

**THE  
TRANSPORT MANAGEMENT SYSTEM**

**Submitted in partial fulfillment of the requirements for the degree of  
M.TECH**

**SubmittedBy  
Deepshikha**

**Under the Supervision  
Mr.SachinSuryan  
( HODin CSE Deptt.)**



**Department of Computer Science and Engineering**

**MERI COLLEGE OF ENGINEERING AND TECHNOLOGY**

**SAMPLA,ROHTAK**

**(May, 2017)**

## **Introduction 1**

---

### **1.1 ABOUT THESIS**

As we know every transport sector has it's own paper work which can be done by the company employee. Transportation means it has lots of paper work in their branches. Transportation is a business and thus it cannot be handled by one person and there are many branches of a transport company. In the existing file system it's all paper work which is done manually by the staff. Operating everything manually becomes very time consuming, difficult to handle, not reliable as there are majority of chances of error. It may also lead to major loss of data due to manual operation. Retrieving records of company is also difficult.

The objective of making transport management system is to make the task easier of handling all the bills of the customer. In this project all the database of the customer is stored in the memory of computer so that when the next time customer places the order then computer will automatically redeem the basic information of the customer. The customer can make the necessary changes according to his or her requirements.

### **1.2 OBJECTIVES & SCOPE OF THE THESIS**

- Even a non-technical user can use the system effectively and without any difficulty.
- It is upgradeable. Hence further enhancement to the system is possible.
- The system maintains speed and accuracy with eye-catching GUI.

### **1.3 GOALS OF THE THESIS**

- Applying information Technology in the Transport Business.
- To provide window based, user friendly and efficient system to the staff and the clients.
- Implement security by preventing unauthorized access.

- This System is window based, user friendly, and efficient.
- System security is implemented by preventing unauthorized access to the system.
- Appropriate master tables are present to record relevant data.
- Backups are provided to prevent data loss.
- All the calculations are done automatically and thus no chances of errors.
- Reports are generated for the following :-
  - Docket Number
  - Lorry Hire Cum Challan
  - Crossing Challan
  - Add Station
  - Deal
  - Monthly Reports

#### **1.4 FEATURES OF SOFTWARE (VISUAL STUDIO 2013)**

In Visual Studio 2013, new features have been introduced that boost productivity and save time when working inside the Editor. Some of these are new features and some are the most popular extensions from Productivity Power Tools. Primary focus for the Editor in this version is to keep the developer in context as much as possible.

##### **Peek Definition (Alt + F12)**

Developers move around code a lot when browsing definitions while staying in context of their code. Peek Definition is one such feature that allows you to view definitions inline in the Editor without having to open a new document tab. To see it in action, right click on a symbol and click on the “Peek Definition” command in the context menu or invoke the keyboard shortcut Alt + F12. Using Peek Definition saves you time when understanding a code base because it allows you to view definitions without losing sight of the code that you are already in. You can even browse through a series of definitions without leaving the original file. When you invoke the Peek Definition command inside an existing Peek view, we navigate you to the definition inside the Peek view and add a breadcrumb for the previous Peek view. These breadcrumbs help you visualize the path you take to reach any given definition.

**Enhanced Scrollbar**

The Enhanced Scrollbar has been one of the most popular Productivity Power Tools extensions and is now part of the Visual Studio 2013 product! The Enhanced Scrollbar provides you visual cues about your file on the vertical scrollbar. Markers on the scrollbar allow you to quickly view the location of errors, warnings, breakpoints, bookmarks, find results and other useful information in your file. Here, too, we focus on bringing information to your fingertips – you don't have to scroll away from your current position to get information about other parts of your file.

**Auto Brace Complete**

The Auto brace complete feature automatically adds closing parenthesis, quotes, braces and brackets for you as you type code in the Editor. This was one of the popular Productivity Power Tools Features that we decided to include in the Visual Studio 2013.

Move Line Up/ Down (Alt Arrow-Up/ Arrow-Down)

Move Line Up/ Down feature allows you to quickly move single or multiple lines up and down using the keyboard shortcut Alt + Up Arrow and Alt + Down Arrow.

## Literature Survey 2

---

### 2.1 EXISTING SYSTEM

The existing system is based on File Management.

#### 2.1.1 Limitations of Existing System

- **Inability of modification of data.:-** as the data is stored in the registers due to which modification of the data becomes very complex or we can say that inability of modification of data.
- **Not user friendly.:-** as data is stored in the register which makes existing system non user friendly i.e data becomes complex. Hence its not a user friendly.
- **Security issues.:-** in existing system as the data is stored in the register which can be easily accessible by anyone. Anyone can open up the register and can acces the data.
- **Lots of paperwork. :-** in existing system all the data is stored in the register due to which so many documents get piled up i.e lots of paperwork get collected.
- **Manual operator control.:-** this can be opertated by any of the person easily.
- **Time consuming.:-**its really time consuming as to search the record you have to search each and every document.
- **More chances of errors.:-** all the record is documented manually which increases more chances of errors.
- **Chances of data loss:-**in this as all the data is collected in registers due to which chances of data loss can take place.
- Due to manual work existing system is very time consuming and there is always a chance of error so that it is not much reliable.
- All billing calculation done by calculator/EXCELL SHEETS which may also generate an error.

- Remembering all the rates and quantity of each item is very difficult for this they have to turn over the page of the register/SHEETS.

## **2.2 PROPOSED SYSTEM**

In this we overcome the limitations of existing system.

- System should be window based, user friendly, and efficient.
- System security must be implemented by preventing unauthorized access to it.
- Appropriate master table must be present to record relevant data.
- Security feature must exist in the system.
- All the calculations are done automatically and thus no chances of errors.

### **2.2.1 Advantages of Proposed System**

- This system has been developed to be simple and user friendly.
- Hence, even a non-technical user can use the system effectively and without any difficulty.
- It is upgradeable. Hence further enhancement to the system is possible.
- The system is capable of generating the necessary report.
- The system maintains speed and accuracy with eye-catching GUI.
- It take care of database security by restricting unauthorized access to the system

## Requirements 3

---

### 3.1 GENERAL REQUIREMENTS

#### 3.1.1 Microsoft Visual Studio 2013

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silver light. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a forms designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that enhance the functionality at almost every level—including adding support for source-control systems (like Subversion) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Team Foundation Server client: Team Explorer).

Visual Studio supports different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C, C++ and C++/CLI (via Visual C++), VB.NET (via Visual Basic .NET), C# (via Visual C#), and F# (as of Visual Studio 2010). Support for other languages such as M, Python, and Ruby among others is available via language services installed separately.

It also supports XML/XSLT, HTML/XHTML, JavaScriptandCSS. Java (and J#) were supported in the past.

### 3.1.2 Microsoft SQL Server 2008 r2

SQL Server 2008 R2 (10.50.1600.1, formerly codenamed "Kilimanjaro") was announced at TechEd 2009, and was released to manufacturing on April 21, 2010. SQL Server 2008 R2 adds certain features to SQL Server 2008 including a master data management system branded as Master Data Services, a central management of master data entities and hierarchies. Also Multi Server Management, a centralized console to manage multiple SQL Server 2008 instances and services including relational databases, Reporting Services, Analysis Services & Integration Services.

SQL Server 2008 R2 includes a number of new services, including PowerPivot for Excel and SharePoint, Master Data Services, StreamInsight, Report Builder3.0, Reporting Services Add-in for SharePoint, a Data-tier function in Visual Studio that enables packaging of tiered databases as part of an application, and a SQL Server Utility named UC (Utility Control Point), part of AMSM (Application and Multi-Server Management) that is used to manage multiple SQL Servers.

The first SQL Server 2008 R2 service pack (10.50.2500, Service Pack 1) was released on July 11, 2011.

The second SQL Server 2008 R2 service pack (10.50.4000, Service Pack 2) was released on July 26, 2012.

The final SQL Server 2008 R2 service pack (10.50.6000, Service Pack 3) was released on September 26, 2014.

## 3.2 FUNCTIONALITIES

Transportation management systems manage four key processes of transportation management:

- **Planning and decision making** – TMS will define the most efficient transport schemes according to given parameters, which have a lower or higher importance according to the user policy: transport cost, shorter lead-time, fewer stops possible to ensure quality, flows regrouping coefficient, etc.
- **Transportation Execution** – TMS will allow for the execution of the transportation plan such as carrier rate acceptance, carrier dispatching, EDI etc..



- **Transport follow-up** – TMS will allow following any physical or administrative operation regarding transportation: traceability of transport event by event (shipping from A, arrival at B, customs clearance, etc.), editing of reception, custom clearance, invoicing and booking documents, sending of transport alerts (delay, accident, non-forecast stops...)
- **Measurement** – TMS have or need to have a logistics key performance indicator (KPI) reporting function for transport.

### 3.3 INTERFACE REQUIREMENTS

#### 3.3.1 Hardware Requirements

- Processor : Intel(R) Core(TM) i5
- Clock Speed : 2.59 GHZ Processor
- Hard disk : 2GB
- Ram : 6GB
- Keyboard : any
- Mouse : any

#### 3.3.2 Software Requirements

- Operating System : Windows 98/2000/Xp/7/Vista/8/8.1/Any
- Back End : Microsoft Sql Server 2008
- Front End : Microsoft Visual Studio 2013
- Programming Language : C#

### 3.4 NON-FUNCTIONAL REQUIREMENTS

- **Security** - is the degree of resistance to, or protection from, harm. It applies to any vulnerable and valuable asset, such as a person, dwelling, community, nation, or organization.

- **Reliability** - the ability of a system or component to perform its required functions under stated conditions for a specified period of time.
- **Availability** - Availability of a system is typically measured as a factor of its reliability - as reliability increases, so does availability. Availability of a system may also be increased by the strategy on focusing on increasing testability & maintainability and not on reliability. Improving maintainability is generally easier than reliability. Maintainability estimates (Repair rates) are also generally more accurate. However, because the uncertainties in the reliability estimates are in most cases very large, it is likely to dominate the availability (prediction uncertainty) problem, even while maintainability levels are very high.
- **Maintainability** - is defined as the probability of performing a successful repair action within a given time. In other words, maintainability measures the ease and speed with which a system can be restored to operational status after a failure occurs.
- **Portability** - the ability of a computer program to be ported from one system to another in computer science

## System Design 4

---

### 4.1 DATA FLOW DIAGRAMS

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

It is common practice to draw the context-level data flow diagram first, which shows the interaction between the system and external agents which act as data sources and data sinks. This helps to create an accurate drawing in the context diagram. The system's interactions with the outside world are modelled purely in terms of data flows across the system boundary. The context diagram shows the entire system as a single process, and gives no clues as to its internal organization.

