FEEDER MANAGEMENT SYSYTEM

PROJECT REPORT

$P-999$

*Submitted in partial fulfillment of the requirements
for the award of the degree of*

MASTER OF COMPUTER APPLICATIONS

Of Bharathiar University

Submitted by

Mr.P.C.MOHANRAJ
Reg. No – 0038M1040

Guided by

Mr.M.NAGESWARAGUPTHA B.E, Lecturer

Department of Computer Science and Engineering

Kumaraguru College of Technology
(Affiliated to Bharathiar University)

Coimbatore – 641 006

2002 - 2003

Kumaraguru College of Technology

(Affiliated to Bharathiar University)

Coimbatore – 641006

Department of Computer Science & Engineering

CERTIFICATE

This is certify that the project entitled

"FEEDER MANAGEMENT SYSYTEM"

Done by

Mr. P.C.Mohanraj
Reg.No: 0038M1040

Submitted in partial fulfillment of the requirement for the award of the degree of
Master of Computer Applications of Bharathiar University.

....
Head of the Department

....
Guide

Submitted for the university examination held on ....

....
Internal Examiner

....
External Examiner

KINSHIP TECHNOLOGIES PRIVATE LIMITED

March 21,2003

TO WHOMSOEVER IT MAY CONCERN

This is to certify that Mr.Mohanraj.P.C. Who is pursuing his final year M.C.A (Master of Computer Applications) at Kumaraguru College of Technology, Coimbatore has successfully completed his project work entitled " **FEEDER MANAGEMENT SYSTEM**" as part of his academic curriculum during December 2002 and March 2003.

During the project tenure his attendance was found to be regular.

Due to operational reasons, we are not in a position to give the source code to the student.

We wish him all the best in his future endeavors

B.G.Giri
Director

ACKNOWLEDGEMENT

I express my profound respect and sincere gratitude to **Dr.K.K.Padmanaban, Ph.D.,** Principal, **Kumaraguru college of technology**, Coimbatore, for his kind co-operation in allowing me to take up this project work.

I record my sincere thanks to the Head of the Department **Dr.S.Thangaswamy, Ph.D.,** Computer Sciences and Engineering, **Kumaraguru college of technology,** for encouraging me to take up the project at Kinship Technologies Private Limited. Chennai.

I express my sincere thanks to **Mr. A.Muthukumar, M.Sc, MCA, M.Phil,** Senior Lecture, Master of Computer Application, without his motivation and guidance I would not been able to embark on a project of this magnitude.

I would like to thank my guide **Mr.M.Nageswaraguptha B.E**, Lecturer, Dept. of Computer Science and Engineering, Kumaraguru College of Technology, who has been a constant source of encouragement for our project. I would like to express my special thanks to all staff members, who helped us for the successful completion of the project.

I wish to extend my gratitude to **Mr.K.G.Giri,** Director –Kinship Technologies Private Limited, Chennai for giving me the opportunity to do this project.

I am greatly privileged to express my deep sense of gratitude to **Mr.Suresh,** Project Manager and **Mr.Baskar,** the Project Leader for their valuable advice and encouragement.

My deep sense of gratitude to my parents, friends and all others, for their invaluable help and consideration towards me.

SYNOPSIS

The project work entitled "FEEDER MANAGEMENT SYSYTEM"- is an application system developed at Kinship Technologies Private Limited, Chennai for Shipping Companies. This system aims in engineering and in integrating the various business activities. The objectives and goals of the software system are to provide a comprehensive software package for the concern. The entire functionality of collecting information regarding shipping can be managed in a simple and easy to use.

"Feeder Management System" is a web based application. This software will be helpful in automating the various business activities for Shipping Companies. All these functionalities will be possible online through this system.

CONTENTS

1. INTRODUCTION

1.1 CURRENT STATUS OF THE PROBLEM TAKEN UP:

This web portal is being developed for a Shipping companies. The company is involved in the transportation of the various Products to various countries. The company is known for its shuttle feeder connection. This system is being developed in order to integrate the various businesses activities. This project is developed to increase the efficiency of the workflow by reducing the existing time and resources being consumed. The project is mainly concerned with the Business process Reengineering. The user can get the information from the web. Thus the information is being given only to the right person at the right time. This project involves a business process redesigning of a Feeder Service Provider. The successful completion of this exercise will result in modifications of the existing business operation.

The objective of the project is to reengineer the existing business process where in the workflow is optimized. The newly designed model of business process is to be framed where in the ability of the business system to provide information at right time to the right person at right place with shrinking in the consumption of time and resources. This may result in a modification of the structure in the business process, there by with the objective of minimizing the cost of operation.

Thus web enabling the business process and its operations would make them easy to monitor the various business activities through the web.

1.2 RELEVANCE AND INPORTANCE

The main objective of this project is to develop a Web Based System and user-friendly interface. The company has branches at different places. The major problem between them is sharing of data.

Drawbacks in the existing system:

> ➤ Sharing of data is possible.
> ➤ Data are entered manually.
> ➤ Data redundancy occurs.
> ➤ Transaction time is relatively high.
> ➤ Tracking can't be done easily.

In the proposed system, all the above drawbacks are removed. The Screens incorporates all predetermined standards and all follow these standards for the uniformity purposes and to be appealing to the user. Availability of reports, is one of the main advantages so as the administrative authorities can easily view the various transactions taking place. Since sharing data takes place, the data regarding that branch can be tracked easily, for this operation the information need not to be sent separately.

2. LITERATURE SURVEY

2.1 EXISTING SYSYTEM

In Existing system they are using FoxPro for their operations. It has so many disadvantages. Listed below...

- ➢ No Graphical Interface, not user friendly.
- ➢ Data's are entered manually, which results in lot of errors.
- ➢ Sharing of data is not possible.
- ➢ Transaction time is more.
- ➢ Globalization does not exist.

These are the major drawbacks in the existing system, which are Rectified in the proposed system

3. SOFTWARE REQUIREMENTS.

SOFTWARE REQUIREMENT SPECIFICATION DETAILS:

3.1 PRODUCT OVERVIEW AND SUMMARY

This project aims at engineering the business activities. The business process involves in transportation of containers from one port to another port.

3.2 DEVELOPMENT ENVIRONMENT

Platform	: Windows
Programming Tools	: Cold Fusion, Oracle 8.0
Operating System	: Windows NT
Application Server	: Apache Web Server

3.3 OPERATION AND MAINTENANCE ENVIRONMENT

SERVER CONFIURATION
SOFTWARE:

Operating System	: Windows NT Workstation 4.0
Front End	: Cold Fusion 4.0
Browser	: Internet Explorer 6.0
Visual Tool	: Macromedia Dream Weaver 4

CLIENT CONFIGURATION
SOFTWARE:

Operating System	: Windows NT Workstation 4.0
Application Server	: Apache Web Server
Front End	: Cold Fusion 4.0
Browser	: Internet Explorer 6.0
Visual Tool	: Macromedia Dream Weaver 4

3.4 DATAFLOW DIAGRAM

LEVEL – 0 DFD

Voyage → Booking → Arrival Status → Departure Status → Invoice → Collection
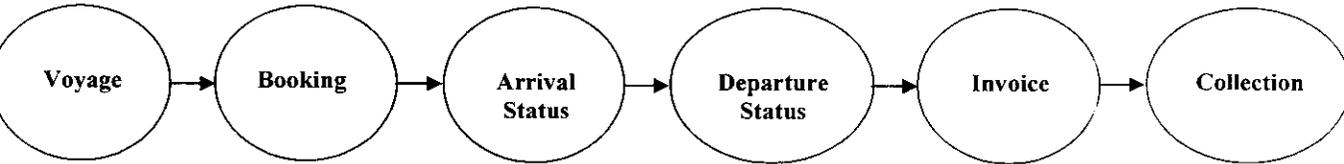
3.5 FUNCTIONAL SPECIFICATIONS

The Project is divided into modules. They are listed as follows

1) Voyage
2) Booking
3) Arrival Status
4) Departure Status
5) Invoice
6) collection

3.5.1 VOYAGE

Voyage is the first stage in the business process. The voyage is defined by selecting the route for the voyage; Route is combination of different sectors. Sector is defined by mentioning the start port and end port. Voyage mentions the appropriate date for the vessel to be arrived in that particular port and finally the vessel is selected for that voyage.

