2	category	String	20	Not Null

20)Project_Master

Sr. No.	Field Name	Data Type	Width	Constraint
1	project_id	Number	5	Primary Key
2	project_name	String	50	Not Null

3.16 Test Procedures and Implementation

Software testing is a critical element of software quality assurance & represents the ultimate review of specification, design and code generation.

It is the process of executing a program with a primary objective of finding errors. Testing gives the guarantee that the software does not fail and runs according to its specification and in the way the end user expects.

This can be done by various software testing techniques which provide a systematic guidance for designing tests that exercise the internal logic of software components, and exercise the input and output domains of the program to uncover errors in programming functions, behavior and performance.

Testing is the exposure of system to trial input to see whether it produces correct output. Testing is the process of detecting presence of faults. Once the source code has been generated, software must be tested to uncover as many errors as possible before delivery to your customer. Our goal is to design a series of test cases that have likelihood of finding errors. That's where Software testing Techniques enter into the picture. A set of test cases designed to exercise both internal logic and external requirements is designed and documented, expected results are defined and actual results are recorded.

Testing Objectives:-

The testing objectives are summarized in the following three steps

1. Testing is the process of executing a program with the intent of finding a bug.
2. A good case is one that has a high probability of finding an as yet undiscovered error.
3. A successful test is the one that uncover yet an undiscovered error.

Unit testing:

Unit testing, also known as component testing refers to tests that verify the functionality of a specific section of code usually at the functional level. In an object-oriented environment, this is usually at class-level and the minimal unit tests include the constructors and destructors. These type of tests are usually written by developers as they work on code (white-box style), to ensure that the specific function is working as expected.

One function might have multiple tests, to catch corner cases or other branches in the code. Unit testing alone cannot verify the functionality of a piece of software, but rather is

used to assure that the building blocks of the software work independently of each other.

Integration Testing

Integration Testing is any type of software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an interactive way or all together ("big bang"). Normally the former is considered a better

practice since it allows interface issues to be localized more quickly and fixed.

Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). Progressively user groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a software.

System Testing :

SystemTesting tests a completely integrated system to verify that it meets its requirements.

The testing phase is an important part of software development, It is the process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied.

Acceptance Testing :

Acceptance testing is performed with realistic data of the client to demonstrate that the software is working satisfactorily. Testing here is focused on external behaviour of the system; the internal logic of the program is not emphasized.

Test cases should be selected so that the largest number of attributes of an equivalence class is exercised at once.

The testing phase is an important part of software development. It is the process of finding errors and missing operations and also a

complete verification to determine whether the objectives are met and the user requirements are satisfied.

Acceptance testing is performed along with the client to show that to see that all requirements are satisfied whatever may be the attributes its working well provided all the attributes are valid. If not it displays corresponding messages for getting valid attributes.

Alpha Testing:

Alpha testing is simulated or actual operational testing by potential users/customers or an independent test team at the developers site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.

Beta Testing:

Beta testing comes after alpha testing and can be considered a form of external user acceptance testing. Versions of the software, known beta versions, are released to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the products have few faults or bugs.

Sometimes, beta versions are made available to the open public to increase the feedback filled to a maximal number of future users.

Usability Testing:

Usability testing is needed to check if the user interface is easy to use and understand. It is connected mainly with the use of the application.

Security Testing:

Security testing is essential for software that processes confidential data to prevent system intrusion by hackers.

White Box Testing :

This is the unit testing method where a unit will be taken at a time and tested thoroughly at a statement level to find the maximum possible errors.

We tested stepwise every piece of code, taking care that every statement in the code is executed at least once; the white box testing is also called glass box Testing.

Black Box Testing

This testing method considers a module as a single unit and checks the unit at interface and communication with other modules rather getting into details as statement level. Output for a given set of input combinations are forwarded other module.

TEST CASES :

TEST CASE ID	SCENARIO TO TEST	STEPS TO PERFORM	EXPECTED RESULT	ACTUAL RESULT	PASS/ FAIL
TCL1	Log-in into application as Employee/Manager /Admin Head	1.Open the log in page of the application. 2.Enter the valid user name. 3.Enter valid password. 4.Click on Log in button.	Application should except valid user name and valid password entered by user and should redirect user to respected dashboard.	Log in into Application is successful	Pass

TCL1.1	Log-in application	into	<p>1.Open the log in page of the application.</p> <p>2.Enter in-valid user name.</p> <p>3.Enter valid password.</p> <p>4.Click on Log in button.</p>	<p>Application should not accept invalid user name. Application should throw message “Invalid Credentials”.</p>	Log in denied with appropriate message	Pass
--------	--------------------	------	--	---	--	------

TCL1.2	Log in application	into	<p>1.Open the log in page of the application.</p> <p>2.Enter the valid user name.</p> <p>3.Enter in-valid password.</p> <p>4.Click on Log in button.</p>	<p>Application should not accept invalid user name.</p> <p>Application should throw message “Invalid Credentials”.</p>	Log in denied with appropriate message	Pass
--------	--------------------	------	--	--	--	------

TEST CASE ID	SCENARIO TO TEST	STEPS TO PERFORM	EXPECTED RESULT	ACTUAL RESULT	PASS/ FAIL
TCNTR2	If user is new to the system	1.Click on New Travel request on the dashboard. 2.Employee Master tab will be opened . 3.Click on next.	1.Validation errors should be displayed with respective fields of Employee Master Page 2.Form should not be redirected to next tabs.	1.Validati on errors displayed with respective fields of Employee Master. 2.Form not redirected to next tab.	Pass

TCNTR2.1	If user is new to the system	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened .</p> <p>3.Fill all the required details.</p>	Redirect user to Travel details	Redirected user to Travel details	Pass
----------	------------------------------	--	---------------------------------	-----------------------------------	------

TCNTR2.2	If user is new to the system	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened and fill all the required details,now click next.</p> <p>3.Travel Details tab will get open ,now click on next.</p>	<p>1.Validation errors should be displayed with respective fields of Travel Details.</p> <p>2.Form should not be redirected to next tabs.</p>	<p>1.Validati on errors displayed with respective fields of Travel Details.</p> <p>2.Form not redirected to next tabs.</p>	Pass
----------	------------------------------	--	---	--	------

TCNTR2.3	If user is new to the system and if user select Domestic from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened and fill all the required details,now click next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select Domestic from Drop down Travel Type.</p> <p>5) Fill all the required details.</p>	Form will get submitted.	Form Submitted	Pass
----------	--	--	--------------------------	----------------	------

TCNTR2.4	If user is new to the system and if user select International from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened and fill all the required details,now click next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required details and click next.</p>	Form will be redirected to Passport Details	Form redirected to Passport Details	Pass
----------	---	--	---	-------------------------------------	------

TCNTR2.5	If user is new to the system and if user select International from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened and fill all the required details,now click next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required details. And click next.</p> <p>5.Passport Detail tabs will be displayed. Click on next.</p>	<p>1.Validation errors should be displayed with respective fields of Passport Details.</p> <p>2.Form should not be redirected to next tabs i.e Insurance Details.</p>	<p>1.Validati on errors displayed with respected fields of Passport Details.</p> <p>2.Form not redirected to next tabs i.e Insurance Details.</p>	Pass
----------	---	---	---	---	------

TCNTR2.6	If user is new to the system and if user select International from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened and fill all the required details,now click next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required details. And click next.</p>	Form will be redirected to Visa Details	Form redirected to Visa Details	Pass
----------	---	---	---	---------------------------------	------

		5.Passport Detail tabs will be displayed. Fill all required details and Click on next.		
--	--	---	--	--

TCNTR2.7	If user is new to the system and if user select International from Travel details.	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened and fill all the required details,now click next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required details. And click next.</p>	<p>1.Validation errors should be displayed with respected fields of Visa Details.</p> <p>2.Form should not be submitted.</p>	<p>1.Validati on errors displayed with respected fields of Visa Details.</p> <p>2.Form not submitted.</p>	Pass
----------	--	---	--	---	------

5.Passport
Detail tabs
will be
displayed.
Fill all
required
details and
Click on next.

6.Visa Details
tabs will get
open .Click
on next

TCNTR2.8	If user is new to the system and if user select International from Travel details.	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened and fill all the required details,now click next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required details. And click next.</p> <p>5.Passport Detail tabs will be displayed. Fill all required</p>	Form will get submitted.	Form submitted.	Pass
----------	--	--	--------------------------	-----------------	------

details and
Click on next.