This context-level DFD is next "exploded", to produce a Level 1 DFD that shows some of the detail of the system being modeled. The Level 1 DFD shows how the system is divided into sub-systems (processes), each of which deals with one or more of the data flows to or from an external agent, and which together provide all of the functionality of the system as a whole. It also identifies internal data stores that must be present in order for the system to do its job, and shows the flow of data between the various parts of the system.

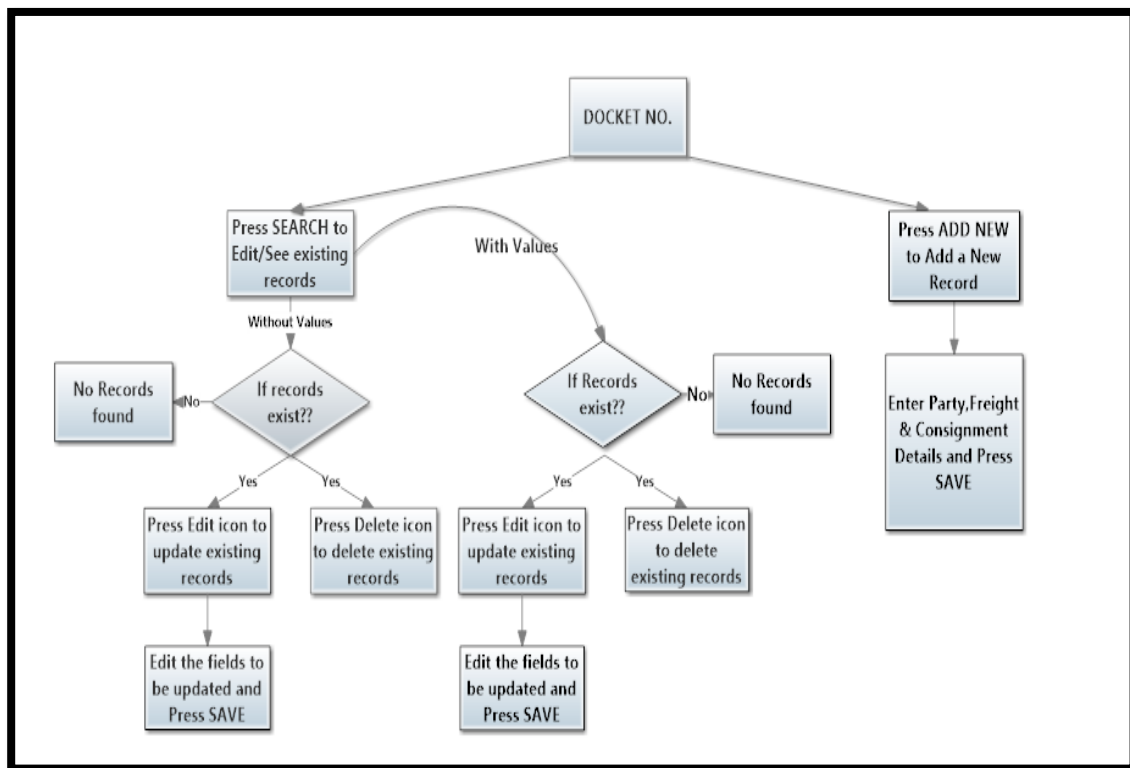
Data flow diagrams are one of the three essential perspectives of the structured-systems analysis and design method SSADM. The sponsor of a project and the end users will need to be briefed and consulted throughout all stages of a system's evolution. With a data flow diagram, users are able to visualize how the system will operate, what the system will accomplish, and how the system will be implemented. The old system's

dataflow diagrams can be drawn up and compared with the new system's data flow diagrams to draw comparisons to implement a more efficient system. Data flow diagrams can be used to provide the end user with a physical idea of where the data they input ultimately has an effect upon the structure of the whole system from order to dispatch to report. How any system is developed can be determined through a data flow diagram model.

In the course of developing a set of levelled data flow diagrams the analyst/designers is forced to address how the system may be decomposed into component sub-systems, and to identify the transaction data in the data model.

Data flow diagrams can be used in both Analysis and Design phase of the SDLC.

There are different notations to draw data flow diagrams (Yourdon & Coad and



Gane&Sarson), defining different visual representations for processes, data stores, data flow, and external entities.

Fig. 4.1 :- Docket Number

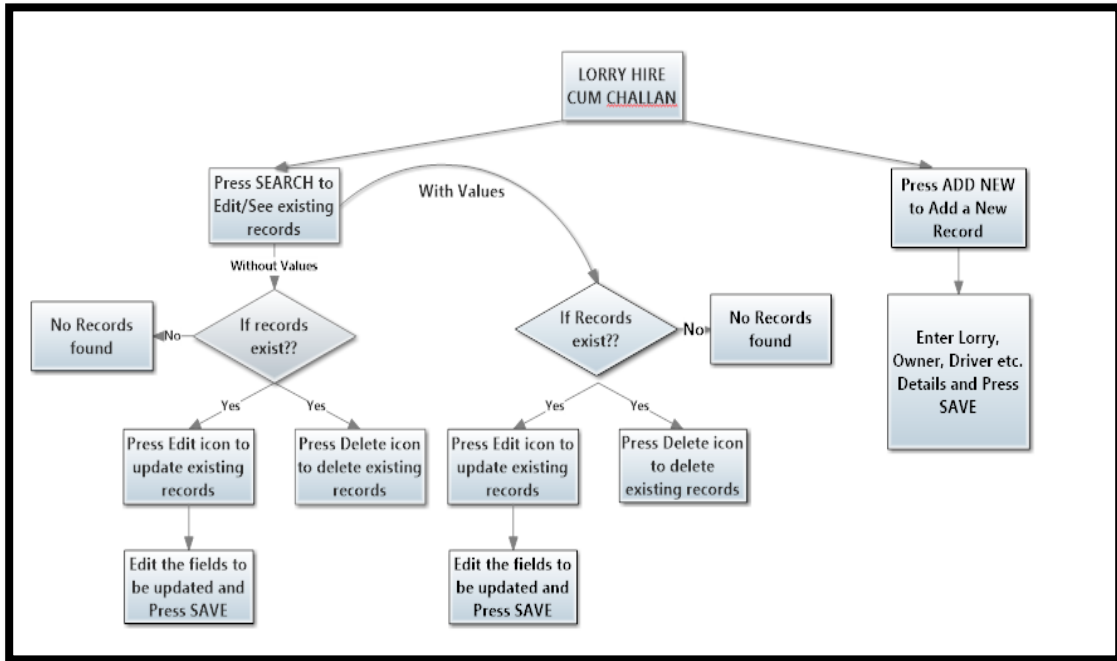
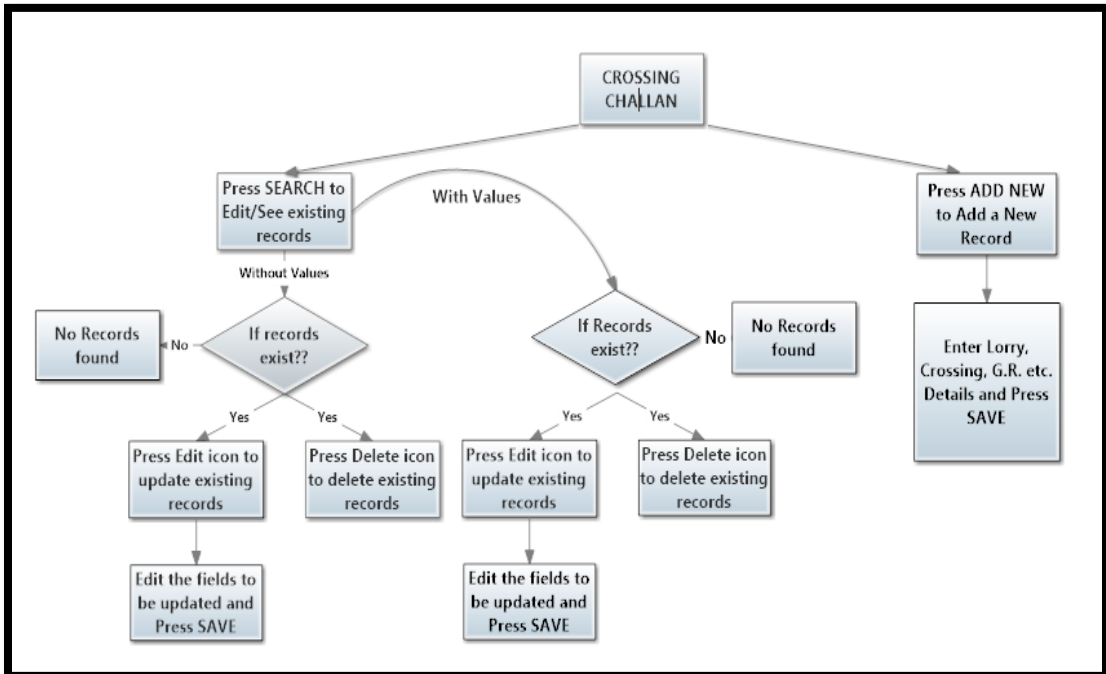