3.5.2 BOOKING

This is the second stage in the business process. After defining the voyage booking is made for different sectors in the voyage. Bookings are made for the sectors with their clients.

3.5.3 ARRIVAL STATUS

When the vessel had arrived at a particular port arrival status is prepared for that particular port. When arrival status is prepared it gives information about the containers that are discharged at that particular port.

3.5.4 DEPARTURE STATUS

After arrival status is prepared departure status is prepared for that voyage in that particular port. Departure status gives the final information about the containers or goods that are loaded in the vessel for that voyage. Even booking is done, if that information is not in the departure status that booking is considered as cancelled.

3.5.5 INVOICE

Invoice is prepared for the client, mentioning the charges for containers that are loaded in the vessel. Charges are defined is defined for that particular client and for that particular sector. Depending upon the charges value invoice amount is calculated and sent to the client.

3.5.6 COLLECTION

Depending upon the invoice value collection is made. Collection can be made at different branches which is mentioned at booking. The collection specifies the mode of collection i.e. by cheque or demand draft or by cash. And if any discount is allowed to that client it is mentioned in collection.

3.6 ACCEPTANCE CRITERIA

The software product, which passes all the test cases, shall be the acceptance criteria for the team to accept the product after User Acceptance Test (UAT).

PROPOSED APPROACH TO THE PRODUCT

4. PROPOSED APPROACH TO THE PRODUCT

The proposed system should be developed in such a way to solve the above problems faced by the present normal system by using latest technology. For this, the ideal environment is the windows environment, as it is the most popular multitasking tool available today. The tool used to develop the system is Cold Fusion for its flexibility and versatility.

The proposed system will help in

- ➤ It will help to automate all the services in the organization.
- ➤ It gives us any details needed at any time including the status etc.
- ➤ All the debugging activities can be done online and so it takes very little time.
- ➤ The system is scalable.
- ➤ User Friendly

5. DETAILS OF DESIGN

The total system is divided into separately named and addressable components called modules, which are integrated to satisfy problem requirements. Here the system is divided into six modules –

- Voyage
- Booking
- Arrival Status
- Departure Status
- Invoice
- Collection

VOYAGE

Voyage is the first stage in the business process. The voyage is defined by selecting the route for the voyage; Route is combination of different sectors. Sector is defined by mentioning the start port and end port. Voyage mentions the appropriate date for the vessel to be arrived in that particular port and finally the vessel is selected for that voyage.

BOOKING

This is the second stage in the business process. After defining the voyage booking is made for different sectors in the voyage. Bookings are made for the sectors with their clients.

Normally booking is done two ways
1) Manual Booking
2) Bulk Booking

MANUAL BOOKING

Booking is done for the containers, which are existing in the voyage. The data's are entered. Booking is made for client. Manual booking is efficient for booking less number of containers. It is used for the container's left out in the bulk booking.

BULK BOOKING

Even though booking can be done manually, it takes a lot of time for bulk data. In this case, Booking is done by accessing the file which is in excel format (.xls). The data read from the file are stored in database. The data is then taken for processing.

ARRIVAL STATUS

When the vessel had arrived at a particular port arrival status is prepared for that particular port. When arrival status is prepared it gives information about the containers that are discharged at that particular port. The discharged container spaces are updated at that particular port and bookings can be done in that voyage at that port.

DEPARTURE STATUS

After arrival status is prepared departure status is prepared for that voyage in that particular port. Departure status gives the final information about the containers or goods that are loaded in the vessel for that voyage. Even booking is done, if that information is not in the departure status that booking is considered as cancelled. Departure status can be done in two ways

   i) Manual Departure

   ii) Bappilie Import.

MANUAL DEPARTURE

In manual departure containers positions i.e. (BAY, ROW, TIER) values are entered manual. It takes lot of time for completion of the process. It is efficient for fewer amounts of bookings in that port.

BAPPILIE IMPORT

Bappilie is an international file format used by the port authorities. During departure status the BAY, ROW, TIER values are added in the file and imported. The information in the Bappilie is the final one. The containers left at booking can also be added in Bappilie. After completion of the departure status, the existing booking status is final one.

INVOICE

Invoice is prepared for the client, mentioning the charges for containers that are loaded in the vessel. Charges are defined i.e. CCA (Connecting Carrier Agreement) is defined for that particular client and for that particular sector. Depending upon the charges value invoice amount is calculated and sent to the client.

COLLECTION

Depending upon the invoice value collection is made. Collection can be made at different branches which is mentioned at booking. The collection specifies the mode of collection i.e. by cheque or demand draft or by cash. And if any discount is allowed to that client it is mentioned in collection.