6. Visa Details
tabs will get
open .Fill all
required
details and
click next.

TCNTR2.9	If user is Existing user	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened . With filled in information of Employee which can be edited by user.</p> <p>3.Click on next.</p>	Form will be directed to Travel Details	Form redirected to Travel Details	Pass
----------	--------------------------	--	---	-----------------------------------	------

TCNTR2.10	If user is Existing User	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened . With filled in information of Employee which can be edited by user. Click on next.</p> <p>3.Travel Details tabs will displayed ,Click on next.</p>	<p>1.Validation errors should be displayed with respected fields of Travel Details.</p> <p>2.Form should not be redirected to next tabs.</p>	<p>1.Validati on errors displayed with respected fields of Travel Details.</p> <p>2.Form not redirected to next tabs.</p>	Pass
-----------	--------------------------	---	--	---	------

TCNTR2.11	If user is Existing User and if user select Domestic from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened . With filled in information of Employee which can be edited by user. Click on next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select Domestic from Drop down Travel Type.</p> <p>Fill all the required details.</p>	Form will get submitted.	Form Submitted	Pass
-----------	--	--	--------------------------	----------------	------

TCNTR2.12	If user is Existing User and if user select International from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened . With filled in information of Employee which can be edited by user. Click on next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required details and click next.</p>	Form will be redirected to Passport Details	Form redirected to Passport Details	Pass
-----------	---	---	---	-------------------------------------	------

TCNTR2.13	If user is Existing User and if user select International from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened . With filled in information of Employee which can be edited by user. Click on next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required</p>	<p>1.Validation errors should be displayed with respected fields of Passport Details.</p> <p>2.Form should not be redirected to next tabs i.e Insurance Details.</p>	<p>1.Validati on errors displayed with respected fields of Passport Details.</p> <p>2.Form not redirected to next tabs i.e Insurance Details.</p>	Pass
-----------	---	---	--	---	------

details. And
click next.

5.Passport
Detail tabs
will be
displayed.
Click on next.

TCNTR2.14	If user is Existing User and if user select International from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened . With filled in information of Employee which can be edited by user. Click on next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required</p>	Form will be redirected to Visa Details	Form redirected to Visa Details	Pass
-----------	---	---	---	---------------------------------	------

details. And
click next.

5.Passport
Detail tabs
will be
displayed.
Fill all
required
details and
Click on next.

TCNTR2.15	If user is Existing User and if user select International from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened . With filled in information of Employee which can be edited by user. Click on next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required</p>	<p>1.Validation errors should be displayed with respected fields of Visa Details.</p> <p>2.Form should not be submitted.</p>	<p>1.Validati on errors displayed with respected fields of Visa Details.</p> <p>2.Form not submitted.</p>	Pass
-----------	---	---	--	---	------

		<p>details. And click next.</p> <p>5.Passport Detail tabs will be displayed. Fill all required details and Click on next.</p> <p>6.Visa Details tabs will get open .Click on next</p>		
--	--	---	--	--

TCNTR2.16	If user is Existing User and if user select International from Travel details	<p>1.Click on New Travel request on the dashboard.</p> <p>2.Employee Master tab will be opened and fill all the required details,now click next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select International from Drop down Travel Type. Fill all the required details. And click next.</p>	Form will get submitted.	Form submitted.	Pass
-----------	---	---	--------------------------	-----------------	------

		<p>5.Passport Detail tabs will be displayed. Fill all required details and Click on next.</p> <p>6.Visa Details tabs will get open .Fill all required details and click next.</p>		
--	--	---	--	--

CHAPTER 4
USER MANUAL

4.1 User Manual

For any system to be successful it is important that the intended user find the system easy to operate. The purpose of the user manual is to make user acquainted with the system and help user understand the system and operate it conveniently. The User Manual is prepared reflexively because it is an item that must accompany every system.

The manual contain several screenshots that describes how to use the entire system. This Manual helps user to navigate efficiently through the system and help user to solve issues wherever they occur.

Information about the system.

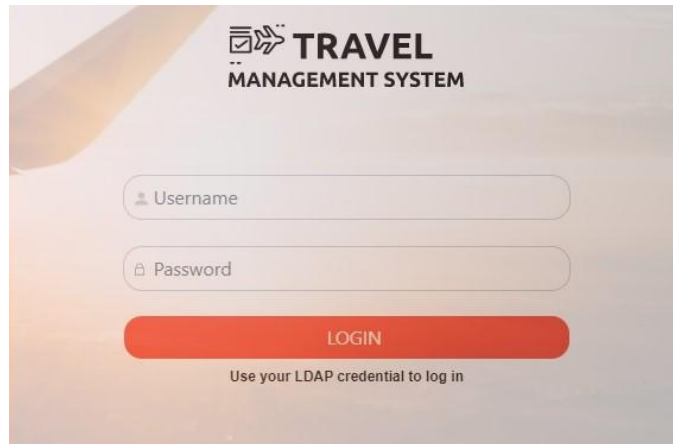
The system contains following users:

- 1) Employee
- 2) Manager
- 3) Admin

The system has following features:

- 1) Employee/Manager Dashboard
- 2) Creation of new travel request
- 3) Approval/Rejection of travel requests
- 4) Uploading of various documents like passport, visa, etc
- 5) Editing of Master data tables
- 6) Uploading bills
- 7) Reports

Log in into the system using LDAP credentials. You are redirected to dashboard after successful credential verification of Username and Password.



Dashboard

This tab denotes the dashboard, which contains all the travel requests made by the user.

In case of Manager and Admin Head, dashboard contains following two tabs:

My Request

This tab in the dashboard, contains all the travel requests made by the manager/admin head.

Pending Request

This tab in the dashboard, indicates all the travel requests made by the user that are pending for approval from manager/admin head.

New Travel Request

This tab in navigation bar is used to create new travel request.

New Travel Request contains following tabs:

Employee Master

Fill user details such as name, employee ID, company, designation, gender, approval manager, mobile number, project, etc.

Travel Details

Fill here all the travel details such as location, departure date, return date, trip type, meal preference, booking details, accommodation etc.

Passport Details

Fill all the passport related information here.

Insurance Details

Fill nominees here for insurance.

Visa Details

Fill all the visa related information here.

The Admin Head contains one more tab in the login:

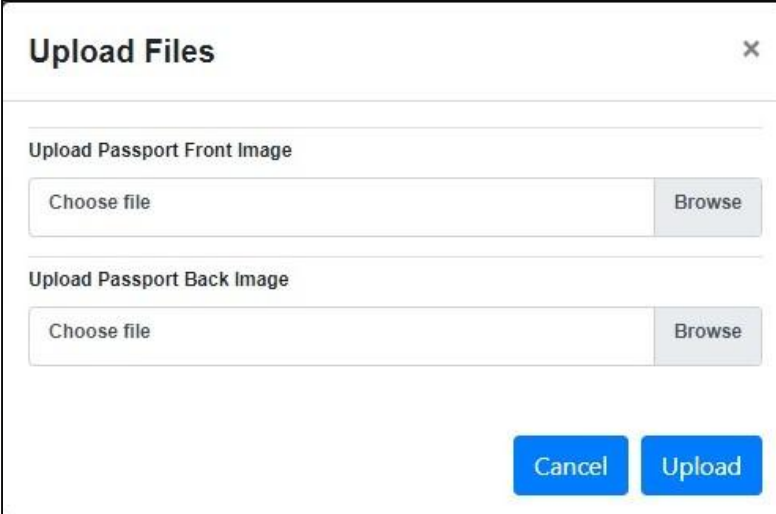
Edit Masters

This tab is used to edit the data used for dropdowns as well as options like budget etc.

Utilities:

This “UPLOAD” button is used to upload documents such as passport, visa, bills etc.

It generates a popup window as follows:



The screenshot shows a modal window titled "Upload Files" with a close button (X) in the top right corner. The window is divided into two sections. The first section is labeled "Upload Passport Front Image" and contains a text input field with the placeholder "Choose file" and a "Browse" button to its right. The second section is labeled "Upload Passport Back Image" and also contains a text input field with the placeholder "Choose file" and a "Browse" button to its right. At the bottom right of the window, there are two blue buttons: "Cancel" and "Upload".

Select the files to upload by using







“BROWSE” button and click on

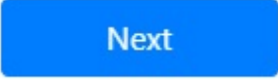

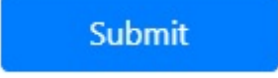
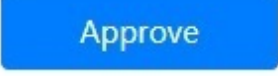

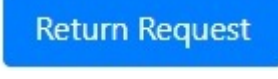
“UPLOAD” to upload.

Here, it is example of passport upload.

4.2 Operations Manual

There are various symbols and buttons on the web application. Here are their descriptions :

Symbol	Meaning
 Pending	Denotes approval status: Request is pending.
 Approved	Denotes approval status: Request is approved.
	Used to view form or other details.
	Used to edit form or other details.
	Used to delete.
	Used to upload bills.

Button	Use
	Go to next tab or page.
	Go to previous tab or page.
	Submit request, bill etc.
	Approve travel request.
	Reject travel request.
	Return travel request for correction.

4.3 Program Specifications

1) Add Employee Details

Module	User Profile Management, Travel Request Management
Program Name	Add Employee Details
Purpose	Add Employee Details to the Employee table, if the employee is creating a travel request for the first time.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	The details of the employee are stored in the Employee table.