Fig. 4.2 :- Lorry Hire Cum Challan



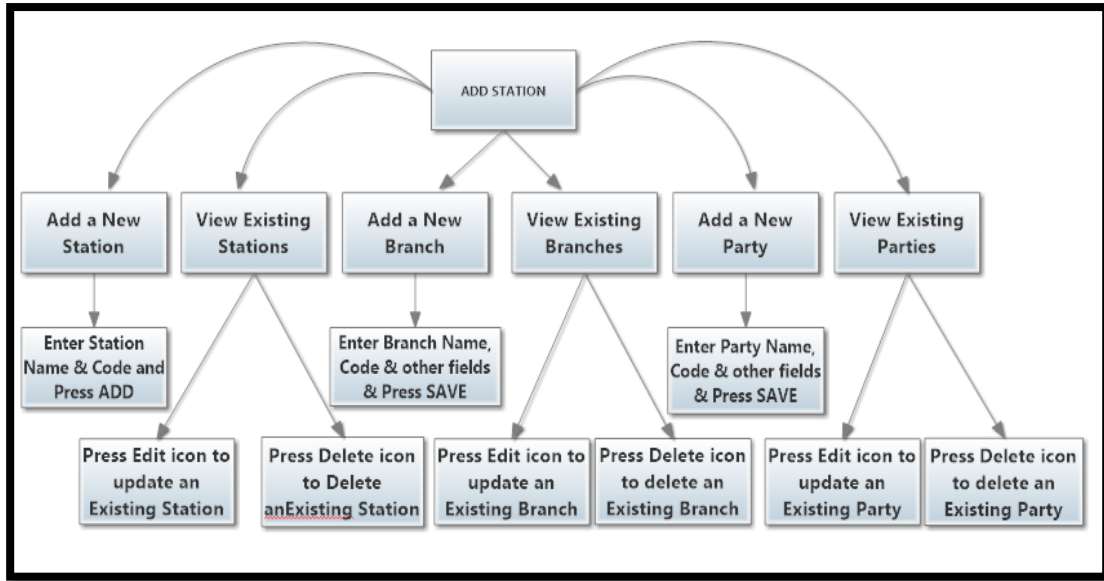


Fig. 4.3 :- Crossing Challan

Fig. 4.4 :- Add Station

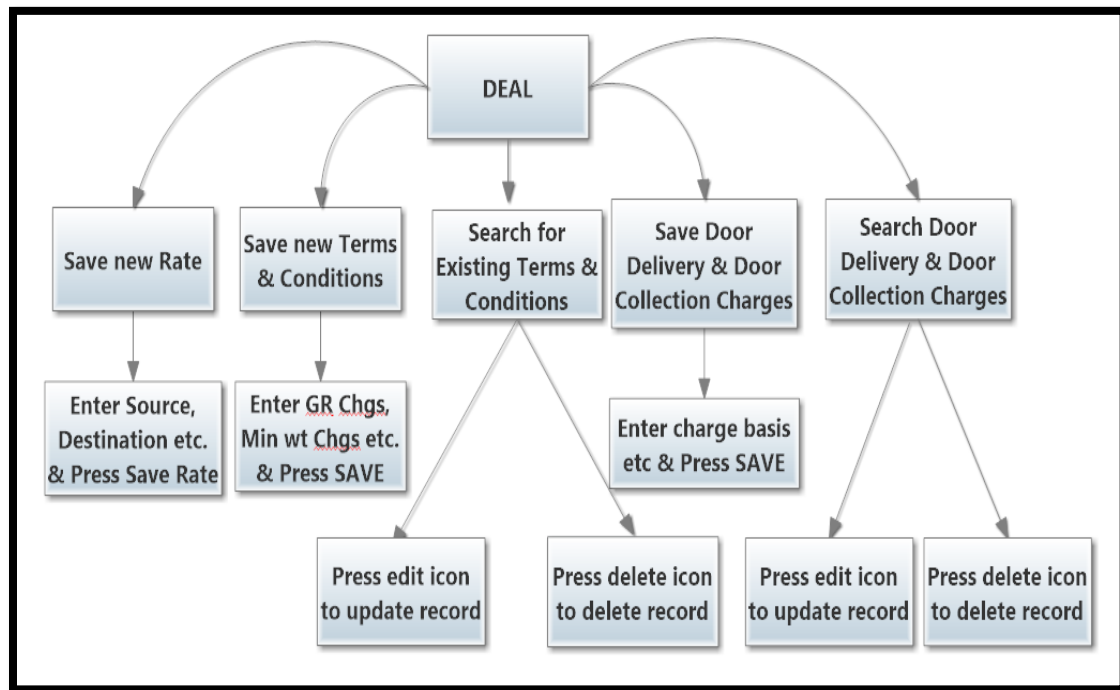


Fig. 4.5 :- Deal

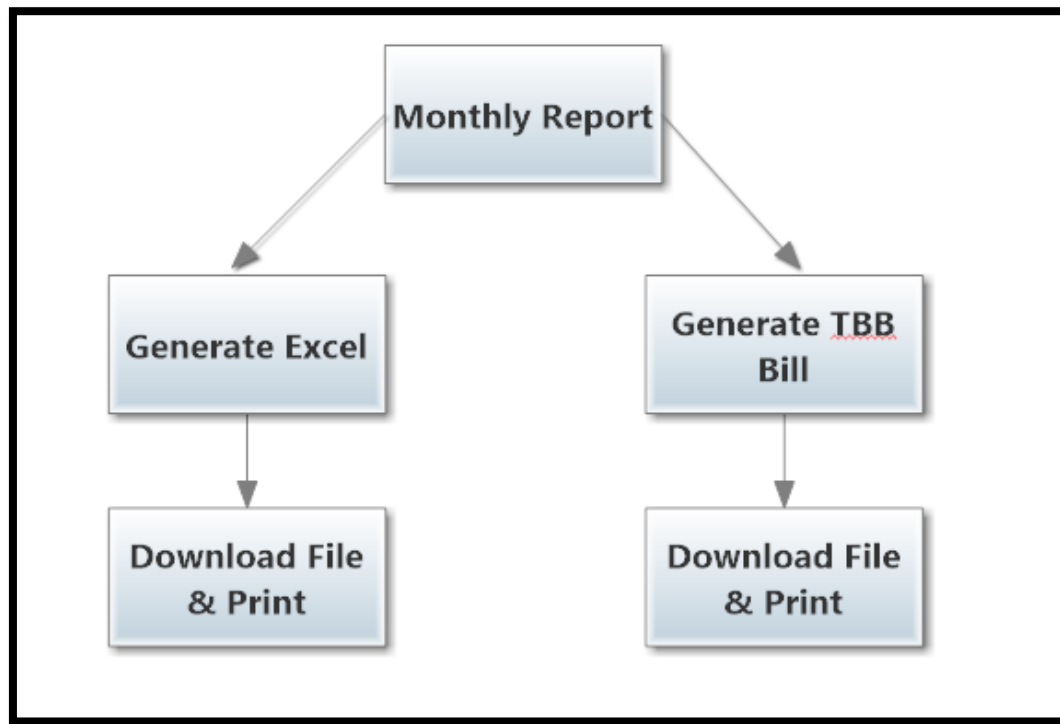


Fig. 4.6 :- Monthly Report

## 4.2 DATABASE DESIGN

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model these are the tables and view. In an object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the database management system (DBMS).

The process of doing database design generally consists of a number of steps which will be carried out by the database designer. Usually, the designer must:

- Determine the relationships between the different data elements.
- Superimpose a logical structure upon the data on the basis of these relationships.

## Data Base Tables

Table No 4.1: Bill Table

S.no.	Column Name	Data type(size)	Allow nulls	Constraints
1	PKID_Bill	Int	NOT NULL	Primary Key, Identity
2	Bill number	Int	NULL	
	Billing date	Datetime	NULL	
4	Name	Varchar(max)	NULL	
5	Gr no.	Int	NULL	
6	Station	Varchar(max)	NULL	
7	Weight	Int	NULL	
8	Freightamount	Int	NULL	
9	LabourLoadingUnloading	Int	NULL	
10	ServiceTax	Decimal(18,0)	NULL	
11	TotalAmount	Int	NULL	
12	BookingTime	Datetime	NULL	

Table No 4.2 Branch table

S no.	Column Name	Data type(size)	Allow nulls	Constraints
1	PKID_Branch	Int	NOT NULL	Primary Key, Identity
2	BranchCode	Varchar(100)	NULL	
3	BranchName	Varchar(100)	NULL	
4	BranchType	Varchar(50)	NULL	
5	Address	Varchar(250)	NULL	
6	ContactNo.	Char(10)	NULL	



Table No 4.3 Consignee details table

S no.	Column Name	Data type(size)	Allow nulls	Constraints
1	PKID_ConsigneeDetail	Int	NOT NULL	Primary Key, Identity
2	Address	Nvarchar(250)	NULL	
3	Name	Varchar(150)	NULL	
4	CstTinNo.	Varchar(50)	NULL	
5	PhoneNo.	Char(11)	NULL	
		Nchar(10)	NOT NULL	

Table No 4.4 Consigner Details Table

S no.	Column Name	Data type(size)	Allow nulls	Constraints
1	PKID_ConsignerDetails	Int	NOT NULL	Primary Key, Identity
2	Address	Nvarchar(250)	NULL	
3	Name	Varchar(150)	NULL	
4	CstTinNo.	Varchar(50)	NULL	
5	PhoneNo.	Char(11)	NULL	

Table No 4.5 Crossing challan Table

S no.	Column Name	Data type(size)	Allow nulls	Constraints
1	PKID_CrossingChallan	Int	NOT NULL	Primary Key, Identity
2	CrossingChallanNo.	Int	NULL	
3	BranchOffice	Varchar(max)	NULL	
4	VehicleNo	Varchar(max)	NULL	
5	ChallanDate	Datetime	NULL	
6	Name	Varchar(max)	NULL	
7	GRNo.	Nvarchar(250)	NULL	
8	BookingDate	Datetime	NULL	

9	Destination	Varchar(max)	NULL	
10	[Content]	Varchar(max)	NULL	
11	Pkgs	Int	NULL	
12	Weight	Int	NULL	
13	CrossFreight	Int	NULL	
14	Crossto	Varchar(max)	NULL	
15	BookingForm	Varchar(max)	NULL	
16	BalanceFreight	Decimal(18,2)	NULL	
17	TotalFreight	Int	NULL	
18	PhoneNo.	Varchar(10)	NULL	
19	PaymentRemarks	Varchar(250)	NULL	
20	PaymentAmount	Int	NULL	
21	PaymentDate	Datetime	NULL	
22	PaymentStatus	Varchar(50)	NULL	
23	CrossingPaymentStatus	Varchar(50)	NULL	
24	TotalToPay	Decimal(18,2)	NULL	
25	TotalWeight	Int	NULL	
26	TotalPkgs	Int	NULL	