5.1 ENTITY – RELATIONSHIP DIAGRAM
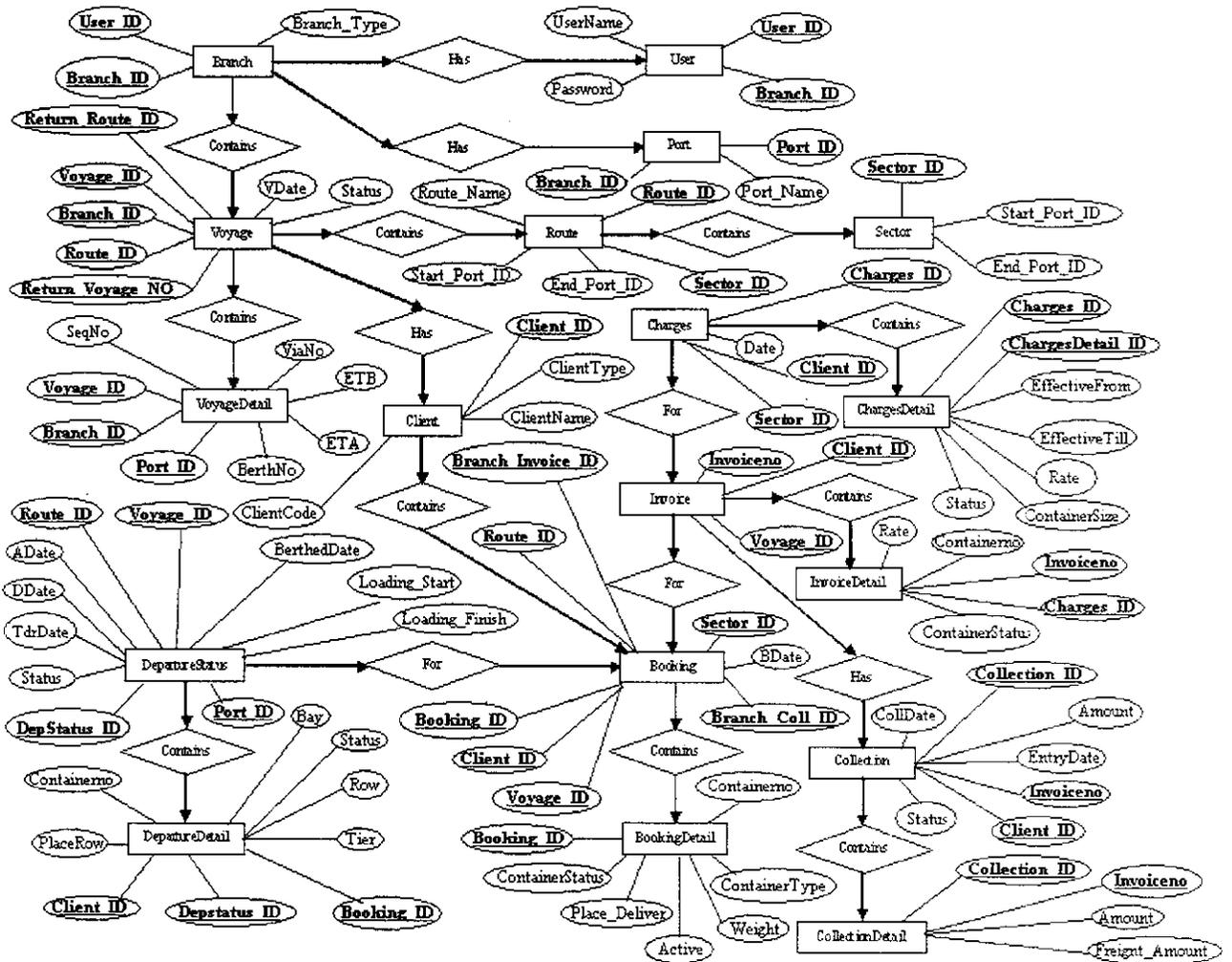
5.2 TABLE DETAILS

USER

FIELD NAME	DATA TYPE	DESCRIPTION
USER_ID	NUMBER(3)	Primary Key
BRANCH_ID	VARCHAR2(10)	Foreign Key
USERNAME	VARCHAR2(30)	
PASSWORD	VARCHAR2(30)	
DESCRIPTION	VARCHAR2(50)	

BRANCH

FIELD NAME	DATA TYPE	DESCRIPTION
BRANCH_ID	VARCHAR2(10)	Primary Key
BRANCH	VARCHAR2(15)	
BRANCHTYPE	NUMBER(1)	
ADDRESS_ID	NUMBER(5)	

PORT

FIELD NAME	DATA TYPE	DESCRIPTION
PORT_ID	VARCHAR2(10)	Primary Key
BRANCH_ID	VARCHAR2(10)	Foreign Key
PORT_NAME	VARCHAR2(5)	
DESCRIPTION	VARCHAR2(50)	

SECTOR

FIELD NAME	DATA TYPE	DESCRIPTION
SECTOR_ID	VARCHAR2(12)	Primary Key
START_PORT_ID	VARCHAR2(10)	
DISCHARGE_PORT_ID	VARCHAR2(10)	
DESCRIPTION	VARCHAR2(50)	

ROUTE

FIELD NAME	DATA TYPE	DESCRIPTION
ROUTE_ID	VARCHAR2(10)	Primary Key
ROUTE_NAME	VARCHAR2(35)	
START_PORT_ID	VARCHAR2(10)	
END_PORT_ID	VARCHAR2(10)	
DESCRIPTION	VARCHAR2(50)	

VOYAGE

FIELD NAME	DATA TYPE	DESCRIPTION
VOYAGE_ID	VARCHAR2(10)	Primary Key
BRANCH_ID	VARCHAR2(10)	Foreign Key
ROUTE_ID	VARCHAR2(10)	Foreign Key
VESSEL_ID	VARCHAR2(10)	Foreign Key
VDATE	DATE	
RETURN_ROUTEID	VARCHAR2(10)	
RETURN_VOYNO	VARCHAR2(10)	
REMARKS	VARCHAR2(100)	
STATUS	NUMBER(1)	

VOYAGE DETAIL

FIELD NAME	DATA TYPE	DESCRIPTION
VOYAGE_ID	VARCHAR2(10)	Foreign Key
BRANCH_ID	VARCHAR2(10)	Foreign Key
ROUTE_ID	VARCHAR2(10)P[;V-0	Foreign Key
PORT_ID	VARCHAR2(10)	Foreign Key
VESSEL_ID	VARCHAR2(10)	Foreign Key
BERTHNO	VARCHAR2(10)	
SEQNO	NUMBER(5)	
VIANO	VARCHAR2(10)	
ETA	DATE	
ETB	DATE	

VESSEL

FIELD NAME	DATA TYPE	DESCRIPTION
VESSEL_ID	VARCHAR2(10)	Primary Key
CLIENT_ID	VARCHAR2(10)	Foreign Key
NAME	VARCHAR2(20)	
BUILT	NUMBER(4)	
FLAG	VARCHAR2(20)	
CLASS	VARCHAR2(10)	
DWT	NUMBER(9)	
GRT	NUMBER(9)	
NRT	NUMBER(9)	
LBP	VARCHAR2(10)	
LOA	NUMBER(11,2)	
CAPACITY	NUMBER(9)	
REEFERS	NUMBER(5)	
SPEED	NUMBER(5)	

BEAM	VARCHAR2(10)	
OFF_NO	NUMBER(8)	
PLCLUB	VARCHAR2(10)	
GEARTYPE	VARCHAR2(10)	
TYPE_VESSEL	VARCHAR2(10)	
NO_OF_CREW	NUMBER(10)	
NO_OF_CRANES	NUMBER(10)	
LIFTCAPACITY	NUMBER(6)	
DRAFT_CAPACITY	NUMBER(10)	
NO_HOLDS	NUMBER(10)	
BBOX_BOW	VARCHAR2(10)	
CALLSIGN	VARCHAR2(10)	
DESCRIPTION	VARCHAR2(50)	
FEEDERCODE	VARCHAR2(5)	

BOOKING

FIELD NAME	DATA TYPE	DESCRIPTION
BOOKING_ID	NUMBER(10)	Primary Key
CLIENT_ID	VARCHAR2(10)	Foreign Key
VOYAGE_ID	VARCHAR2(10)	Foreign Key
BRANCH_ID	VARCHAR2(10)	Foreign Key
ROUTE_ID	VARCHAR2(10)	Foreign Key
BOOKING_BRANCH_ID	VARCHAR2(10)	Foreign Key
BRANCH_INVOICE_ID	VARCHAR2(10)	Foreign Key
BRANCH_COLL_ID	VARCHAR2(10)	Foreign Key
SECTOR_ID	VARCHAR2(12)	Foreign Key
BDATE	DATE	
BOOKINGSTATUS	CHAR(1)	
REMARKS	VARCHAR2(100)	

BOOKING DETAIL

FIELD NAME	DATA TYPE	DESCRIPTION
BOOKING_ID	NUMBER(10)	Foreign Key
IMCODE	NUMBER(4)	
UNCODE	NUMBER(4)	
CONTAINERSIZE	NUMBER(3)	
CONTAINERNO	VARCHAR2(15)	
CONTAINERSTATUS	CHAR(1)	
CONTAINERTYPE	VARCHAR2(60)	
PLACE_DELIVER	VARCHAR2(20)	
OOG_ID	NUMBER(4)	
REEFER_VOLT	NUMBER(10)	
WEIGHT	NUMBER(10)	