2) Add Travel Details

Module	Travel Request Management
Program Name	Add Travel Details.
Purpose	Add Travel details to Travel_Details table like travel date, source city, destination city, etc.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	The travel details are stored in the Travel_Details table and a travel id is generated automatically.

3) Add Passport Details

Module	Travel Request Management
Program Name	Add Passport Details
Purpose	Add passport details to Passport table, if the employee is travelling abroad.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	The passport details are stored in the Passport table along with scanned copy of the passport.

4) Add Insurance Details

Module	Travel Request Management
Program Name	Add Insurance Details
Purpose	Add insurance details to Insurance table, if the employee is travelling abroad.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	The nominee details are stored in the Insurance table.

5) Add Visa Details

Module	Travel Request Management
Program Name	Add Visa Details
Purpose	Add Visa details to Visa table, if the employee is travelling abroad.
Input Details	The required fields should not be blank and the user should provide valid data for each field.

Output	The visa details are stored in the Visa table along with scanned copy of the visa.
--------	--

6) Add Accomodation Details

Module	Travel Request Management
Program Name	Add Accomodation Details
Purpose	Add accomodation details to Accomodation table, if the employee is doing bookings through travel desk of the company.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	The accommodation details are stored in the Accommodation table.

7) Upload Bills

Module	Travel Request Management
Program Name	Upload Bills
Purpose	Add bills related to a particular tour/visit, after that tour/visit is completed.
Input Details	The bill image and bill category must be provided by the user.
Output	The bills associated with a particular travel id are added to the Bills table.

8) Approve Travel Request

Module	Approval Status
Program Name	Approve Travel Request
Purpose	Approve a pending request from an employee.

Input Details	Delivery Manager should select the expense bearer. Comments are optional to both the admin head and the delivery manager.
Output	The request is approved and the requestor is acknowledged regarding the same through email.

9) Reject Travel Request

Module	Approval Status
Program Name	Reject Travel Request
Purpose	Reject a pending request from an employee.
Input Details	The manager or admin head must select the rejection reason. Comments are optional.
Output	The request is rejected and the requestor is acknowledged regarding the same through email.

10) Return Travel Request

Module	Approval Status
Program Name	Return Travel Request
Purpose	Return a pending request from an employee for correction(s).
Input Details	The manager or admin must specify the changes required to be made by the employee.
Output	The request is returned to the requestor for some correction(s) and the requestor is acknowledged regarding the same through email.

11)Edit Masters

Module	Master Data Management
Program Name	Edit Masters
Purpose	Add, Edit or Delete master data from the master tables.
Input Details	The admin head should specify the modifications to be made in the master data.
Output	The data from the master table(s) is updated.

12)Report Generation

Module	Reports
Program Name	Report Generation
Purpose	Show reports of travel requests based on various categories.
Input Details	Data will be fetched from the databse based on the category selected
Output	Reports will be displayed in Table format and Graph format.

Drawbacks and Limitations:

1) Forget password functionality is not available. User needs to go to the SAP application if he/she forgets the account password.

2) Profile can be updated in Harbinger Network (intranet) only.

3) Accommodation and transport bookings done by the concerned vendor cannot be viewed in this application.

4) Whether an employee actually travelled or only made a travel request, is not tracked in this application.

5) Travel booking vendor does not have access/login to the application.

6) Bills are currently sent to Admin Head through email and cannot be viewed in the application.

Proposed Enhancements :

- 1) Add vendor login to the application.
- 2) Employee can view the travel bookings made by the vendor for his/her travel request.
- 3) Separate screen will be given either to employee or admin head to update whether the employee actually travelled or not.
- 4) Separate screen will be given to Admin head to view the bills uploaded by the employee after his/her tour. The admin head can then either approve or reject each individual bill.
- 5) Visa application form will be provided to the employee, if he/she does not have a valid visa of the destination country.
- 6) A budget will be set for a travel request based on the designation of the employee. Using that budget constraint, the booking vendor will be asked to provide the quotation of the entire tour.

Conclusions :

All the requirements stated by the user have been addressed in this application. The application includes the following :

Employee :

- 1) Login to system using LDAP – SSO credentials.
- 2) Create new travel request
- 3) View travel request
- 4) Edit travel request
- 5) Upload Bills

Delivery Manager :

- 1) Login to system using LDAP – SSO credentials.
- 2) View Pending requests
- 3) Approve Request/Reject Request/Return Request for correction
- 4) Create new travel request

5) View travel request

6) Edit travel request

7) Upload Bills

Admin Head :

1) Login to system using LDAP – SSO credentials

2) View pending requests

3) Approve Request/Reject Request/Return Request for correction

4) Update Master data

5) View Reports

The application is tested successfully.

As we know, that any project, even on completion, requires constant improvement and changes which gives way for release of new version.

This application is very user-friendly.

For developing this application, technologies such as HTML, CSS, React JS, Bootstrap, Node JS are used which are in great demand in IT market currently for developing Web Applications.

Bibliography

Books :

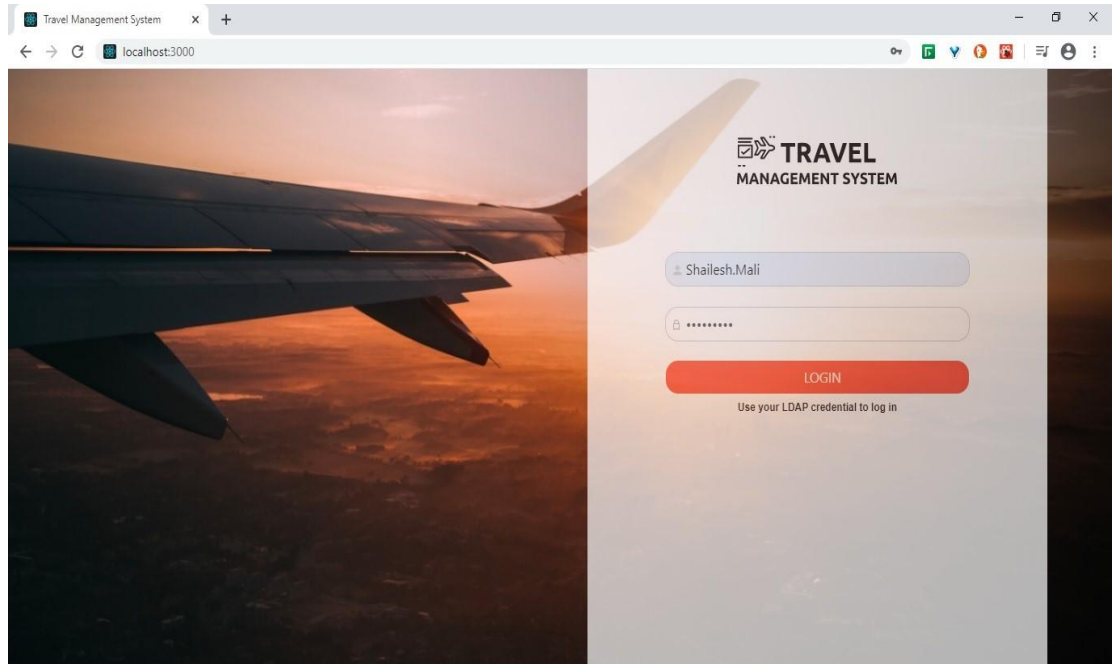
- Node.js Complete Reference Guide - *by Valentin Bojinov, David Herron, Diogo Resende*

Websites :

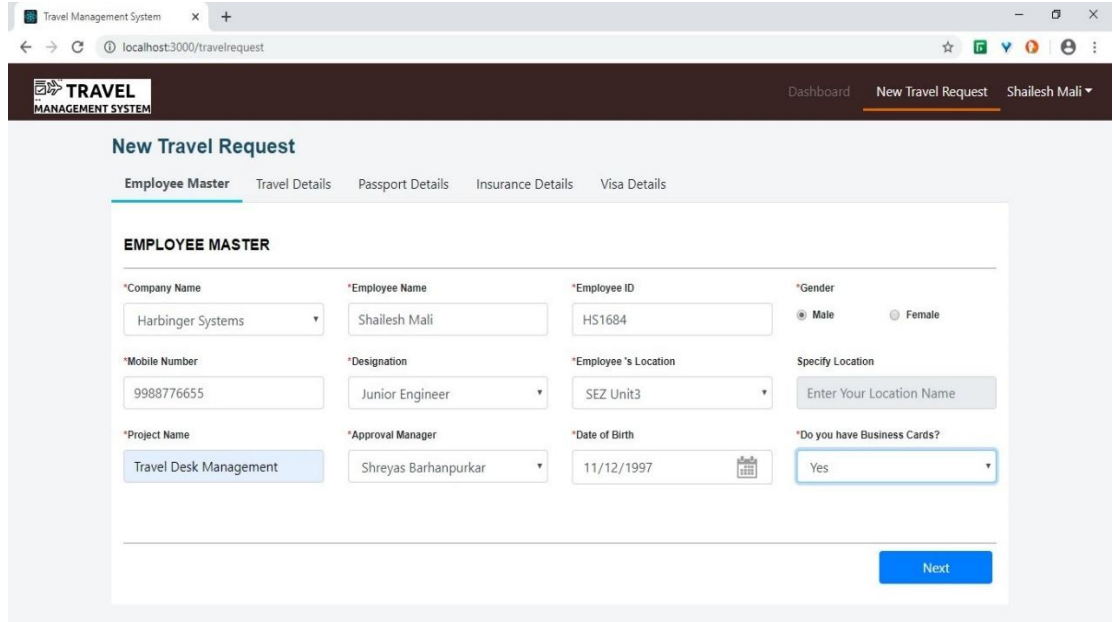
- <https://nodejs.org/en/>
- <https://mongoosejs.com/docs/guide.html>
- <https://stackoverflow.com/tags/node.js>
- <https://reactjs.org/>
- <https://github.com/nodejs>
- <https://github.com/reactjs>