Table No 4.6 DDCC table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_DDCC	Int	NOT NULL	Primary Key, Identity
2	PartyCode	Varchar(50)	NULL	
3	ChargeBasis	Varchar(50)	NULL	
4	Range	Int	NULL	
5	[to]	Int	NULL	
6	Charge	Int	NULL	
7	Sameasdd	Char(1)	NULL	
8	ChargeBasisDCC	Varchar(50)	NULL	
9	RangeDCC	Int	NULL	

10	ToDCC	Int	NULL	
11	ChargeDCC	Nchar(10)	NULL	

Table No 4.7 Deal table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_PartyDetails	Int	NOT NULL	Primary Key, Identity
2	PartyName	Varchar(150)	NULL	
3	PartyCode	Varchar(50)	NULL	
4	RateType	Varchar(50)	NULL	
5	Source	Varchar(50)	NULL	
6	Destination	Varchar(50)	NULL	
7	Rate	Int	NULL	
8	TransitTime	Int	NULL	

Table No 4.8 Docket no. table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_DocketNo.	Int	NOT NULL	Primary Key, Identity
2	DocketNo.	Int	NOT NULL	
3	ConsignerName	Varchar(max)	NULL	
4	ConsignerAddress	Varchar(max)	NULL	
5	ConsignerContactNo.	Varchar(10)	NULL	
6	ConsignerCST_TIN_No.	Varchar(50)	NULL	
7	ConsigneeName	Varchar(max)	NULL	
8	ConsigneeAddress	Varchar(max)	NULL	
9	ConsigneeContactNo.	Varchar(10)	NULL	
10	ConsigneeCST_TIN_no.	Varchar(50)	NULL	
11	ConsignmentValue	Decimal(18,2)	NULL	
12	FreightInWords	Varchar(max)	NULL	
13	PkgWt	Int	NULL	

14	ActWt	Int	NULL	
15	ChWt	Int	NULL	
16	BookingDate	Datetime	NULL	
17	BookingFrom	Varchar(50)	NULL	
18	Destination	Varchar(50)	NULL	
19	RoadPermitNo	Varchar(150)	NULL	
20	Description	Varchar(550)	NULL	
21	SpeacialNote	Varchar(550)	NULL	
22	AFrt	Int	NULL	
23	AOC	Int	NULL	
24	CRisk	Int	NULL	
25	Hamali	Int	NULL	
26	FOV	Int	NULL	
27	Others	Int	NULL	
28	StaticalCharges	Int	NULL	
29	Total	Int	NULL	
30	STax	Decimal(18,2)	NULL	
31	GTotal	Decimal(18,2)	NULL	
32	InvoiceNo.	Varchar(50)	NULL	
33	PaymentMode	Char(10)	NULL	
34	Branch	Varchar(50)	NULL	
35	PackingMethod	Varchar(50)	NULL	
36	Status	Varchar(50)	NULL	
37	PaymentRemark	Varchar(150)	NULL	
38	DeliveryDate	Datetime	NULL	
39	PartyCode	Varchar(50)	NULL	

Table No 4.9 Lorry hire cum challan Details table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_LHCDetails	Int	NOT NULL	Primary Key, Identity
2	ChallanNo.	Int	NOT NULL	
3	Freight	Int	NULL	
4	Hamali	Int	NULL	
5	Extrafor	Int	NULL	
6	Total	Int	NULL	
7	Lessadvance	Int	NULL	
8	Balance	Int	NULL	
9	ChqNo.	Nvarchar(50)	NULL	
10	ChqName	Nvarchar(150)	NULL	
11	Paidat	Nvarchar(50)	NULL	
12	Chokedby	Nvarchar(50)	NULL	
13	Loadedby	Nvarchar(50)	NULL	

Table No 4.10 lorry hire cum challan table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_LorryHireCumChallan	int	NOT NULL	Primary Key, Identity
2	ChassisNo.	Varchar(max)	NULL	
3	EngineNo.	Varchar(max)	NULL	
4	OwnerName	Varchar(max)	NULL	
5	OwnerAddress	Varchar(max)	NULL	
6	ContactNo.	Varchar(10)	NULL	
7	PAN	Varchar(50)	NULL	
8	DriverName	Varchar(max)	NULL	
9	LicNo.	Varchar(max)	NULL	
10	DriverAddress	Varchar(max)	NULL	

11	BrokerName	Varchar(max)	NULL	
12	ChallanDate	Datetime	NULL	
13	Branch	Varchar(max)	NOT NULL	
14	ChallanNo.	Int	NOT NULL	
15	VehicleNo.	Varchar(50)	NULL	
16	BookingFrom	Varchar(50)	NULL	
17	Destination	Varchar(50)	NULL	
18	GRNo.	Nvarchar(250)	NULL	
19	Pkgs	Int	NULL	
20	PackingMethod	Varchar(50)	NULL	Default
21	Contents	Varchar(max)	NULL	
22	Weight	Int	NULL	
23	PaymentMode	Varchar(50)	NULL	
24	Freight	Int	NULL	
25	Hamali	Int	NULL	
26	Extrafor	Int	NULL	
27	Total	Int	NULL	
28	Lessadvance	Int	NULL	
29	Balance	Int	NULL	
30	PaidAt	Varchar(50)	NULL	
31	ChqNo.	Varchar(150)	NULL	
32	ChqName	Varchar(150)	NULL	
33	VehiclesCheckedBy	Varchar(150)	NULL	
34	GoodsLoadedBy	Varchar(150)	NULL	
35	DriverContactNo.	Varchar(10)	NULL	
36	BrokerContactNo.	Varchar(10)	NULL	
37	Amountpaid	Int	NULL	
38	ReceiptNo.	Int	NULL	
39	PaymentStatus	Varchar(50)	NULL	
40	PaymentRemark	Nvarchar(250)	NULL	

41	TotalPkgs	Int	NULL	
42	TotalWeight	Int	NULL	
43	TotaltoPay	Decimal(18,2)	NULL	

Table No 4.11 MRC table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_MRC	Int	NOT NULL	Primary Key, Identity
2	MRCNo.	Int	NULL	
3	Name	Varchar(max)	NULL	
4	CreatedDate	Datetime	NULL	
5	PaymentMode	Char(10)	NULL	
6	BookingStation	Varchar(50)	NULL	
7	Weight	Int	NULL	
8	GRNo.	Int	NULL	
9	Pkgs	Int	NULL	
10	Description	Varchar(max)	NULL	
11	Freight	Int	NULL	
12	Hamali	Int	NULL	
13	DeliveryCharge	Int	NULL	
14	StorageCharge	Int	NULL	
15	StaticalCharge	Int	NULL	
16	Total	Int	NULL	
17	ServiceTax	Decimal(18,2)	NULL	
18	GTotal	Decimal(18,2)	NULL	
19	ArrivalDate	Datetime	NULL	
20	GTotalinWords	Varchar(max)	NULL	

Table No 4.12 Party table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_Party	Int	NOT NULL	Primary Key, Identity
2	PartyName	Varchar(max)	NULL	
3	PartyAddress	Varchar(max)	NULL	
4	PartyState	Varchar(max)	NULL	
5	PartyDistrict	Varchar(max)	NULL	
6	PinCode	Int	NULL	
7	Emailid	Varchar(max)	NULL	
8	PartyCST_TIN_No.	Varchar(50)	NULL	
9	PartyContactNo.	Char(11)	NULL	
10	PartyCode	Varchar(50)	NULL	

Table No 4.13 Registration table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_Register	Int	NOT NULL	Primary Key, Identity
2	Name	Varchar(150)	NULL	
3	EmailId	Varchar(150)	NULL	
4	Password	Varchar(50)	NULL	
5	ContactNo.	Varchar(10)	NULL	

Table No 4.14 State Table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_State	Int	NOT NULL	Primary Key, Identity
2	Code	Varchar(100)	NULL	
3	StateName	Varchar(100)	NULL	



Table No 4.15 Station table

S no.	Column Name	Data type(size)	Allow null	Constraints
1	PKID_Station	Int	NOT NULL	Primary Key, Identity
2	StationName	Varchar(100)	NULL	
3	StationCode	Varchar(100)	NULL	

Table No 4.16 TNC Table

S no.	ColumnName	Data type(size)	Allow null	Constraints
1	PKID_TNC	Int	NOT NULL	Primary Key, Identity
2	PartyCode	Varchar(50)	NULL	
3	GRCharges	Varchar(50)	NULL	
4	MinChargeWt	Int	NULL	
5	RecieptCharge	Int	NULL	
6	LabourCharges	Int	NULL	
7	Basisson	Varchar(50)	NULL	

### 4.2.1 Design Process

- Determine the purpose of the database - This helps prepare for the remaining steps.
- Find and organize the information required - Gather all of the types of information to record in the database, such as product name and order number.
- Divide the information into tables - Divide information items into major entities or subjects, such as Products or Orders. Each subject then becomes a table.
- Turn information items into columns - Decide what information needs to be stored in each table. Each item becomes a field, and is displayed as a column in the table. For example, an Employees table might include fields such as Last Name and Hire Date.
- Specify primary keys - Choose each table's primary key. The primary key is a column, or a set of columns, that is used to uniquely identify each row. An example might be Product ID or Order ID.