OOG	CHAR(1)	
DESCRIPTION	VARCHAR2(100)	
ACTIVE	CHAR(1)	
OHEIGHT	NUMBER(5)	
OLENGTH	NUMBER(5)	
OWIDTH	NUMBER(5)	
TRANSPORT	NUMBER(1)	
TEMP_UNIT	CHAR(1)	
OLENGTH1	NUMBER(5)	
OWIDTH1	NUMBER(5)	

CLIENT

FIELD NAME	DATA TYPE	DESCRIPTION
CLIENT_ID	VARCHAR2(10)	Primary Key
CLIENTTYPE	VARCHAR2(10)	
ADDRESS_ID	NUMBER(5)	Foreign Key
CLIENTNAME	VARCHAR2(50)	
CLIENTCODE	VARCHAR2(10)	
DESCRIPTION	VARCHAR2(50)	

ADDRESS

FIELD NAME	DATA TYPE	DESCRIPTION
ADDRESS_ID	NUMBER(5)	Primary Key
ADDRESS1	VARCHAR2(50)	
ADDRESS2	VARCHAR2(80)	
CITY	VARCHAR2(40)	
STATE	VARCHAR2(20)	
COUNTRY	VARCHAR2(30)	
PINCODE	VARCHAR2(15)	
EMAIL1	VARCHAR2(200)	
EMAIL2	VARCHAR2(200)	
PHONE1	VARCHAR2(15)	
PHONE2	VARCHAR2(15)	
FAX1	VARCHAR2(15)	
FAX2	VARCHAR2(15)	
MOBILE	VARCHAR2(15)	
PORT_ID	VARCHAR2(10)	
STATUS	CHAR(1)	
BRANCH_ID	VARCHAR2(10)	
SECTOR_ID	VARCHAR2(10)	

DEPATURE DETAIL

FIELD NAME	DATA TYPE	DESCRIPTION
CONTAINERNO	VARCHAR2(15)	
PLACED_ROW	NUMBER(4)	
BAY	NUMBER(4)	
TIER	NUMBER(4)	
STATUS	NUMBER(1)	
CLIENT_ID	VARCHAR2(10)	Foreign Key
DEPSTATUS_ID	NUMBER(10)	Primary Key
PORTSTATUS	NUMBER(1)	
BOOKING_ID	NUMBER(10)	

DEPATURE STATUS

FIELD NAME	DATA TYPE	DESCRIPTION
DEPSTATUS_ID	NUMBER(10)	Foreign Key
VOYAGE_ID	VARCHAR2(10)	Foreign Key
ROUTE_ID	VARCHAR2(10)	Foreign Key
PORT_ID	VARCHAR2(10)	Foreign Key
ADATE	DATE	
DDATE	DATE	
BERTHEDDATE	DATE	
TDRDATE	DATE	
UNLOADING_START	DATE	
UNLOADING_FINISH	DATE	
LOADING_START	DATE	
LOADING_FINISH	DATE	
ARRIVAL_AVG_SPEED	VARCHAR2(10)	
SAILEDON	DATE	
REQUIRED_HFO	NUMBER(7,2)	
REQUIRED_IFO	NUMBER(7,2)	
REQUIRED_MGO	NUMBER(7,2)	
REQUIRED_FW	NUMBER(7,2)	
REQUIRED_BALLAST	NUMBER(7,2)	
CONSUMED_HFO	NUMBER(7,2)	
CONSUMED_IFO	NUMBER(7,2)	
CONSUMED_MGO	NUMBER(7,2)	
CONSUMED_FW	NUMBER(7,2)	
CONSUMED_BALLAST	NUMBER(7,2)	
POB	DATE	
A_HFO	NUMBER(7,2)	
A_BALLAST	NUMBER(7,2)	
A_IFO	NUMBER(7,2)	

D_IFO	NUMBER(7,2)	
D_BALLAST	NUMBER(7,2)	
D_MGO	NUMBER(7,2)	
D_FW	NUMBER(7,2)	
SHIFT_REMARKS	VARCHAR2(200)	
STATUS	NUMBER(1)	
SUPPLIED_HFO	NUMBER(7,2)	
SUPPLIED_IFO	NUMBER(7,2)	
SUPPLIED_MGO	NUMBER(7,2)	
SUPPLIED_FW	NUMBER(7,2)	
SUPPLIED_BALLAST	NUMBER(7,2)	

INVOICE

FIELD NAME	DATA TYPE	DESCRIPTION
INVOICENO	VARCHAR2(15)	Primary Key
CLIENT_ID	VARCHAR2(10)	Foreign Key
SECTOR_ID	VARCHAR2(12)	Foreign Key
BRANCH_INVOICE_ID	VARCHAR2(12)	Foreign Key
CHARGES_ID	VARCHAR2(10)	Foreign Key
VOYAGE_ID	NUMBER(10)	Foreign Key
BRANCH_ID	VARCHAR2(10)	Foreign Key
ROUTE_ID	VARCHAR2(10)	Foreign Key
BOOKING_ID	NUMBER(10)	Foreign Key
INVOICE_DATE	DATE	
PAYABLEBY	DATE	
INVOICE_STATUS	CHAR(1)	
COLL_STATUS	CHAR(1)	
AMOUNT	NUMBER(10,2)	
DISCOUNT	NUMBER(8,2)	
DESCRIPTION	VARCHAR2(200)	
BRANCH_COLL_ID	VARCHAR2(10)	Foreign Key
DESCRIPTION	VARCHAR2(250)	

INVOICE DETAIL

FIELD NAME	DATA TYPE	DESCRIPTION
INVOICENO	VARCHAR2(15)	Foreign Key
CHARGESDETAIL_ID	NUMBER(10)	Foreign Key
CONTAINERNO	VARCHAR2(15)	
RATE	NUMBER(8,2)	
CONTAINERSIZE	NUMBER(3)	
CONTAINERSTATUS	CHAR(1)	
REMARKS	VARCHAR2(100)	

CHARGES

FIELD NAME	DATA TYPE	DESCRIPTION
CHARGES_ID	NUMBER(10)	Primary Key
INVOICE_BRANCH_ID	VARCHAR2(10)	Foreign Key
COLLECTION_BRANCH_ID	VARCHAR2(10)	Foreign Key
CLIENT_ID	VARCHAR2(10)	Foreign Key
SECTOR_ID	VARCHAR2(12)	Foreign Key
OOGPERCENT	NUMBER(3)	
CHARGESDATE	DATE	
CREDIT_PERIOD	NUMBER(2)	
EFFECTIVEFROM	DATE	
EFFECTIVETILL	DATE	
DESCRIPTION	VARCHAR2(200)	
SUPPORT	NUMBER(5)	
STATUS	NUMBER(1)	
PAYMENTTERMS	VARCHAR2(5)	
VOYAGE_ID	VARCHAR2(10)	Foreign Key
BRANCH_ID	VARCHAR2(10)	Foreign Key
ROUTE_ID	VARCHAR2(10)	Foreign Key
INVOICETYPE	VARCHAR2(20	
SEALCONTENT	VARCHAR2(150)	

CHARGESDETAIL

FIELD NAME	DATA TYPE	DESCRIPTION
CHARGESDETAIL_ID	NUMBER(10)	Primary Key
CHARGES_ID	NUMBER(10)	Foreign Key
SUBTYPE	NUMBER(4)	
GROUP	NUMBER(4)	
OOG	NUMBER(4)	
SHIPMENTTYPE	CHAR(1)	
CONTAINERSIZE	NUMBER(3)	
CONTAINERSTATUS	CHAR(1)	
SLABSTARTPT	NUMBER(5)	
OOGPERCENT	NUMBER(3)	
RATE	NUMBER(8,2)	
SLABENDPT	NUMBER(5)	
STATUS	NUMBER(1)	
EFFECTIVEFROM	DATE	
EFFECTIVETILL	DATE	