ANNEXURE 1
USER INTERFACE SCREEN

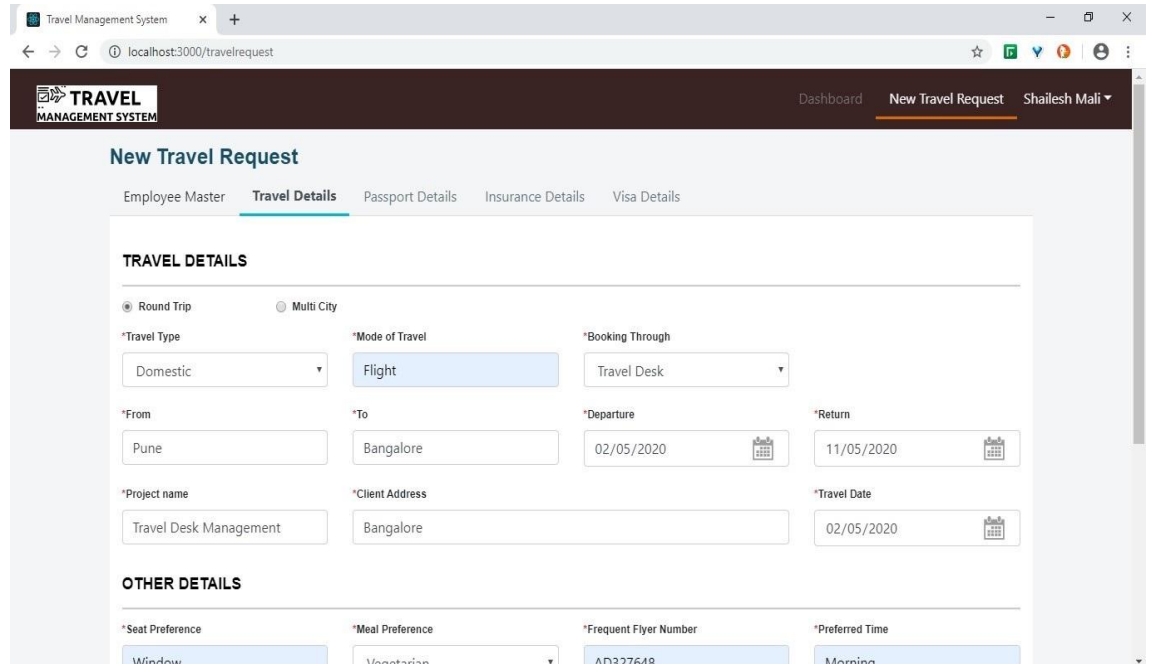
1) Login Page



2) New Travel Request – Employee Details



3) New Travel Request - Travel Details



Travel Management System x +

localhost:3000/travelrequest

Dashboard New Travel Request Shailesh Mali

New Travel Request

Employee Master **Travel Details** Passport Details Insurance Details Visa Details

TRAVEL DETAILS

Round Trip Multi City

*Travel Type: Domestic
*Mode of Travel: Flight
*Booking Through: Travel Desk

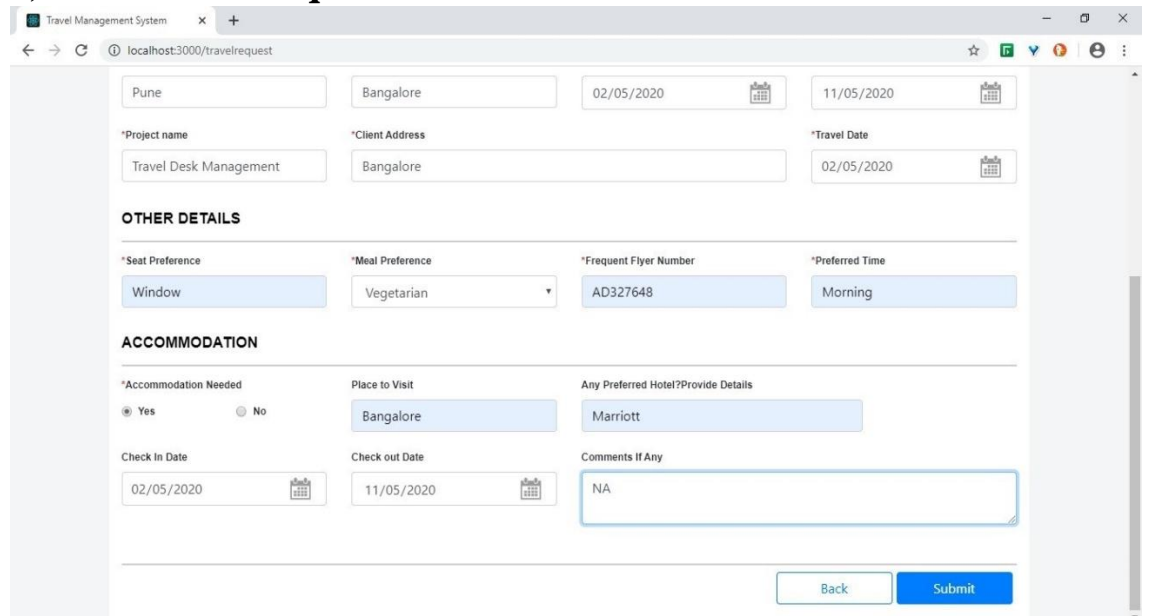
*From: Pune
*To: Bangalore
*Departure: 02/05/2020
*Return: 11/05/2020

*Project name: Travel Desk Management
*Client Address: Bangalore
*Travel Date: 02/05/2020

OTHER DETAILS

*Seat Preference: Window
*Meal Preference: Vegetarian
*Frequent Flyer Number: AD327648
*Preferred Time: Morning

4) New Travel Request – Travel Details



Travel Management System x +

localhost:3000/travelrequest

Dashboard New Travel Request Shailesh Mali

New Travel Request

Employee Master **Travel Details** Passport Details Insurance Details Visa Details

TRAVEL DETAILS

Round Trip Multi City

*Travel Type: Domestic
*Mode of Travel: Flight
*Booking Through: Travel Desk

*From: Pune
*To: Bangalore
*Departure: 02/05/2020
*Return: 11/05/2020

*Project name: Travel Desk Management
*Client Address: Bangalore
*Travel Date: 02/05/2020

OTHER DETAILS

*Seat Preference: Window
*Meal Preference: Vegetarian
*Frequent Flyer Number: AD327648
*Preferred Time: Morning

ACCOMMODATION

*Accommodation Needed: Yes No

Place to Visit: Bangalore
Any Preferred Hotel? Provide Details: Marriott

Check In Date: 02/05/2020
Check out Date: 11/05/2020
Comments If Any: NA

Back Submit

5) Employee Dashboard

Request ID	Requested by	Project	Requested On	Manager Approval	Admin Head Approval	Action
272	Shailesh Mali	Travel Desk Management	2020-02-21	Pending	Pending	View Edit

6) View Travel Request

Travel Details

Round Trip

Travel Type: Domestic Mode Of Travel: Flight Booking Through: Travel Desk

From: Pune To: Bangalore Departure: 2020-05-02 Return: 2020-05-11

Project Name: Travel Desk Management Client Address: Bangalore Travel Date: 2020-05-02

Other Details

7) View Travel Request

Travel Request

Other Details

Seat Preference	Meal Preference	Frequent Flyer No.	Preferred Time
Window	Vegetarian	AD327648	Morning

Accommodation

Accommodation Needed	Place to Visit	Preferred Hotel Details
Yes	Bangalore	Marriott
Check-in Date	Check-out Date	Comments if any
2020-02-27	2020-03-11	NA

Close Edit

8) International Travel Request – Travel Details

New Travel Request

Employee Master **Travel Details** Passport Details Insurance Details Visa Details