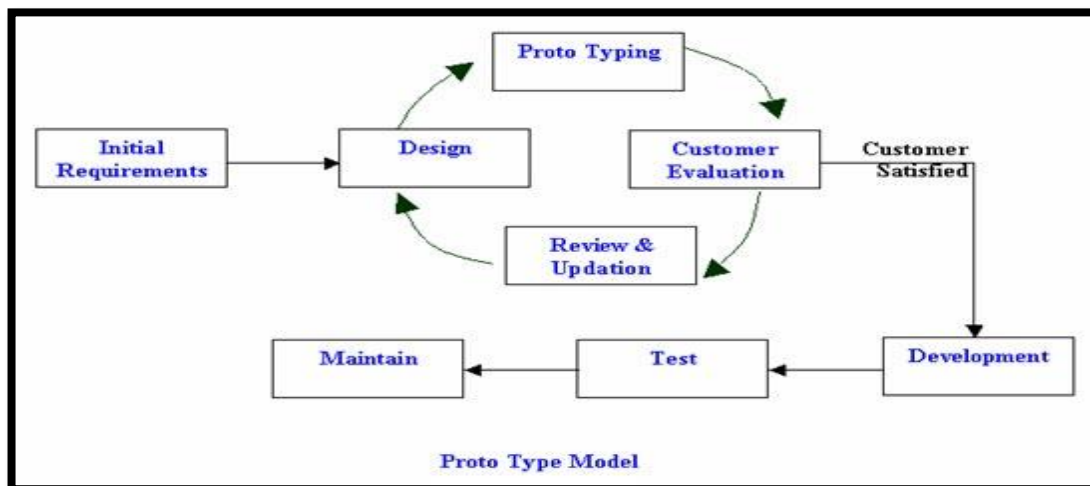
- Set up the table relationships - Look at each table and decide how the data in one table is related to the data in other tables. Add fields to tables or create new tables to clarify the relationships, as necessary.
- Refine the design - Analyze the design for errors. Create tables and add a few records of sample data. Check if results come from the tables as expected. Make adjustments to the design, as needed.
- Apply the normalization rules - Apply the data normalization rules to see if tables are structured correctly. Make adjustments to the tables

## Methodology Used<sup>5</sup>

### 5.1 WATERFALL METHODOLOGY & ITS DISADVANTAGES

To develop the software the basic thing is that what is the model will be used. The model will be used is of great importance for development of the software.

In the waterfall model, the sequence of the activities performed in a software development project: requirement analysis, project planning, system design, detailed design, coding, and unit testing, system integration and testing. This is the order the different phases will be apply for develop of the project. Linear ordering of the phases is the important for the development of the software because by using this model you can follow the phases in the linear order. First the requirement analysis is done then you can proceed the next phase like designing. This model is used when you know all the phases in advance i.e. you know the requirement of the client. Because requirement analysis is important for the development of the software. For the development of the software there are two parties are involved in the development of the software – a client and a developer. Most often the requirement of the client does not know and the requirement of the client will be changed. So before using this model requirement of the client is known



in advanced.

Fig. 5.1 Prototype Model

To overcome these problems we can use the prototype model. The basic idea for using this model is that the requirements are not known initially. In this model a prototype is built to understand the requirements. This prototype is developed based on the currently known requirements. The prototype means to develop the formal model of the software which meets the current requirements. By using this prototype, the client can get an actual feeling of the system because the interaction with the prototype can enable the client to better understand the requirements of the desired system. By using this model there is no work manually. By using this model we can create a prototype, which meets the primary requirements of the TRANSPORTATION MANAGEMENT SYSTEM. In the waterfall model the requirements had been known in advance. To solve this problem we can use the prototype model.

## **5.2 PROTOTYPE METHODOLOGY**

The development of the prototype typically starts when the preliminary version of the requirements specification document has been developed. After developing the prototype, the end users and clients are given an opportunity to use the prototype. Based on the experience, they provide the feedback to the developers regarding the prototype: what is correct, what needs to be modified, what is missing, what is not needed, etc. In this software when we give the prototype to the client & they give the feedback like the needed of the login in the software.

Software prototyping, is the development approach of activities during software development, the creation of prototypes, i.e., incomplete versions of the software program being developed.

The basic principles are:

- Not a standalone, complete development methodology, but rather an approach to handle selected parts of a larger, more traditional development methodology (i.e. incremental, spiral, or rapid application development (RAD)).
- Attempts to reduce inherent project risk by breaking a project into smaller segments and providing more ease-of-change during the development process.
- User is involved throughout the development process, which increases the likelihood of user acceptance of the final implementation.

- Small-scale mock-ups of the system are developed following an iterative modification process until the prototype evolves to meet the users' requirements.
- While most prototypes are developed with the expectation that they will be discarded, it is possible in some cases to evolve from prototype to working system.

## Implementation 6

---

### 6.1 FEATURES OF LANGUAGE C#

C# (pronounced as see sharp) is a multi-paradigm programming language encompassing strong typing,imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines. It was developed by Microsoft within its .NET initiative and later approved as a standard by Ecma (ECMA-334) and ISO (ISO/IEC 23270:2006). C# is one of the programming languages designed for the Common Language Infrastructure.

C# is intended to be a simple, modern, general-purpose, object-oriented programming language. Its development team is led by Anders Hejlsberg. The most recent version is C# 5.0, which was released on August 15, 2012.

### 6.2 MAIN FEATURES OF C#

#### 6.2.1 Simple

- Pointers are missing in C#.
- Unsafe operations such as direct memory manipulation are not allowed.
- In C# there is no usage of "::" or "->" operators.
- Since it's on .NET, it inherits the features of automatic memory management and garbage collection.
- Varying ranges of the primitive types like Integer,Floats etc.
- Integer values of 0 and 1 are no longer accepted as boolean values.Boolean values are pure true or false values in C# so no more errors of "="operator and "=="operator.
- "==" is used for comparison operation and "=" is used for assignment operation.

### 6.2.2. Modern

- C# has been based according to the current trend and is very powerful and simple for building interoperable, scalable, robust applications.
- C# includes built in support to turn any component into a web service that can be invoked over the internet from any application running on any platform.

### 6.2.3. Object Oriented

- C# supports Data Encapsulation, inheritance, polymorphism, interfaces.
- (int, float, double) are not objects in java but C# has introduced structures (structs) which enable the primitive types to become objects.
- `inti=1;`
- `string a=i.ToString(); //conversion (or) Boxing`

### 6.2.4. Type Safe

- In C# we cannot perform unsafe casts like convert double to a boolean.
- Value types (primitive types) are initialized to zeros and reference types (objects and classes) are initialized to null by the compiler automatically.
- arrays are zero base indexed and are bound checked.
- Overflow of types can be checked.

### 6.2.5. Interoperability

- C# includes native support for the COM and windows based applications.
- Allowing restricted use of native pointers.
- Users no longer have to explicitly implement the unknown and other COM interfaces, those features are built in.
- C# allows the users to use pointers as unsafe code blocks to manipulate your old code.
- Components from VB NET and other managed code languages and directly be used in C#.

### 6.2.6. Scalable and Updatable

- .NET has introduced assemblies which are self describing by means of their manifest. manifest establishes the assembly identity, version, culture and digital signature etc. Assemblies need not to be register anywhere.
- To scale our application we delete the old files and updating them with new ones. No registering of dynamic linking library.

## 6.3 SAMPLE CODE

### 6.3.1 Login Page Code

#### Login.aspx.cs (Code Behind File)

```
public partial class Login : System.Web.UI.Page
{
    Signupobj = new Signup();
    DataTable result;

    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void btnLogin_Click(object sender, EventArgs e)
    {
        result=obj.AuthenticateUser(txtUserid.Text, txtPassword.Text);
        if (result.Rows.Count>0)
        {
            Session["UserName"] = result.Rows[0]["Name"].ToString();
            Response.Redirect("DocketNo.aspx");

        }
        else
        {
            txtUserid.Text = "";
            txtPassword.Text = "";
            txtUserid.Focus();
        }
    }

    protected void btnsignup_Click(object sender, EventArgs e)
    {
    }
}
```



```

if (txtRegPassword.Text == txtConfirmRegPassword.Text)
    {
obj.InsertRegistration(txtfullname.Text,      txtemail.Text,      txtRegPassword.Text,
txtcontactno.Text);
    }
else
    {
txtConfirmRegPassword.Text = "";
txtRegPassword.Text = "";
    }
}
}

```

### 6.3.2 Docket Number Page Code

#### DocketNo.aspx Code (Source Code)

```

<script>
    $(function () {
        $("#ContentPlaceHolder1_txtdate").datepicker({
onSelect: function () {
currenttab = $(this).attr("tabindex");
nexttab = currenttab * 1 + 1;
                $("#ContentPlaceHolder1_txtconsignername").each(function () {
if ($(this).attr("tabindex") == nexttab)
                    $(this).focus();
                });
            });
        $("#ContentPlaceHolder1_txtdeleverydate").datepicker({
onSelect: function () {
currenttab = $(this).attr("tabindex");
nexttab = currenttab * 1 + 1;
                $("#ContentPlaceHolder1_ddlstatus").each(function () {
if ($(this).attr("tabindex") == nexttab)
                    $(this).focus();
                });
            });
        $("#ContentPlaceHolder1_txtreceiptdate").datepicker({
onSelect: function () {
currenttab = $(this).attr("tabindex");
nexttab = currenttab * 1 + 1;
                $("#ContentPlaceHolder1_ddlpaymentmode").each(function () {
if ($(this).attr("tabindex") == nexttab)

```

```

function SearchConsigner() {
    $("#ContentPlaceHolder1_txtconsignername").autocomplete({
    source: function (request, response) {
        $.ajax({
        type: "POST",
        contentType: "application/json; charset=utf-8",
        url: "DocketNo.aspx/GetAutoCompleteConsigner",
        data:                "{ 'username': " +
        document.getElementById('ContentPlaceHolder1_txtconsignername').value + " }",
        dataType: "json",
        success: function (data) {
        response(data.d);
        },
        error: function (result) {
        alert("Error");
        }
        });
        }
    });
}

```

### 6.3.3 LorryHireCumChallan Page Code

#### LorryHireCumChallan.aspx.cs Code (Code Behind File)

```

public partial class LorryHireCumChallan : System.Web.UI.Page
{
    VehicleHireCumChallan objvhcchallan = new VehicleHireCumChallan();
    CommonMethods objcomn = new CommonMethods();
    DataTable result;

    protected void Page_Load(object sender, EventArgs e)
    {
        EnableControls(false);
        if (!IsPostBack)
        {
            BindDropdown();
        }
    }

    public string Bookingfrom()
    {
        return ddlbookingfrom.SelectedItem.Text.Trim();
    }
}