COLLECTION

FIELD NAME	DATA TYPE	DESCRIPTION
COLLECTION_ID	NUMBER(10)	Primary Key
CLIENT_ID	VARCHAR2(10)	Foreign Key
COLL_DATE	DATE	
AMOUNT	NUMBER(12,2)	
BANK	VARCHAR2(25)	
CHEQUENO	VARCHAR2(25)	
CHEQUEDATE	DATE	
STATUS	CHAR(1)	
INVOICENO	VARCHAR2(15)	Foreign Key
MODEOFPAYMENT	NUMBER(1)	
REMARKS	VARCHAR2(100)	
ENTRYDATE	DATE	

COLLECTIONDETAIL

FIELD NAME	DATA TYPE	DESCRIPTION
SERIALNUMBER	NUMBER(8)	Primary Key
COLLECTION_ID	NUMBER(10)	Foreign Key
INVOICENO	VARCHAR2(15)	Foreign Key
AMOUNT	NUMBER(12,2)	
FREIGHT_AMOUNT	NUMBER(12,2)	
PHC_AMOUNT	NUMBER(12,2)	
DESCRIPTION	VARCHAR2(100)	

6. IMPLEMENTATION DETAILS

This System is developed using Cold Fusion, as it is one of the best available Graphical User Interface (GUI), enables to create a Website that is dynamic, fast and Interactive without requiring the clients to worry about the capabilities of browsers. Cold Fusion is one of the best available Web based User Interface environments.

REASONS FOR SELECTING COLD FUSION

Cold Fusion is a Web application development tool that enables the rapid creation of interactive, dynamic, and information-rich Web sites. Cold Fusion does not require coding in traditional programming languages. Instead, creating applications by extending your standard HTML files with high level formatting functions, conditional operators, and database commands. These commands are instructions to the Cold Fusion processor and form the building blocks on which to build industrial strength applications. This method of creating Web applications has significant advantages over conventional application development.

> ➢ Cold Fusion applications can be developed rapidly because no coding is required other than use of simple HTML style tags.
> ➢ Cold Fusion applications are easy to test and roll out.
> ➢ The Cold Fusion language contains all the processing and formatting functions you'll need (and the ability to create your own functions if you really run into a dead end).
> ➢ Cold Fusion applications are easy to maintain because there is no compilation or linking step, so the files you create are the files used by Cold Fusion.
> ➢ Cold Fusion provides all the tools you need to be able to troubleshoot and debug applications.
> ➢ Cold Fusion comes with all the hooks needed to link to almost any database application.
> ➢ Cold Fusion is fast.

THE DYNAMIC PAGE ADVANTAGE

Linking your Web site to live data is a tremendous advantage, but the benefits of database interaction go beyond extending the capabilities of your Web site. With Cold Fusion you can create dynamic, data-driven Web pages. Dynamic Web pages are becoming the norm, and for a good reason.

STATIC WEB PAGES

Static Web pages are made up of text and images, and HTML formatting tags. These pages are manually created and maintained so when information changes, so must the page. This usually involves loading the page into an editor, making the changes, reformatting text if needed, and then saving the file. Of course, not everyone in the organization can make these changes. The Webmaster, or Web design team, is responsible for maintaining the site and implementing all changes and enhancements. This often means that by the time information finally makes it onto the Web site, it is out of date.

DYNAMIC WEB PAGES

Dynamic Web pages contain very little actual text. Instead, they pull needed information from other applications. Dynamic Web pages communicate with databases to extract information.

FEATURES OF COLDFUSION

Cold Fusion provides you with a full range of database interaction functions to create complete dynamic, data-driven Web pages. The features include:

- ➤ The ability to query existing database applications for data.
- ➤ The ability to create dynamic queries facilitating more flexible data retrieval.
- ➤ The ability to execute stored procedures in databases that support them.
- ➤ The ability to execute conditional code on-the-fly, to customize responses for specific situations.
- ➤ The ability to enhance the standard HTML form capabilities with data validation functions.
- ➤ The ability to customize the display of dates, times, and currency values with formatting functions.

REASONS FOR SELECTING JAVASCRIPT

JavaScript is a compact, object-based scripting language for developing client and server Internet applications. Browser interprets JavaScript statements embedded in an Webpage, and Livewire enables you to create server-based applications similar to Common Gateway Interface (CGI) programs. JavaScript is an easy to use object-scripting language designed for creating live online applications that link together objects and resources on both client and server. JavaScript is designed for use by HTML page authors and enterprise application developers to dynamically script the behavior of objects running on either a client or server. Netscape develops JavaScript and Microsoft develops Jscript. With JavaScript we can easily create interactive web pages.

Java script is

 ➤ Designed to create net-centric application
 ➤ Complementary to and integrate with java
 ➤ Complementary to and integrate with HTML
 ➤ Open and cross platform.

REASONS FOR SELECTING ORACLE 8.0

A database management system is essentially a collection of interrelated data and set of programs to access the data. This collection of data is called the database. RDBMS is the acronym for Relational Database Management System. Oracle products are based on a concept known as the 'Client Server Technology'. This concept involves segregating the processing of an application between two systems. One performs all the activities related to the database (server) and the other performs activities that help the user to interact with the application (client). A client or front end database application also interacts with the database by requesting and receiving information from the 'database server'. It acts as an interface between the user and the database. Further it also checks for validation against the data entered by the user. The database server or back end is used to manage the database tables optimally among multiple clients who concurrently request the server for the same data. It also enforces data integrity across all client applications and

Oracle provides various tools for application development and for performing administrative functions:

> Oracle Enterprise Manager (OEM)
> SQL*Plus
> PL/SQL
> Net8
> Precompilers

SQL*Plus

The only interface available between end users and an RDBMS is Structured Query Language (SQL). All other applications and tools that users utilize to interact with the RDBMS act as translators/interpreters. These tools generate SQL commands based on a user's request and pass the generated SQL commands on to the RDBMS.

SQL*Plus, Oracle's version of SQL, is one of the most commonly used Oracle tools. SQL*Plus enables users to instruct the Oracle instance to perform the following SQL functions:

> Data definition or DDL operations, such as creating, altering, and dropping database objects
> Data query to select or retrieve the stored data
> Data manipulation or the DML operations to insert, update, and delete data
> Access and transfer data between the databases
> Allow user to enter data interactively
> DBA functions or the database administrative tasks such as managing users (creating, altering, and dropping users), managing space (creating, altering, and dropping table spaces), and backup and recovery

7. TESTING

Testing is an important process, which leads to the success of the system. System is mainly performed with the intention of finding errors to give the client an error free system. System testing makes a logical assumption that all parts of the system are correct and move towards it to make the system error free.

Inadequate testing of the system will lead to errors, which even can arise after long time of system implementation. The purpose of system testing is to consider all the likely variations, and then push the system to its limits and achieve the results. The various testing done were,

7.1 UNIT TESTING

Each and every program needed to be tested, which is known as unit testing. The module interface is to be tested to ensure that information properly flows in and out of the unit under test. The local data structure is to be examined to ensure that data stored temporarily maintains integrity throughout the unit's lifetime. Boundary condition is tested thoroughly. All independent paths through the control structure are to be exercised to ensure that all the statements in the modules have been executed at least once. All error-handling paths are to be tested.