TRAVEL DETAILS

Round Trip Multi City

*Travel Type: International
*Mode of Travel: Flight
*Booking Through: Travel Desk

*From: Pune
*To: California
*Departure: 14/07/2020

*From: California
*To: NewYork
*Departure: 30/07/2020

⊕ Add Another City

*Project name *Client Address *Travel Date

9) International Travel Request – Travel Details

Travel Management System x +

localhost:3000/travelrequest

[Add Another City](#)

*Project name: Travel Desk Management

*Client Address: California

*Travel Date: 14/07/2020

OTHER DETAILS

*Seat Preference: Window

*Meal Preference: Non-Vegetarian

*Frequent Flyer Number: AD327648

*Preferred Time: 10:00pm

ACCOMMODATION

*Accommodation Needed: Yes No

Place to Visit: California, New York

Any Preferred Hotel? Provide Details: Marriott

Check In Date: 15/06/2020

Check out Date: 28/07/2020

Comments If Any: Deluxe Suite Required

[Back](#) [Next](#)

10) International Travel Request – Upload Passport

TRAVEL MANAGEMENT SYSTEM Dashboard [New Travel Request](#) Shailesh Mali

New Travel Request

Employee Master [Travel Details](#) **Passport Details** [Insurance Details](#) [Visa Details](#)

PASSPORT DETAILS

*Has a Passport: Yes

*Passport Number: A87364872

*Date of Expiry: 24/12/2020

Upload Files

Upload Passport Front Image: download.png [Browse](#)

Upload Passport Back Image: download1.jpg [Browse](#)

[Cancel](#) [Upload](#)

[Back](#) [Next](#)

Please refer following links for Harbinger Travel Policy [Domestic](#) | [International](#)

11) International Travel Request – Passport Details

The screenshot shows the 'New Travel Request' form in the 'Passport Details' tab. The form includes the following fields and options:

- *Has a Passport:** A dropdown menu with 'Yes' selected.
- If yes, please provide details:** A text label.
- *Upload Passport:** An 'Upload' button and two links: 'download.png' and 'download1.jpg'.
- *Passport Number:** A text input field containing 'A87364872'.
- *Place Of Issue:** A text input field containing 'Pune'.
- *Date of Issue:** A date picker field containing '12/02/2020'.
- *Date of Expiry:** A date picker field containing '20/02/2020'.

At the bottom of the form, there are 'Back' and 'Next' buttons. A footer note reads: 'Please refer following links for Harbinger Travel Policy [Domestic](#) | [International](#)'.

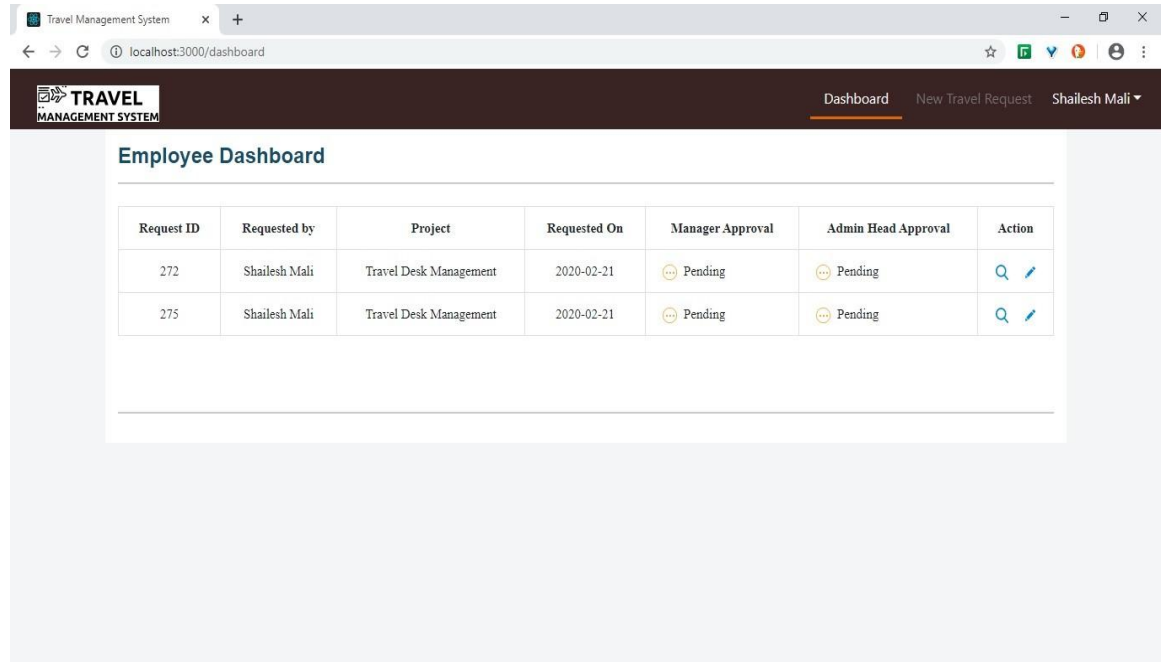
12) International Travel Request – Visa Details

The screenshot shows the 'New Travel Request' form in the 'Visa Details' tab. The form includes the following fields and options:

- *Valid Visa:** Radio buttons for 'Yes' (selected) and 'No'.
- *Upload Visa:** An 'Upload' button and a link: 'download.png'.
- *Visa Number:** A text input field containing 'A76768768'.
- *Visa Country:** A dropdown menu with 'United States' selected.
- *Visa Category:** A dropdown menu with 'Business Visa' selected.
- *Visa Type:** A dropdown menu with 'L1' selected.
- *Valid From:** A date picker field containing '05/02/2020'.
- *Valid Till:** A date picker field containing '18/03/2020'.
- *Intended Date Of Travel:** A date picker field containing '27/02/2020'.
- *Type of Entry:** A dropdown menu with 'Single' selected.
- *Mention if Additional Visa Available:** A dropdown menu with 'No' selected.
- *Total Work Experience:** A text input field containing '4'.
- *Total Harbinger Experience:** A text input field containing '2'.
- *Current Project Total Experience:** A text input field containing '1'.

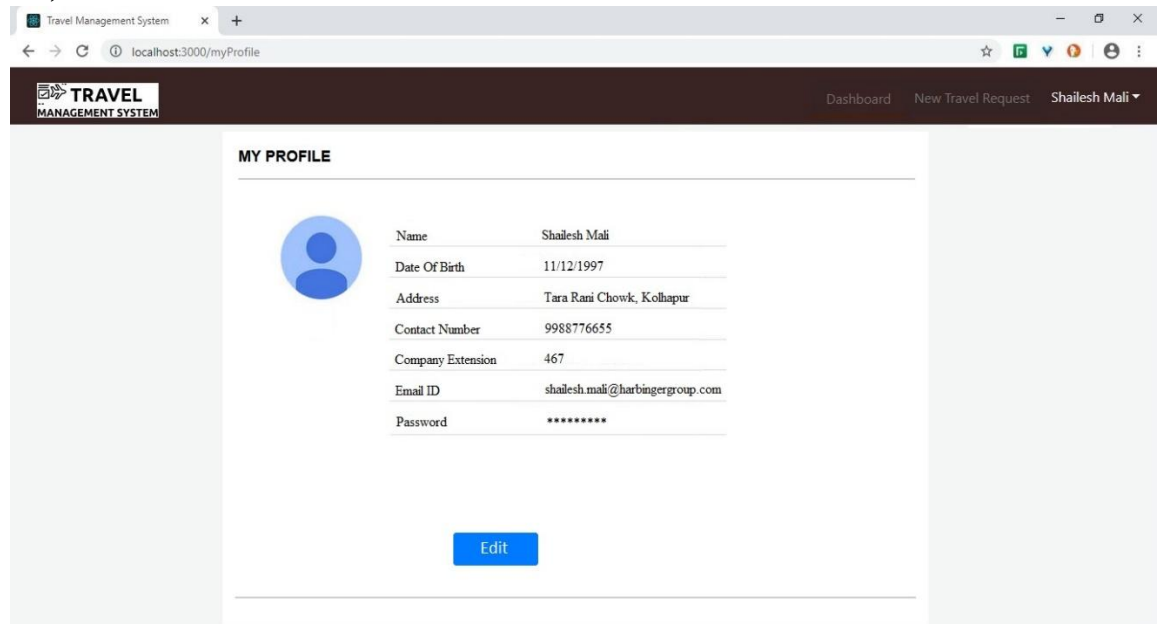
At the bottom of the form, there are 'Back', 'Cancel', and 'Submit' buttons. A footer note reads: 'Please refer following links for Harbinger Travel Policy [Domestic](#) | [International](#)'.