```

```

[WebMethod]
publicstatic List<string> GetAutoCompleteGrno(string grno)
{
    List<string> result = newList<string>();

    using (SqlConnection con =
newSqlConnection(System.Configuration.ConfigurationManager.AppSettings["Connecti
onString"]))
    {
        if (con.State != ConnectionState.Open)
        {
            con.Open();
        }
        using (SqlCommand cmd = newSqlCommand("select distinct top 10 DocketNo from
tbl_DocketNo where DocketNo LIKE +@SearchText+'% ' ", con))
        {
            cmd.Parameters.AddWithValue("@SearchText", grno);
            SqlDataReader dr = cmd.ExecuteReader();
            while (dr.Read())
            {
                result.Add(dr["DocketNo"].ToString());
            }
        }
        return result;
    }
}

```

### 6.3.4 CrossingChallan Page Code

#### CrossingChallan.aspx.cs (Code Behind File)

```

[WebMethod]
publicstatic List<string> GetAutoCompleteGrno(string grno)
{
    List<string> result = newList<string>();

    using (SqlConnection con =
newSqlConnection(System.Configuration.ConfigurationManager.AppSettings["Connecti
onString"]))
    {
        if (con.State != ConnectionState.Open)
        {
            con.Open();
        }
        using (SqlCommand cmd = newSqlCommand("select distinct top 10 DocketNo from
tbl_DocketNo where DocketNo LIKE +@SearchText+'% ' ", con))
        {

            cmd.Parameters.AddWithValue("@SearchText", grno);

```

```

SqlDataReader dr = cmd.ExecuteReader();
while (dr.Read())
    {
result.Add(dr["DocketNo"].ToString());
    }
return result;
    }
}
[WebMethod]
public static List<string> GetAutoCompleteGrnoSearch(string grno)
{
List<string> result = newList<string>();

using (SqlConnection con =
new SqlConnection(System.Configuration.ConfigurationManager.AppSettings["Connecti
onString"]))
    {
if (con.State != ConnectionState.Open)
    {
con.Open();
    }
using (SqlCommand cmd = new SqlCommand("select distinct top 10 GRNo from
tbl_CrossingChallan where GRNo LIKE +@SearchText+'%'", con))
    {

cmd.Parameters.AddWithValue("@SearchText", grno);
SqlDataReader dr = cmd.ExecuteReader();
while (dr.Read())
    {
result.Add(dr["GRNo"].ToString());
    }
return result;
    }
}
}
}

```

### 6.3.5 Deal Page Code

#### Deal.aspx.cs (Code Behind File)

```

public partial class Deal : System.Web.UI.Page
{
DealnDdccoobj = new DealnDdcc();
Partyobjp = new Party();
CommonMethodsobjcomn = new CommonMethods();
}
}

```

```

DataTable result;
protectedvoidPage_Load(object sender, EventArgs e)
{

if (!IsPostBack)
{
BindDropDown();
}
}

[WebMethod]
publicstaticList<String>GetAutoCompletePartyCode(StringPartyCode)
{
List<String> result = newList<String>();

using (SqlConnection con =
newSqlConnection(System.Configuration.ConfigurationManager.AppSettings["Connecti
onString"]))
{
if (con.State != ConnectionState.Open)
{
con.Open();
}
using (SqlCommandcmd = newSqlCommand("select top 10 PartyCode from tbl_Party
where PartyCode LIKE +@SearchText+'% ' ", con))
{
cmd.Parameters.AddWithValue("@SearchText", PartyCode);
SQLDataReaderdr = cmd.ExecuteReader();
while (dr.Read())
{
result.Add(dr["PartyCode"].ToString());
}
}
return result;
}
}

publicvoidBindDropDown()
{
result = objcomn.BindDropDown();
ddlsource.DataSource = result;
ddlsource.DataTextField = "StationName";
ddlsource.DataValueField = "StationName";
ddlsource.DataBind();

ddldestination.DataSource = result;

```

```

ddldestination.DataTextField = "StationName";
ddldestination.DataValueField = "StationName";
ddldestination.DataBind();

ddlsource.Items.Insert(0, newListItem("--Select--", "-1"));
ddldestination.Items.Insert(0, newListItem("--Select--", "-1"));

    }
publicvoid BindGrid()
    {
result = obj.BindGridParty();
GridView_PartynTNC.DataSource = result;
GridView_PartynTNC.DataBind();
// tblGrid.Visible = true;
    }
publicvoid BindGridDCC()
    {
GridView_DDCC.EditIndex = -1;
result = obj.BindGridDDCC();
GridView_DDCC.DataSource = result;
GridView_DDCC.DataBind();

// tblGrid.Visible = true;
    }
#region Ger Serial No
publicint getSerialNo()
    {
int x = 0;
x = GridView_PartynTNC.PageIndex * GridView_PartynTNC.PageSize;

foreach (GridViewRow row in GridView_PartynTNC.Rows)
    {
x = x + 1;
    }
return x + 1;
    }
#endregion
protectedvoid btnsavepartydetails_Click(object sender, EventArgs e)
    {
if (btnsavepartydetails.Text == "SAVE")
    {
obj.InsertParty(txtpartyname.Text.Trim(), txtpartycode.Text.Trim(),
ddlratetypeparty.SelectedItem.Text.Trim(), ddlsource.SelectedItem.Text.Trim(),
ddldestination.SelectedItem.Text.Trim(), txtrate.Text.Trim(), txttransittime.Text.Trim());
BindGrid();
    }
    }

```

```
else
{
obj.UpdateParty(txtpartyname.Text.Trim(), txtpartycode.Text.Trim(),
ddlratetypeparty.SelectedItem.Text.Trim(), ddlsource.SelectedItem.Text.Trim(),
ddldestination.SelectedItem.Text.Trim(), txtrate.Text.Trim(), txttransittime.Text.Trim());
GridView_PartynTNC.EditIndex = -1;
BindGrid();
btnsavepartydetails.Text = "SAVE";
}

}

protected void btnsaveddcndccdetails_Click(object sender, EventArgs e)
{
if (btnsaveddcndccdetails.Text == "SAVE")
{
if (chkboxsameas.Checked)
{
obj.InsertDDCC(txtpartycode.Text.Trim(), ddlchargebasis.SelectedItem.Text.Trim(),
txtrange.Text.Trim(), txtto.Text.Trim(), txtcharge.Text.Trim(), "1",
ddlchargebasis.SelectedItem.Text.Trim(), txtrange.Text.Trim(), txtto.Text.Trim(),
txtcharge.Text.Trim());
BindGridDCC();
}
else
{
obj.InsertDDCC(txtpartycode.Text.Trim(), ddlchargebasis.SelectedItem.Text.Trim(),
txtrange.Text.Trim(), txtto.Text.Trim(), txtcharge.Text.Trim(), "0",
ddlchargebasis.SelectedItem.Text.Trim(), txtrange.Text.Trim(), txtto.Text.Trim(),
txtcharge.Text.Trim());
BindGridDCC();
}
}

else
{
if (chkboxsameas.Checked)
{
obj.UpdateDDCC(txtpartycode.Text.Trim(), ddlchargebasis.SelectedItem.Text.Trim(),
txtrange.Text.Trim(), txtto.Text.Trim(), txtcharge.Text.Trim(), "1",
ddlchargebasis.SelectedItem.Text.Trim(), txtrange.Text.Trim(), txtto.Text.Trim(),
txtcharge.Text.Trim());
GridView_DDCC.EditIndex = -1;
BindGridDCC();
btnsaveddcndccdetails.Text = "SAVE";
}
else
```

```

    {
obj.UpdateDDCC(txtpartycode.Text.Trim(),    ddlchargebasis.SelectedItem.Text.Trim(),
txtrange.Text.Trim(),    txtto.Text.Trim(),    txtcharge.Text.Trim(),    "0",
ddlchargebasis.SelectedItem.Text.Trim(),    txtrange.Text.Trim(),    txtto.Text.Trim(),
txtdcccharge.Text.Trim());
GridView_DDCC.EditIndex = -1;
BindGridDCC();
btnsaveddndccdetails.Text = "SAVE";
    }

}

}

```

### 6.3.6 Monthly Report Page

#### MonthlyReport.aspx (Source Code)

```

<% @PageTitle="" Language="C#" MasterPageFile="~/RoadCarriers.master" AutoEvent
Wireup="true" CodeFile="MonthlyReport.aspx.cs" Inherits="MonthlyReport"% >

```

```

<asp:ContentID="Content1" ContentPlaceHolderID="head" Runat="Server">

```

```

<styletype="text/css">

```

```

td {

```

```

}

```

```

body{font-size:12px!important;}

```

```

#ContentPlaceHolder1_txtfreightinwords {

```

```

height: 40px;

```

```

width: 276px;

```

```

}

```

```

#ContentPlaceHolder1_txtdescription, #ContentPlaceHolder1_txtspecialnote {

```

```

width: 94%;

```

```

}

```

```

#ContentPlaceHolder1_ddlbookingfrom, #ContentPlaceHolder1_ddlbookingfromsearch,

```

```

#ContentPlaceHolder1_ddldestinationsearch, #ContentPlaceHolder1_ddldestination {

```

```

height: 22px;

```

```

width: 142px;

```

```

}

```

```

#ContentPlaceHolder1_chkremarks {

```

```

height: 22px;

```

```

width: 242px;