TEST CASE

- ☑ Testing of improper formats of data entered.
- ☑ Testing for incorrect values for the fields, which are below the range or above the range.
- ☑ Testing date fields by giving date values that is above or below the range.
- ☑ Testing the behavior when null values are entered

7.2 INTEGRATION TESTING

Once the modules are tested individually under the unit testing strategy, it is necessary to put all these modules together. It is here that the data can be lost across the interface. One module can have an inadvertent, adverse effect on another.

Integration testing is a systematic technique for constructing the program structure while at the same time conducting test to uncover errors associated with interfacing. The objective is to take unit-testing modules and build a program

TEST CASE

☑ The navigation between different blocks has been tested for any inconsistencies.

☑ The block is tested in various modes such as insert, delete and updating entries.

☑ Also it has been ensured that the changes done in one module are reflected in all the other modules.

TEST CASE

COMMON FUNCTIONALITY TO ALL SCREENS

SL.NO	TEST CASE	RESULT
1	Click button in the screen	Enters the appropriate screen
2	Invalid Date input	Returns an alert message asking the user to enter in the dd/mm/yyyy format
3	On Loading combo boxes	All data has to be loaded relating particular box, and sorted
4	Invalid Numeric input	Returns an alert message
5	NOT NULL check	Returns an alert message
6	Length Check for all text fields	Not allow to insert beyond the length
7	Click on Hyper Links	Loads appropriate screens

8. CONCLUSION

To achieve higher speed and ultimately the speed of light has been the aim of those in information Technology. Faster microchips, faster methods always keep cropping now and then. Keeping this in mind this software has been developed. The sole aim is to ease the burden on the user and enable the user to perform complex queries in a very less time.

PROJECT ACHIEVMENTS

The system entitled "Feeder Management System" is a very generic application that can be used to solve the various problems. This software is user friendly and the users can view the various transactions from any part of the world. With the help of the various reports the Administration is able to check the various business transactions taking place. The screens are being designed in such a way that there is minimum input and a lot of choice inputs. The forms are designed in a user-friendly manner.

The system is able to make the information available to the right person at the right time. With the help of reports the administration is able to take decisions about the company's Development.

9. REFERENCES

- ➢ Arman Danesh and Kristin Aileen Motlagh, " **Mastering Cold Fusion 4.0** " First Edition, Sybex Inc 2000
- ➢ Charles Mohnnike , "**SAMS Teach Yourself Allaire's Cold fusion in 21 Days**", First Edition, Sams Publishing , 2000
- ➢ "**Allaire's COLDFUSION 4.0**" E-book
- ➢ "**Oracle 8 The Complete Reference**" – E-Book
- ➢ "**Teach Yourself Oracle 8 in 21 Days**" – E-Book.

10. OUTPUT AND SAMPLE SCREENS

LOGIN SCREEN
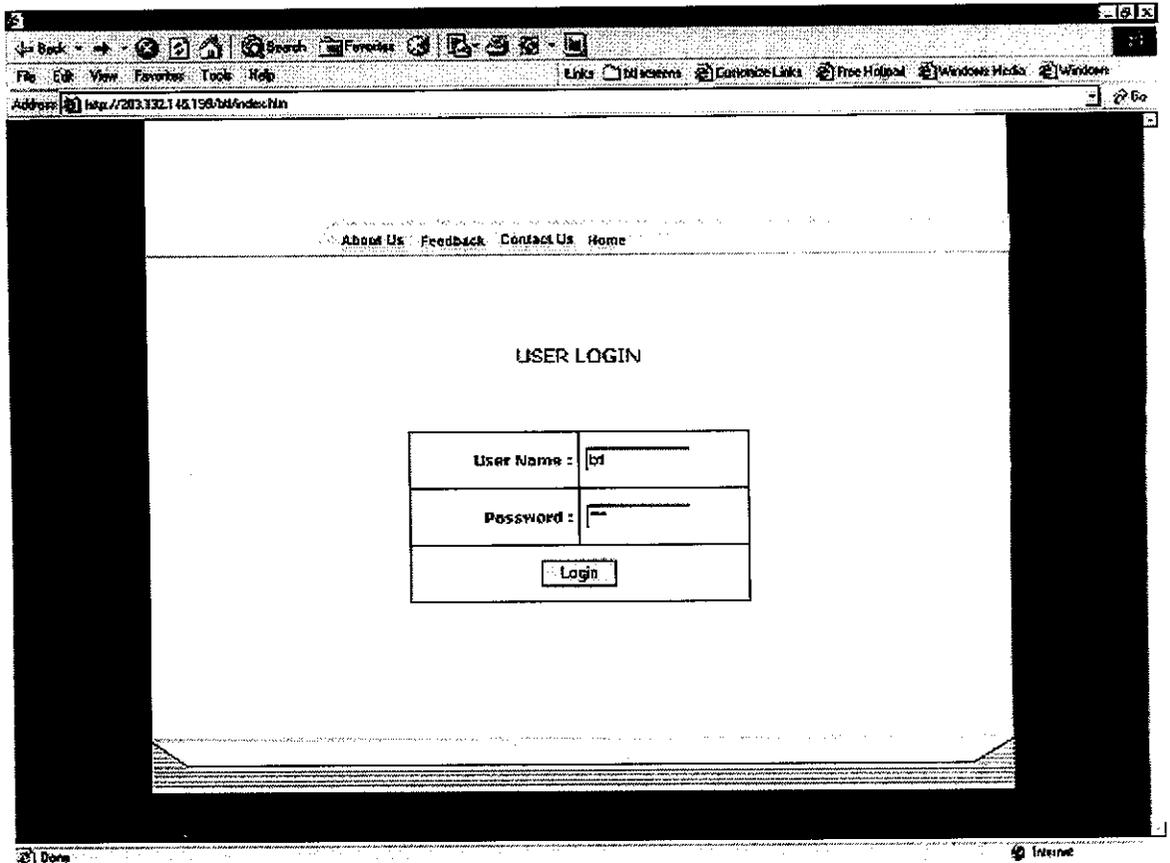
This is the login screen. After verifying the user name and password the user allowed to use the software. The user can view only his related information. Unauthorized users can be avoided.