13) Employee Dashboard




Request ID	Requested by	Project	Requested On	Manager Approval	Admin Head Approval	Action
272	Shailesh Mali	Travel Desk Management	2020-02-21	Pending	Pending	Q ✎
275	Shailesh Mali	Travel Desk Management	2020-02-21	Pending	Pending	Q ✎

14) View Profile



MY PROFILE



Name	Shailesh Mali
Date Of Birth	11/12/1997
Address	Tara Rani Chowk, Kolhapur
Contact Number	9988776655
Company Extension	467
Email ID	shailesh.mali@harbingergroup.com
Password	*****

[Edit](#)

15) Edit Profile

EDIT PROFILE

Employee Name Shailesh Mali

Date of Birth 11/12/1997

Address Tara Rani Chowk, Kolhapur

Mobile Number 9988776655

Company Extension 467

Email ID shailesh.mali@harbingergroup.com

Current Password Enter Your Old Password

New Password Enter New Password

Confirm New Password Confirm New Password

[Change](#) [Save](#)

16) Manager Dashboard – My Requests

Manager Dashboard

[My Request](#) [Pending Request](#)

Request ID	Requested by	Project	Requested On	Manager Approval	Admin Head Approval	Action
181	Shreyas Barhanpurkar	CDMS	2020-01-29	Pending	Pending	Q ✎
184	Shreyas Barhanpurkar	WT	2020-02-03	Pending	Pending	Q ✎
203	Shreyas Barhanpurkar	WT	2020-02-05	Pending	Pending	Q ✎
243	Shreyas Barhanpurkar	CDMS	2020-02-09	Pending	Pending	Q ✎
173	Shreyas Barhanpurkar	Travel Desk Management	2020-01-15	Approved	Approved	Q ⬇

1 2

17) Upload Bills

Travel ID : 173

From: Pune To: Mumbai Departure: 02/11/2020

From: Mumbai To: Bangalore Departure: 02/18/2020

*Upload Bill Image: Upload 01.jpg *Bill Category: Food

*Upload Bill Image: Upload download.png *Bill Category: Transport

+ Add Bill Submit

18) Manager Dashboard – Pending Requests

Manager Dashboard

My Request Pending Request

Select Request	Request ID	Requested by	Project	Requested On	Manager Approval	Admin Head Approval	Action
<input type="radio"/>	265	Ajinkya Ajgaonkar	Travel Desk Management	2020-02-20	⋮ Pending	⋮ Pending	Q
<input type="radio"/>	271	Ajinkya Ajgaonkar	Travel Desk Management	2020-02-21	⋮ Pending	⋮ Pending	Q
<input type="radio"/>	272	Shailesh Mali	Travel Desk Management	2020-02-21	⋮ Pending	⋮ Pending	Q
<input type="radio"/>	273	Pooja Gore	Travel Desk Management	2020-02-21	⋮ Pending	⋮ Pending	Q
<input type="radio"/>	275	Shailesh Mali	Travel Desk Management	2020-02-21	⋮ Pending	⋮ Pending	Q

1 2

Return Request Reject Approve

19) Approve Request

The screenshot shows a web browser window with the URL `localhost:3000/dashboard`. The page title is "Travel Management System". The main content area is titled "Manager Dashboard" and has two tabs: "My Request" and "Pending Request". The "Pending Request" tab is active, displaying a table of requests. A modal dialog box titled "Approve Request" is open in the foreground. The dialog contains the following fields:

- Client Name: Travel Desk Management
- The Expenses will be Charged to: Harbinger Systems (dropdown menu)
- Notes: NA (text input field)

At the bottom of the dialog are "Cancel" and "Approve" buttons. In the background, the "Pending Request" table has columns for "Select Request", "Request ID", "Requester", "Admin Head Approval", and "Action". The table contains five rows of data, all with "Pending" status and a magnifying glass icon in the "Action" column.

20) Reject Request

The screenshot shows a web browser window with the URL `localhost:3001/dashboard`. The page title is "TRAVEL MANAGEMENT SYSTEM". The main content area is titled "Manager Dashboard" and has two tabs: "My Request" and "Pending Request". The "Pending Request" tab is active, displaying a table of requests. A modal dialog box titled "Reject Request" is open in the foreground. The dialog contains the following fields:

- Client Name: Travel Desk Management
- Reason: Change in requirement (text input field)

At the bottom of the dialog are "Cancel" and "Reject" buttons. In the background, the "Pending Request" table has columns for "Select Request", "Request ID", "Requester", "Admin Head Approval", and "Action". The table contains five rows of data, all with "Pending" status and a magnifying glass icon in the "Action" column. At the bottom of the page, there are navigation buttons: "Return Request", "Reject", and "Approve".

21) Return Request for Correction

The screenshot shows a web browser window with the URL `localhost:3001/dashboard`. The application is titled 'TRAVEL MANAGEMENT SYSTEM' and the user is logged in as 'Shreyas Barhanpurkar'. The main content area is the 'Manager Dashboard' with a 'Pending Request' tab selected. A modal dialog titled 'Return Request for Correction' is open, displaying the following information:

- Client Name: Travel Desk Management
- Changes to be made: Please Postpone travel date by 1 week...

The modal has 'Cancel' and 'Return Request' buttons. In the background, a table of pending requests is visible:

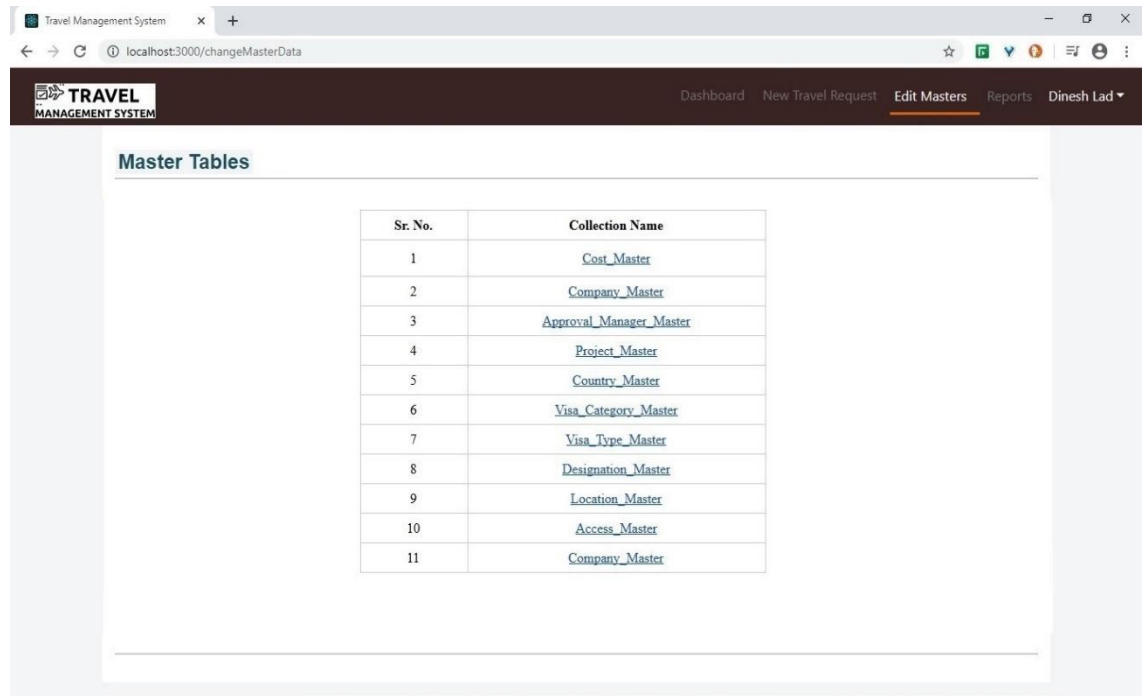
Select Request	Request ID	Request	Admin Head Approval	Action
<input type="radio"/>	265	Ajinkya Ajgaonkar	Pending	?
<input type="radio"/>	272	Shailesh	Pending	?
<input type="radio"/>	273	Pooja C	Pending	?
<input type="radio"/>	275	Shailesh	Pending	?
<input type="radio"/>	185	Pooja C	Pending	?

22) Admin Dashboard – Pending Requests

The screenshot shows a web browser window with the URL `localhost:3000/dashboard`. The application is titled 'TRAVEL MANAGEMENT SYSTEM' and the user is logged in as 'Dinesh Lad'. The main content area is the 'Admin Dashboard' with a 'Pending Request' tab selected. A table of pending requests is displayed:

Select Request	Request ID	Requested by	Project	Requested On	Manager Approval	Admin Head Approval	Action
<input type="radio"/>	189	Ajinkya Ajgaonkar	Travel Desk Management	2020-02-01	Approved	Pending	?
<input type="radio"/>	271	Ajinkya Ajgaonkar	Travel Desk Management	2020-02-21	Approved	Pending	?
<input type="radio"/>	203	Shreyas Barhanpurkar	WT	2020-02-05	Pending	Pending	?
<input type="radio"/>	243	Shreyas Barhanpurkar	CDMS	2020-02-09	Pending	Pending	?
<input type="radio"/>	265	Ajinkya Ajgaonkar	Travel Desk Management	2020-02-20	Pending	Pending	?