```

```

}

```



```
#dvdDocketNo {
background-color: aliceblue;
width: 100%;
font-size: 12px;
}

.partyDetailBox {
width: 68%;
float: left;
border-radius: 5px0px0px5px;
box-shadow: inset0px0px3px#ccc;
margin-left: 10px;
border: 1pxsolid#ccc;
}

.FreightDetailBox {
width: 30%;
border-left: 0px!important;
float: left;
border: 1pxsolid#ccc;
border-radius: 0px5px5px0px;
box-shadow: inset0px0px3px#ccc;
}

.partyDetailBoxtable {
width: 100%;
}

.partyDetailBoxtrtd {
padding: 1px5px;
vertical-align: middle;
}

.FreightDetailBoxtrtdinput[type="text"] {
width: 87%!important;
}

.FreightDetailBoxtrtdtextarea {
width: 86.5%!important;
}

.partyDetailBoxtrtdtextarea {
width: 86.5%!important;
}
```

```

.partyDetailBoxtrtdselect {
width: 89%!important;
}

.FreightDetailBoxtrtdselect {
width: 88%!important;
}

.FreightDetailBoxtrtd {
padding: 1px5px;
vertical-align: middle;
}

.FreightDetailBoxtable {
width: 100%;
}

.HeaderHeading {
box-shadow: inset0px0px3px#ccc;
font-weight: bold;
font-size: 16px;
padding: 5px0px;
text-align: center;
border: 1pxsolid#ccc;
border-left: 0px!important;
border-right: 0px!important;
}

.DifferZFix {
border: 1pxsolid#ccc;
box-shadow: inset0px0px3px#ccc;
border-left: 0px!important;
border-right: 0px!important;
margin-bottom: 5px;
}
#ContentPlaceholder1_chkremarkssearch{ width:65%!important;}
#ContentPlaceholder1_txtbookingdatesearch{ width:70%!important;}
.BtnCenterinput[type="submit"]{ margin:0px10px; }
</style>
</asp:Content>
<asp:ContentID="Content2"ContentPlaceHolderID="ContentPlaceHolder1"Runat="Server">
<divclass="MonthlyReportShow">
<divstyle="margin: 10px; width: 98.1%; "class="FreightDetailBox">
<divstyle="width:60%; margin:auto">
<tablestyle="width: 100%;">

```

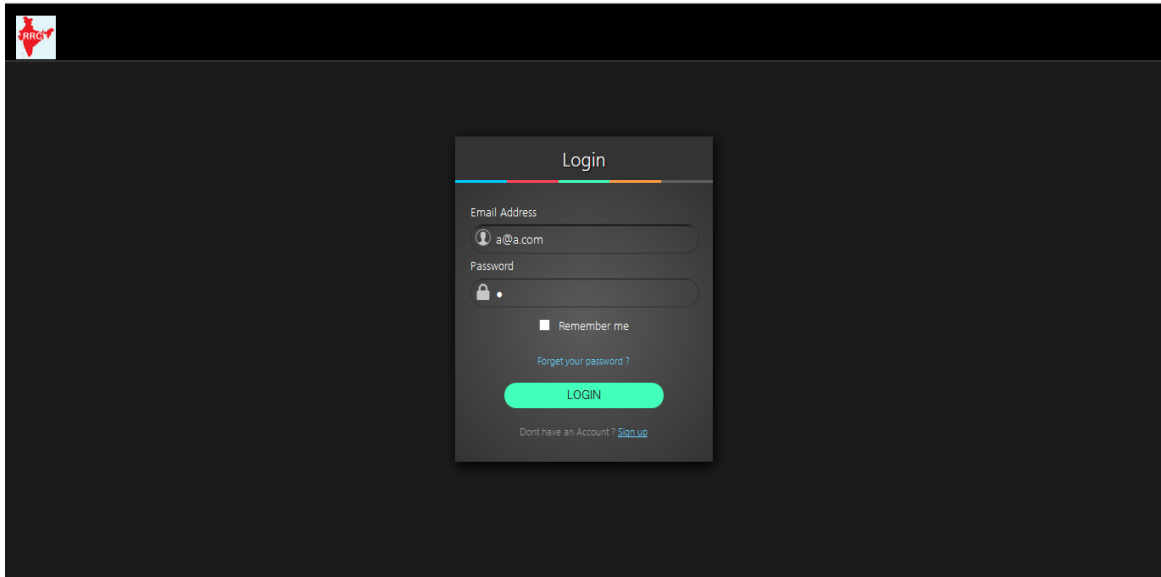
```

<tr>
<td>&nbsp;</td>
<td>&nbsp;</td>
<td>&nbsp;</td>
<td>&nbsp;</td>
</tr>
<tr>
<tdstyle="width:80px;">Party Code</td>
<td>
<asp:DropDownListID="ddlpartycode"runat="server"></asp:DropDownList>
</td>
<tdstyle="width:50px;">Month</td>
<td>
<asp:DropDownListID="ddlmonth"runat="server"TabIndex="1">
</asp:DropDownList>
</td>
<tdstyle="width:40px;">Year</td>
<td>
<asp:DropDownListID="ddlyear"runat="server"TabIndex="2">
</asp:DropDownList>
</td>
</tr>
<tr>
<td colspan="6" style="text-align:center">
<asp:ButtonID="btngenerateexcel"runat="server"OnClick="btngenerateexcel_Click"Tab
Index="3"Text="Generate Excel"/>
<asp:ButtonID="btnGenerateTBBBill"runat="server"OnClick="btnGenerateTBBBill_Cli
ck"TabIndex="3"Text="Generate TBB Bill"/>
</td>
</tr>
</table>
</div>
<div style="clear:both"></div>
</div>
</asp:Content>
<asp:ContentID="Content3"ContentPlaceHolderID="ContentPlaceHolder2"Runat="Serv
er">
</asp:Content>

```

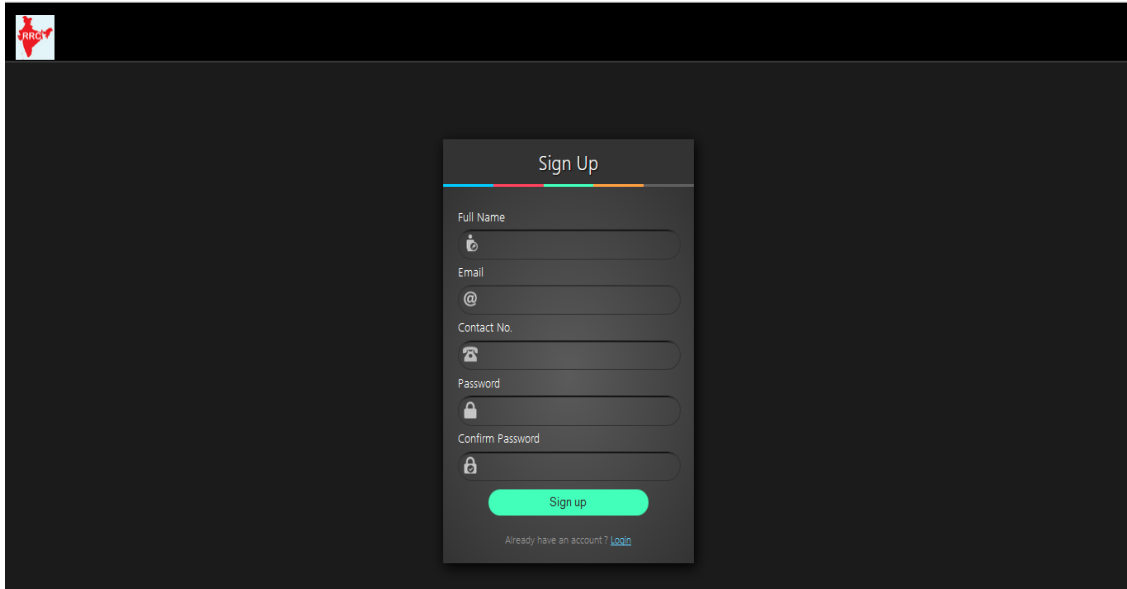
## 6.4 SCREENSHOTS

## Login Page

A screenshot of a login page with a dark background. In the top left corner, there is a small red logo with the letters 'IRRCT'. The main content is a white login form titled 'Login'. The form has two input fields: 'Email Address' with the placeholder 'a@a.com' and 'Password' with a lock icon. Below the password field is a checkbox labeled 'Remember me' and a link 'Forget your password?'. At the bottom of the form is a large blue 'LOGIN' button and a link 'Don't have an Account? Sign up'.

Authorised Users can login by entering valid Email Address And Password. Login Page ensures authorized access to the Software and ensures security to the System.

## Sign Up Page

A screenshot of a sign up page with a dark background. In the top left corner, there is a small red logo with the letters 'IRRCT'. The main content is a white sign up form titled 'Sign Up'. The form has five input fields: 'Full Name', 'Email' with a '@' icon, 'Contact No.' with a phone icon, 'Password' with a lock icon, and 'Confirm Password' with a lock icon. Below the fields is a large blue 'Sign up' button and a link 'Already have an account? Login'.

A new user can register/signup into the system by entering the details and clicking on the Sign Up button. Only valid users will be registered into the system.

## Docket No. Page

After Logging the user is redirected to the Login Page where he/she can make a new

**Ravindra Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

**Docket No.** | Lorry Hire Cum Challan | Crossing Challan | Add Station | Deal | Monthly Report |

Branch:  | Docket No.:  | Booking Date:   
 Booking From:  | Destination:  | Payment Type:  PAID  TO PAY  T.B.B.  
 Consigner Name:  | Consignee Name:  |

Docket entry or search the existing entries and update them according to the requirements.

If the User clicks the Search Button without entering any field values then all the database records are shown in the Grid. The user can also search the records by entering 1 or more field values where the grid will contain only the matching records based on field values.

**Ravindra Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

**Docket No.** | Lorry Hire Cum Challan | Crossing Challan | Add Station | Deal | Monthly Report |

Branch:  | Docket No.:  | Booking Date:   
 Booking From:  | Destination:  | Payment Type:  PAID  TO PAY  T.B.B.  
 Consigner Name:  | Consignee Name:  |

Edit	Sr.No.	DocketNo	Consigner Name	Consignee Name	Booking Date	Booking From	Destination	Pkgs	ChWt	Grand Total	Remarks
	1	7850	Ravindra Singh	Mohit Singh	04/15/2015	Delhi	Delhi	13	123	1041.21	T.B.B
	2	7878	Ravindra Singh	Mohit Singh	04/15/2015	Noida	Delhi	13	123	0.00	PAID
	3	7851	Ravindra Singh	Mohit Singh	04/15/2015	Delhi	Delhi	13	123	14638.78	PAID

Now if the user wants to update any existing records then he/she can do so by clicking on the Edit icon which will redirect him/her to the Edit Page where the data can be edited and updated by clicking on the UPDATE button.