MAIN SCREEN
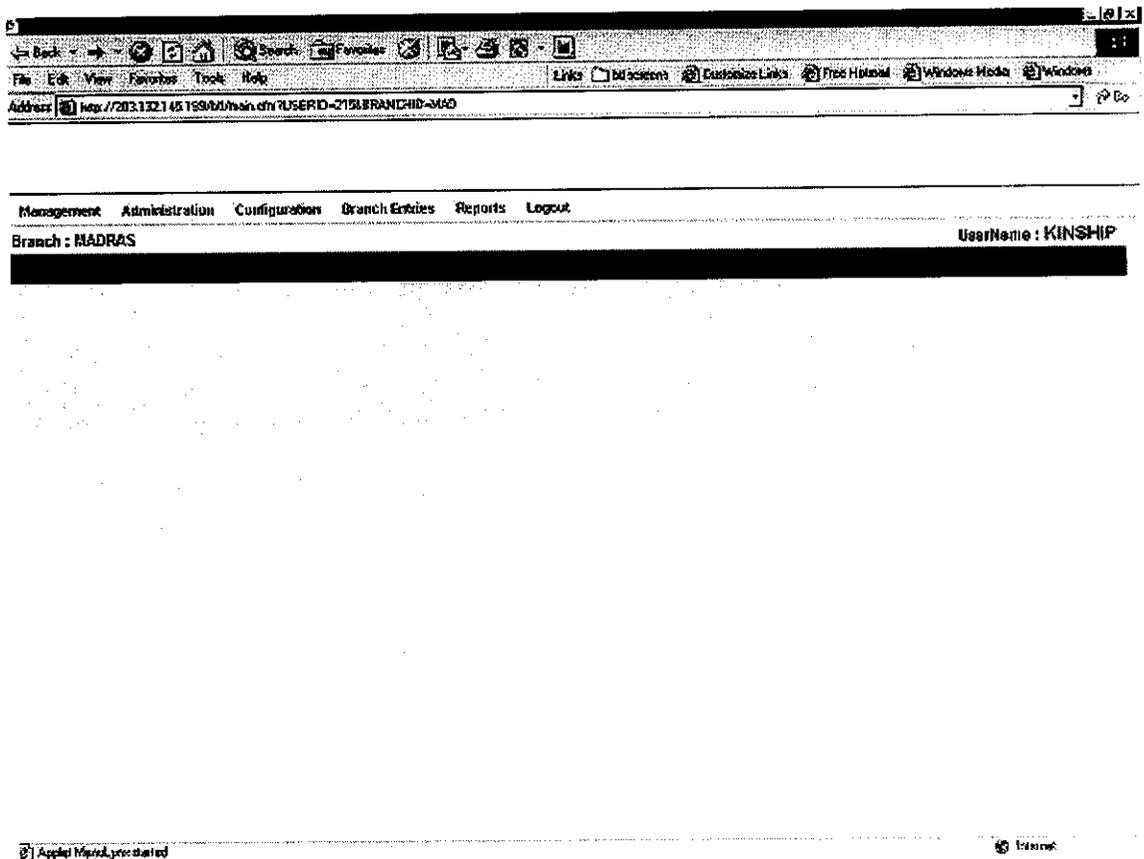
This is the main screen. The main screen contains a menu. All the operations to be performed can be selected from the menu.

BRANCH

The screen uses to add a branch. The branch details are entered here and stored. Similarly screens are designed to delete, modify and view the branch details.

USER

This Screen used to add user. Similarly we can modify, delete and view the users.

PORT

This screen uses to add port information. Options are provided to modify, delete and view the port information

SECTOR
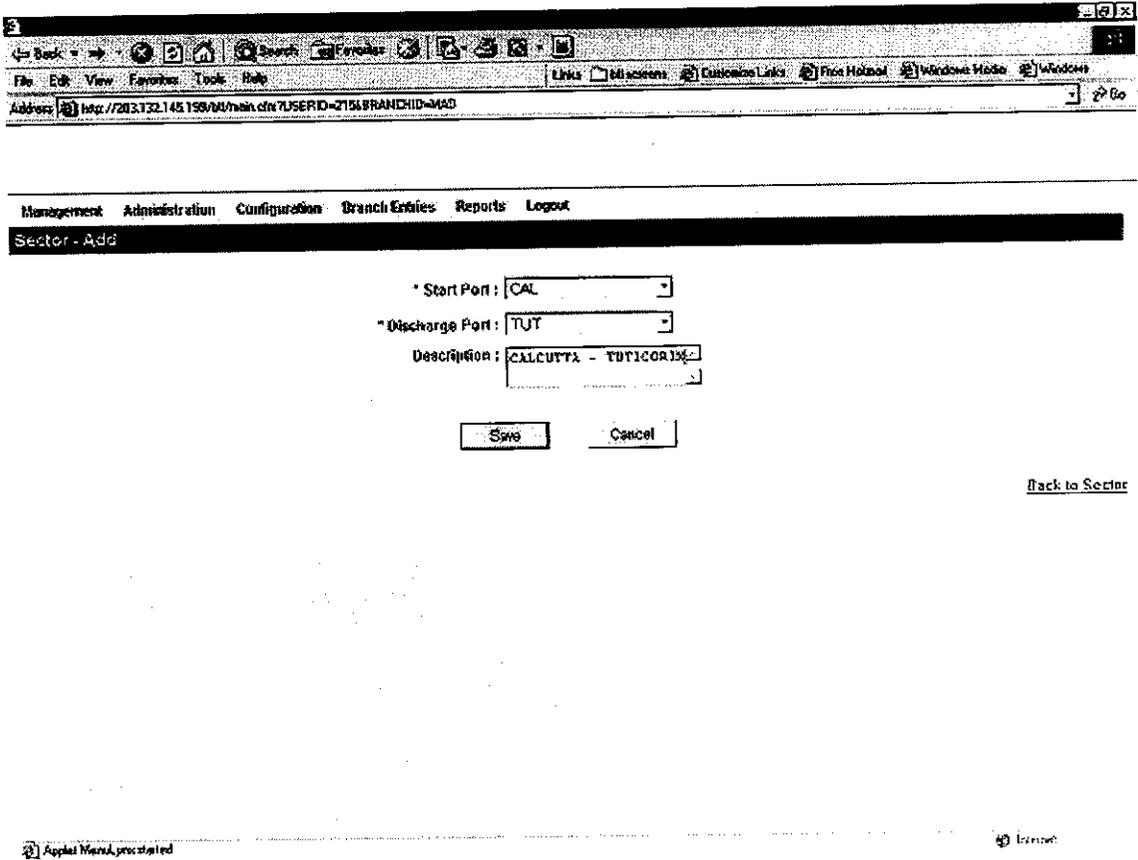
This screen defines the sector combinations by selecting the ports.

ROUTE
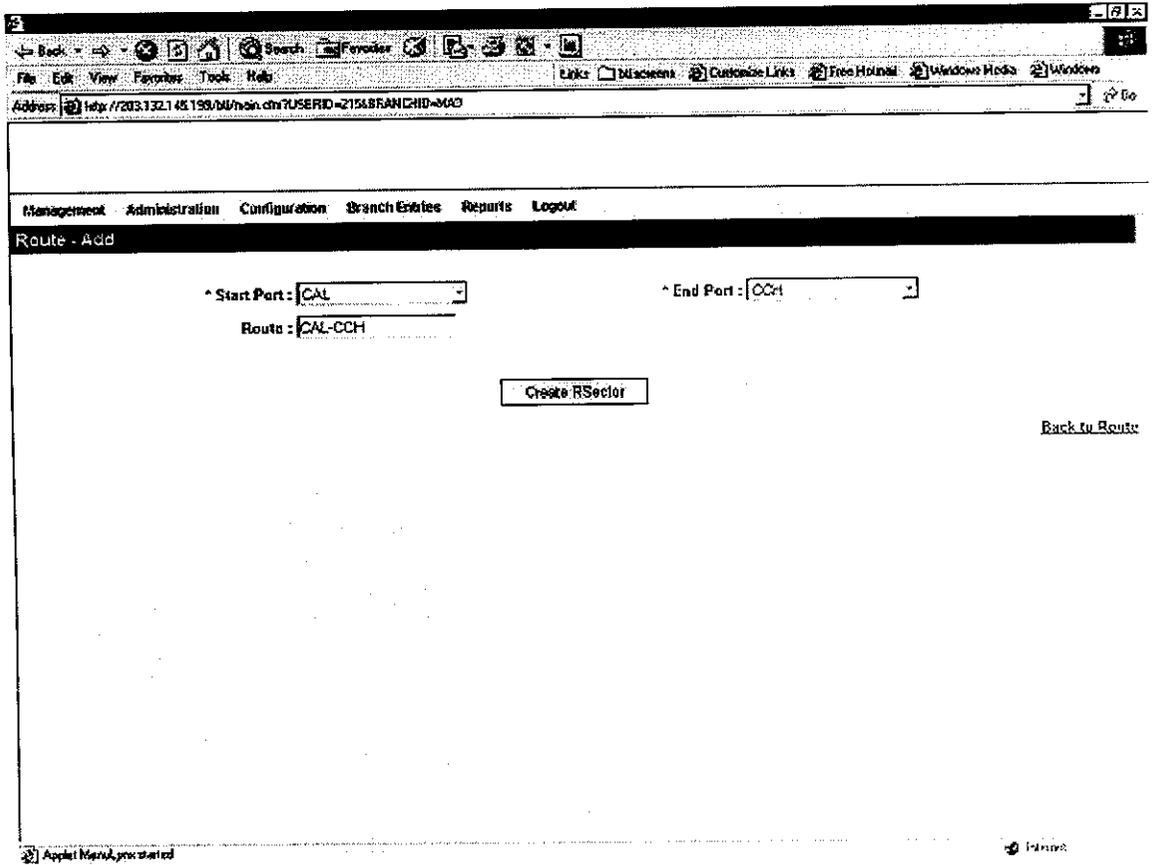
Route is defined by selecting the start port and end port.