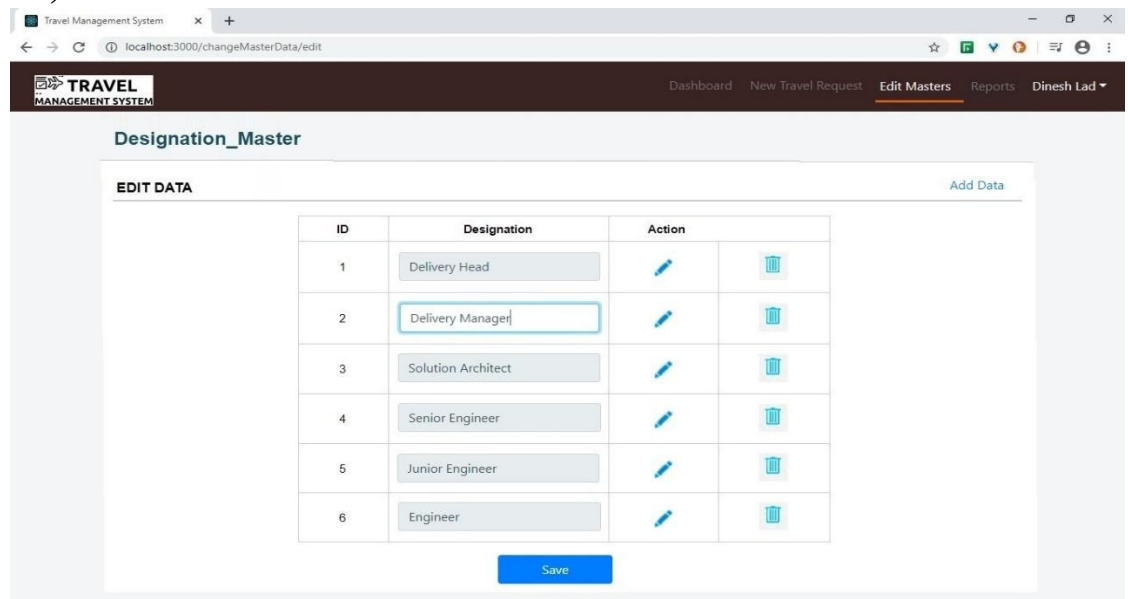
23) Edit Masters















The screenshot shows the 'Master Tables' page in the Travel Management System. The page header includes the system logo and navigation links: Dashboard, New Travel Request, Edit Masters (active), Reports, and Dinesh Lad. The main content area is titled 'Master Tables' and contains a table with the following data:

Sr. No.	Collection Name
1	Cost_Master
2	Company_Master
3	Approval_Manager_Master
4	Project_Master
5	Country_Master
6	Visa_Category_Master
7	Visa_Type_Master
8	Designation_Master
9	Location_Master
10	Access_Master
11	Company_Master

24) Edit Masters

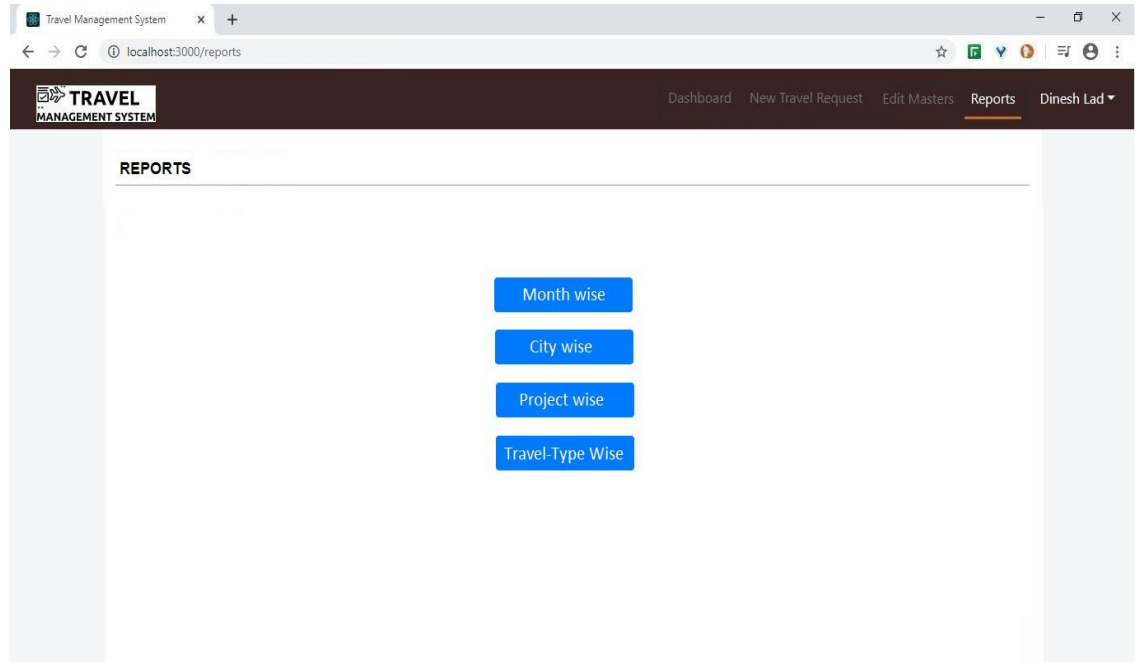


The screenshot shows the 'Designation_Master' edit page in the Travel Management System. The page header includes the system logo and navigation links: Dashboard, New Travel Request, Edit Masters (active), Reports, and Dinesh Lad. The main content area is titled 'Designation_Master' and contains an 'EDIT DATA' section with an 'Add Data' link. The table below shows the current designations and their actions:

ID	Designation	Action
1	Delivery Head	 
2	Delivery Manager	 
3	Solution Architect	 
4	Senior Engineer	 
5	Junior Engineer	 
6	Engineer	 

Save

25) Reports



The screenshot displays a web browser window with the following elements:

- Browser Tab:** Travel Management System
- Address Bar:** localhost:3000/reports
- Page Header:** TRAVEL MANAGEMENT SYSTEM (with logo) | Dashboard | New Travel Request | Edit Masters | **Reports** | Dinesh Lad
- Main Content Area:**
 - REPORTS** (Section Header)
 - Four blue buttons stacked vertically:
 - Month wise
 - City wise
 - Project wise
 - Travel-Type Wise

ANNEXURE 2
OUTPUT REPORTS WITH DATA

1) Month-wise Reports – Tabular Format

Travel Management System | Dashboard | New Travel Request | Edit Masters | **Reports** | Dinesh Lad

MONTH-WISE REPORT

*Select Month: January [View Graph](#)

Sr. No	Request ID	Requested By	Project Name	Travel Date	Source City	Destination City
1	168	Pooja Gore	CDMS	2020-01-05	Pune	Bangalore
2	169	Ajinkya Ajaonkar	WT	2020-01-07	Pune	Mumbai
3	170	Shailesh Mali	Travel Desk Management	2020-01-11	Pune	Bangalore
4	173	Shreyas Barhanpurkar	Travel Desk Management	2020-01-11	Pune	Bangalore
5	174	Shraddha Sandhansive	HR-Payroll System	2020-01-12	Pune	New York
6	176	Tanaya Bambulkar	HR-Payroll System	2020-01-12	Pune	New York
7	178	Ankita Dongarkar	HR-Payroll System	2020-01-12	Pune	New York
8	179	Sonali Dhiman	CDMS	2020-01-19	Pune	Delhi
9	180	Rasika Pashlikar	CDMS	2020-01-19	Pune	Delhi
10	182	Amita Lakshkar	CDMS	2020-01-19	Pune	Delhi

1 2 3

2) Month-wise Reports – Graph Format

Travel Management System | Dashboard | New Travel Request | Edit Masters | **Reports** | Dinesh Lad

MONTH-WISE REPORT

Travel Requests - 2020 | Date : 12-03-2020

Month	No. of Travel Requests
Jan	27
Feb	11
March	4
April	0
May	0
June	0
July	0
August	0
Septemabr	0
October	0
November	0
December	0

3) City-wise Reports – Tabular Format

CITY-WISE REPORT

*Select City:

[View Graph](#)

Sr. No	Request ID	Requested By	Project Name	Travel Date	Source City
1	168	Pooja Gore	CDMS	2020-01-05	Pune
2	170	Shailesh Mali	Travel Desk Management	2020-01-27	Pune
3	173	Shreyas Barhanpurkar	Travel Desk Management	2020-01-27	Pune
4	185	Pooja Gore	CDMS	2020-02-03	Pune
5	191	Ajinkya Ajaonkar	Travel Desk Management	2020-02-03	Pune
6	193	Sonali Dhiman	WT	2020-02-11	Mumbai
7	194	Rasika Pashilkar	WT	2020-02-11	Mumbai
8	241	Shreyas Barhanpurkar	Travel Desk Management	2020-03-01	Pune