ROUTE SECTOR
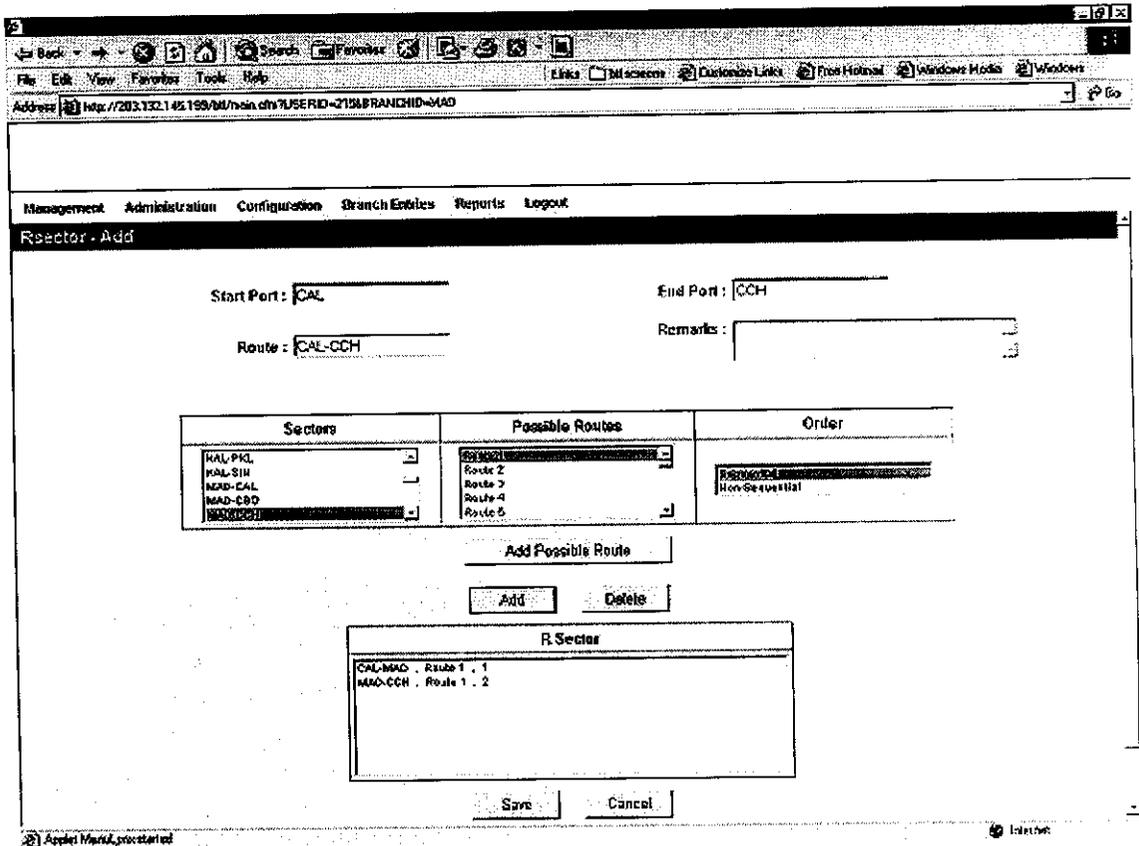
Route Sector is combination of different sectors. The sector combination defines via of the route.

CLIENT

This screen is used to add the client information. Options are provided for deleting, modifying and viewing the client information.

VESSEL

This screen adds the vessel information. Options are provided for deleting, modifying and viewing the vessel information.

VOYAGE

This screen is used to add the new voyage. The information such as vessel used, on which date the voyage is going to start are given as input.

SCREEN 2:

BOOKING

This screen adds a new container booking in the voyage. The information given are container detail, sector on which booking is made, invoice branch and collection branch and for which client the booking is made.

ARRIVAL STATUS

When the vessel arrives at a port, arrival status is prepared. The arrival status informs the containers that are going to be discharged at that particular port.

DEPARTURE STATUS

Departure status is prepared when the vessel is ready for departure. The screen provides to add the departure information such as container positions and so on.

SCREEN 2:

CCA - CHARGES

CCA defines the charges for the client booking that exists in the voyage. Depending upon the charges value invoice is generated.

SCREEN 2:

INVOICE

Invoice is generated for the booking in the voyage. Invoice value is generated depending upon the charges value.

COLLECTION

Invoice amount is to be collected. This makes an entry when the amount is collected. This contains information such as invoice no and amount.

REVENUE REPORT

Revenue Report is a special form of report, which is generated after departure status; this gives information about revenue which is going to be got, with this voyage.

CCA REPORT

CCA Report

Page No : 1

Charges_ID : 38   Date : 20/12/2002
Client : CUENTRAJ   Description : MADRAS CLIENT
Sector : MAD-CBO   Credit Days : 0
Collection Branch : MAD   Invoice Branch : MAD
Effective From : 20/12/2002   Effective Till : 19/01/2003
CCA Type : FLAT CCA   Status : Active

Shipment	Container Size	Container Status	Start Point	End Point	Group	OOG Type	OOG %	Status	Charge Type	Rate (in USD)
Transhipment	20	LADEN	--	--	NORMAL	LF-LB-WR-WL-H	3	Active	FRT	1200.00

This report mentions the cca information for the clients and sector.

TERMINAL DEPARTURE REPORT:

This report gives the details about the container's that are loaded in that port.

SCREEN 2:

The browser window shows the following navigation menu and table:

Management Administration Configuration Branch Entries Reports Logout

LOADING SUMMARY(COLOMBO)

MLO'S	LADEN			REEFER			EMPTY			TOTAL			WEIGHT
	20"	40"	45'	20'	40"	45'	20'	40'	45'	20"	40'	45"	
A/C : BTL													
3	1	0	0	0	0	0	0	0	0	1	0	0	1,200.00
TOTAL	1	0	0	0	0	0	0	0	0	1	0	0	1,200.00

TOTAL UNITS : 1
TOTAL TEUS : 1

TERMINAL ARRIVAL REPORT:

This report is generated after preparing arrival status. This gives the information about containers that are ready to discharge at that particular port.