4) City-wise Reports – Graph Format

CITY-WISE REPORT

Travel Requests 2020 Date: 12-03-2020

City	No. of Travel Requests
Bangalore	8
Delhi	13
Mumbai	9
New York	8
Paris	4

5) Project-wise Reports – Tabular Format

Travel Management System x +

localhost:3000/reports/project-wise

TRAVEL MANAGEMENT SYSTEM Dashboard New Travel Request Edit Masters Reports Dinesh Lad

PROJECT-WISE REPORT

*Select Project

[View Graph](#)

Sr. No	Request ID	Requested By	Travel Date	Source City	Destination City
1	170	Shailesh Mali	2020-01-11	Pune	Bangalore
2	173	Shreyas Barhanpurkar	2020-01-11	Pune	Bangalore
3	191	Ajinkya Ajgaonkar	2020-02-03	Pune	Bangalore
4	209	Amita Lakshkar	2020-03-01	Pune	Mumbai
5	242	Sonali Dhiman	2020-03-01	Pune	Mumbai
6	265	Ajinkya Ajgaonkar	2020-04-11	Pune	Mumbai
7	271	Ajinkya Ajgaonkar	2020-04-15	Pune	Delhi
8	272	Shailesh Mali	2020-05-02	Pune	Bangalore
9	273	Pooja Gore	2020-05-02	Pune	Bangalore
10	275	Shailesh Mali	2020-07-14	Pune	New York

1 2

6) Project-wise Reports – Graph Format

Travel Management System x +

localhost:3000/reports/project-wise/view-graph

TRAVEL MANAGEMENT SYSTEM Dashboard New Travel Request Edit Masters Reports Dinesh Lad

PROJECT-WISE REPORT

Travel Requests 2020 Date : 12-03-2020

Project	No. of Travel Requests
Travel Desk Management	13
CDMS	9
WT	12
TMS	2
HR-Payroll System	6

7) Travel-Type-wise Report – Tabular Format

Travel Management System x +

localhost:3000/reports/travel-type-wise

TRAVEL MANAGEMENT SYSTEM Dashboard New Travel Request Edit Masters Reports Dinesh Lad

TRAVEL-TYPE-WISE REPORT

*Select Project

[View Graph](#)

Sr. No	Request ID	Requested By	Project Name	Travel Date	Source City	Destination City
1	174	Shraddha Sandhansive	HR-Payroll System	2020-01-12	Pune	New York
2	176	Tanaya Bambulkar	HR-Payroll System	2020-01-12	Pune	New York
3	178	Ankita Dongarkar	HR-Payroll System	2020-01-12	Pune	New York
4	183	Amita Lakshkar	CDMS	2020-07-07	Pune	Paris
5	184	Tanaya Bambulkar	CDMS	2020-07-07	Pune	Paris
6	192	Ajinkya Ajgaonkar	WT	2020-05-13	Pune	California
7	231	Sonali Dhiman	WT	2020-05-13	Pune	California
8	239	Shraddha Sandhansive	HR-Payroll System	2020-06-15	Pune	Singapore
9	260	Ankita Dongarkar	HR-Payroll System	2020-06-15	Pune	Singapore
10	266	Shreyas Barhanpurkar	CDMS	2020-07-07	Pune	Germany

1 2

8) Travel-Type-wise Report – Graph Format

Travel Management System x +

localhost:3000/reports/travel-type-wise/view-graph

TRAVEL MANAGEMENT SYSTEM Dashboard New Travel Request Edit Masters Reports Dinesh Lad

TRAVEL-TYPE-WISE REPORT

Travel Requests 2020 Date: 12-03-2020

Travel Type	No. of Travel Requests
Domestic	25
International	17

ANNEXURE 3
SAMPLE PROGRAM CODE

1) **travelDetailsController.js**

```
import {TravelModel} from '../../mongo-models/travelDetails';
```

```
import {travelDetValidations} from '../travelValidations';
```

```
import {mail} from '../../utils/mailer';
```

```
/**
```

```
 * Method to save travel details when user clicks on next button.
```

```
 * @param {object} req http request
```

```
 * @param {object} res http res
```

```
 * @returns {object}
```

```
*/
```

```
export const saveTravelDetails = async (req,res,next) => {
```

```
  try
```

```
  {
```

```
    const data = req.body;
```

```
data.approval_manager = data.approval_manager.replace(' ','');

travelDetValidations(data);

const travelData = await TravelModel.create(data);

mail('details_confirmation', data);

mail('approval_request',data);

if(data.travel_type=== 'International')

{

    mail('international_travel',data);

}

return res.status(200).send(data);

}

catch(error)

{

    console.log(error);
```

```
    return res.status(500).send(error);

  }

}

export const updateTravelDetails = async (req,res,next) => {

  try

  {

    const data = req.body;

    data.approval_manager = data.approval_manager.replace(' ','');

    travelDetValidations(data);

    const travelData = await TravelModel.updateOne({'travel_id' :
data.travel_id},data);

    mail('update',data);

    return res.status(200).send(travelData);

  }

  catch(error)
```

```
{  
  
  console.log(error);  
  
  return res.status(500).send(error);  
  
}  
  
}  
  
  
  
  
export const updateRequestStatus = async (req,res,next) => {  
  
  try  
  
  {  
  
    const data = req.body;  
  
    const travelData = await TravelModel.updateOne({'travel_id' :  
data.travel_id},data);  
  
    if(data.dm_approval_status == "Approved")  
  
    {
```

```
        mail('rejected_confirmation',data);

        mail('rejected_response',data);

    }

    else if(data.dm_approval_status == "Rejected")

    {

        mail('approved_response',data);

        mail('approval_confirmation',data);

    }

    return res.status(200).send(travelData);

}

catch(error)

{

    return res.status(500).send(error);

}
```

```
}
```

2) viewDataController.js

```
import {TravelModel} from '../../mongo-models/travelDetails';
```

```
import {employeeModel} from '../../mongo-models/employeeData';
```

```
export const get_travel_details = async (req,res) => {
```

```
  try
```

```
  {
```

```
    const data = req.query;
```

```
    const page_number=1;
```

```
    const travelData = await TravelModel.aggregate([ {
```

```
      $lookup:
```

```
{  
  
  from: 'employeeDetails',  
  
  localField:'user_id',  
  
  foreignField: 'user_id',  
  
  as: 'Employee_travel_details'  
  
},  
  
},  
  
{  
  
  $match : { user_id : (data.user_id) }  
  
},  
  
{  
  
  $unwind:'$Employee_travel_details'  
  
},  
  
{
```

```
"$project":  
  
  {  
  
    'travel_id':1,  
  
    'user_id':1,  
  
    'project_name': 1,  
  
    'travel_date':1,  
  
    'employee_name': '$Employee_travel_details.employee_name',  
  
    'request_created_on':1,  
  
    'request_created_by':1,  
  
    'dm_approval_status' :1,  
  
    'admin_approval_status' :1,  
  
  }  
  
},  
  
D);
```



```
}

catch(error)

{

    return res.status(500).send(error);

}

}

export const get_travel_details_for_dm = async(req,res)=>{

try

{

    var page_data_limit = parseInt(process.env.PAGE_NUMBER);

    const data = req.query;

    console.log("data===",data);

    const travelData_for_dm = await TravelModel.aggregate([

        $lookup:
```

```
{  
  
  from: 'employeeDetails',  
  
  localField:'user_id',  
  
  foreignField: 'user_id',  
  
  as: 'Employee_travel_details_for_dm'  
  
},  
  
},  
  
{  
  
  $match : { approval_manager : data.user_id }  
  
},  
  
{  
  
  $unwind:'$Employee_travel_details_for_dm'  
  
},  
  
{
```

```
"$project":  
  
  {  
  
    'travel_id':1,  
  
    'project_name': 1,  
  
    'approval_manager':1,  
  
  
    'employee_name': '$Employee_travel_details_for_dm.employee_name',  
  
    'request_created_on':1,  
  
    'request_created_by':1,  
  
    'visa_details.intended_date_of_travel':1,  
  
    'dm_approval_status' :1,  
  
    'admin_approval_status' :1,  
  
  }  
  
},
```

```
    ]);  
  
  }  
  
  catch(error)  
  
  {  
  
    console.log(error);  
  
    return res.status(500).send(error);  
  
  }  
  
}  
  
export const view_form = async(req,res)=>{  
  
  try  
  
  {  
  
    const data=req.query;  
  
    const travel_id =parseInt(data.travel_id);  
  
    const user_id=data.user_id;
```

```
    console.log(data)

    const form_data_travel = await
TravelModel.findOne({'travel_id': travel_id });

    const form_data_user = await
employeeModel.findOne({'user_id':user_id});

    return
res.status(200).json({"travel_data":form_data_travel,"user_data":for
m_data_user});

}

catch(error)

{

    console.log(error);

    return res.status(500).send(error);

}

}
```