**Ravindra Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrtransport@live.com Phone No.: 011-22126610

**Docket No.** Lorry Hire Cum Challan Crossing Challan Add Station Deal Monthly Report Bhanu Pratap

Branch: Delhi Docket No.: 7850 Booking Date: 04/15/2015

**Party Detail**

Consigner Name: Ravindra Singh Consignee Name: Mohit Singh  
 Consigner Address: Delhi Consignee Address: Noida  
 Phone No.: 9311294242 Phone No.: 12345678  
 CST/TIN No.: 123q CST/TIN No.:  
 Booking From: Delhi Destination: Delhi

**Consignment Details**

Pkgs: 13 Pkg Method: Cash Description: NEW  
 Act. Wt.: 223 Ch. Wt.: 123  
 Consignment Value: 4664.00 Invoice No.: 1212 Road Permit No.: 1212a  
 Special Note: Handle With Care.

**Payment Details**  PAID  TO PAY  T.B.B. Select Party Code: PartyRRC001

Receipt No.: Date:  
 Payment Mode: Cash Amount:  
 Received From:

**Freight Details**

A. Frt: 100  
 A.O.C: 120  
 C. Risk: 120  
 Hamali: 50  
 FOV: 50  
 Others: 450  
 Statical Charges: 120  
 Total: 1010  
 S. Tax: 31.21  
 G. Total: 1041.21  
 Freight in Words: One Thousand Forty One

Delivery Date: 04/21/2015  
 Status: Pending  
 Payment Remark: GHEHEJ

Edit	Sr.No.	DocketNo	Consigner Name	Consignee Name	Booking Date	Booking From	Destination	Pkgs	ChWt	Grand Total	Remarks
	1	7850	Ravindra Singh	Mohit Singh	04/15/2015	Delhi	Delhi	13	123	1041.21	T.B.B
	2	7878	Ravindra Singh	Mohit Singh	04/15/2015	Noida	Delhi	13	123	0.00	PAID
	3	7851	Ravindra Singh	Mohit Singh	04/15/2015	Delhi	Delhi	13	123	14638.78	PAID

If the user wants to add a new Docket entry then he can click on the ADD NEW button on the Docket No. Page

**Ravindra Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrtransport@live.com Phone No.: 011-22126610

**Docket No.** Lorry Hire Cum Challan Crossing Challan Add Station Deal Monthly Report Bhanu Pratap

Branch: --Select-- Docket No.: Booking Date:  
 Booking From: --Select-- Destination: --Select-- Payment Type:  PAID  TO PAY  T.B.B  
 Consigner Name: Consignee Name:

And then he/she will be redirected to the Add New Page where the new record can be entered and saved by clicking on the SAVE button.

The screenshot displays the 'Ravindera Road Carriers' web application interface. At the top, the company logo and contact information are visible. Below the header, a navigation menu includes 'Docket No.', 'Lorry Hire Cum Challan', 'Crossing Challan', 'Add Station', 'Deal', and 'Monthly Report'. The 'Lorry Hire Cum Challan' option is currently selected. The main form is divided into several sections: 'Party Detail' (with fields for Consignor and Consignee names, addresses, and phone numbers), 'Consignment Details' (with fields for Pkgs, Pkg Method, Description, Act. Wt., Ch. Wt., Invoice No., Road Permit No., and Special Note), 'Payment Details' (with checkboxes for PAID, TO PAY, and T.B.B., and fields for Receipt No., Date, Payment Mode, and Amount), and 'Freight Details' (with fields for A. Frt, A.O.C, C. Risk, Hamali, FOV, Others, Statical Charges, Total, S. Tax, G. Total, and Freight in Words). The form also includes a 'Delivery Date' field, a 'Status' dropdown menu (set to 'Pending'), and a 'Payment Remark' field. At the bottom right, there are 'SAVE', 'BACK', and 'Clear' buttons.

### Lorry Hire Cum Challan Page

The user can go to the Lorry Hire Cum Challan Page by clicking on the menu displaying all the Pages where he/she can make a new Lorry Hire Cum Challan entry or search the existing entries and update them according to the requirements.

The screenshot displays the 'Ravindera Road Carriers' web application interface, specifically the search page for 'Lorry Hire Cum Challan'. The navigation menu is the same as in the previous screenshot, with 'Lorry Hire Cum Challan' selected. The search form includes fields for 'Challan No.', 'Branch', 'Challan Date', 'Booking From', and 'Destination'. There are 'SEARCH' and 'ADD NEW' buttons. The 'SEARCH' button is highlighted, indicating it is the active element.

If the User clicks the Search Button without entering any field values then all the database records are shown in the Grid. The user can also search the records by entering

1 or more field values where the grid will contain only the matching records based on field values.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. **Lorry Hire Cum Challan** Crossing Challan Add Station Deal Monthly Report Bhanu Pratap

Challan No.  Branch  --Select-- Challan Date   
 Booking From  --Select-- Destination  --Select--

Edit	Sr. No	Challan No.	G.R.No.	Booking From	Destination	Payment Status
	1	1234	7850,7878,7851	Noida	Noida	Settled

Now if the user wants to update any existing records then he/she can do so by clicking on the Edit icon which will redirect him/her to the Edit Page where the data can be edited and updated by clicking on the UPDATE button.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. **Lorry Hire Cum Challan** Crossing Challan Add Station Deal Monthly Report Bhanu Pratap

Branch  Delhi Challan No.  1234 Challan Date  04-20-2015  
 Booking From  Noida Destination  Noida

**Lorry Details**  
 Vehicle No.  up12w1234 Chassis No.  12345 Engine No.  12345

**Owner Details**  
 Owner Name  bhanu Phone No.  56754 PAN No.  4554  
 Owner Address  h3g93k

**Driver Details**  
 Driver Name  ghfghj Phone No.  11111 Licence No.  jhkk7868  
 Driver Address  h3g93k

**Broker Details**  
 Broker Name  bhjkgk Phone No.  111111

**Goods Details**  
 G.R.No.  7850,7878,7851  
**Total Pkgs**  39 **Total Weight**  369 **Total to Pay**  15679.99

**Lorry Freight**  
 Freight  1200  
 Hamali  120  
 Extra for  100  
 Total  1420  
 Less Advance  100  
 Balance  1320

**Payment Details**  
 Balance Paid at  Delhi  
 Amount Paid  1320  
 Receipt No.  786  
 Payment Status  Settled  
 Payment Remarks  complete payment dfgdfgdf

Edit	Sr. No	Challan No.	G.R.No.	Booking From	Destination	Payment Status
	1	1234	7850,7878,7851	Noida	Noida	Settled



If the user wants to add a new Lorry Hire Cum Challan entry then he can click on the ADD NEW button on the Docket No. Page.

The screenshot shows the Ravindera Road Carriers web application interface. At the top, there is a logo and contact information: H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095, E-Mail: rrtransport@live.com Phone No.: 011-22126610. Below this is a navigation menu with 'Lorry Hire Cum Challan' selected. A search bar contains fields for 'Challan No.', 'Branch', 'Challan Date', 'Booking From', and 'Destination', with 'SEARCH' and 'ADD NEW' buttons.

And then he/she will be redirected to the Add New Page where the new record can be entered and saved by clicking on the SAVE button.

The screenshot shows the 'Add New' page in the Ravindera Road Carriers web application. It features a comprehensive form with several sections:
 

- General Info:** Branch, Booking From, Destination, Challan No., and Challan Date.
- Lorry Details:** Vehicle No., Chassis No., and Engine No.
- Owner Details:** Owner Name, Phone No., and PAN No.
- Driver Details:** Driver Name, Phone No., and Licence No.
- Broker Details:** Broker Name and Phone No.
- Goods Details:** G.R.No.
- Summary:** Total Pkgs (39), Total Weight (369), and Total to Pay (15679.99).
- Lorry Freight:** Freight, Hamali, Extra for, Total, Less Advance, and Balance.
- Payment Details:** Balance Paid at, Amount Paid, Receipt No., Payment Status (Pending), and Payment Remarks.

 At the bottom right, there are 'SAVE', 'BACK', and 'Clear' buttons.

## Crossing Challan Page

The user can go to the Lorry Hire Cum Challan Page by clicking on the menu displaying all the Pages where he/she can make a new Crossing Challan entry or search the existing entries and update them according to the requirements.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan **Crossing Challan** Add Station Deal Monthly Report Bhanu Pratap

Challan No.  Challan Date   
 Crossing Transport

If the User clicks the Search Button without entering any field values then all the database records are shown in the Grid. The user can also search the records by entering 1 or more field values where the grid will contain only the matching records based on field values.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan **Crossing Challan** Add Station Deal Monthly Report Bhanu Pratap

Challan No.  Challan Date   
 Crossing Transport

Edit	Sr. No	Chalan No.	G.R.No.	Booking From	Destination	Crossing Payment Status
	1	456	7850,7851,7878	Delhi	Noida	Settled

Now if the user wants to update any existing records then he/she can do so by clicking on the Edit icon which will redirect him/her to the Edit Page where the data can be edited and updated by clicking on the UPDATE button.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan **Crossing Challan** Add Station Deal Monthly Report Bhanu Pratap

Branch: Delhi | Challan No: 456 | Challan Date: 04-20-2015

**Crossing Details**  
 Crossing To (TPT): dfgdf | Phone No.: 3456436

**Lorry Details**  
 Vehicle No.: dfg54854 | Booking From: Delhi | Destination: Noida

**GR Details**  
 G.R. No.: 7850, 7851, 7878

**Freight Details**  
 Our Freight: 5000 | Crossing Freight: 3005 | Crossing Payment Status: Pending

**Payment Details**  
 Balance Freight: 15679.99 | Payment Amount: 2000 | Payment Date: 04-20-2015 | Payment Status: Settled

**Summary:**  
 Total Pkgs: 39 | Total Weight: 369 | Total to Pay: 15679.99

Edit	Sr. No	Chalan No.	G.R.No.	Booking From	Destination	Crossing Payment Status
	1	456	7850,7851,7878	Delhi	Noida	Settled

If the user wants to add a new Crossing Challan entry then he can click on the ADD NEW button on the Docket No. Page.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan **Crossing Challan** Add Station Deal Monthly Report Bhanu Pratap

Challan No.  | Challan Date

Crossing Transport  |  |

And then he/she will be redirected to the Add New Page where the new record can be entered and saved by clicking on the SAVE button.

**Ravindera Road Carriers**  
H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
E-Mail: rrtransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan **Crossing Challan** Add Station Deal Monthly Report Bhanu Pratap

Branch	<input type="text"/>	Challan No	<input type="text"/>	Challan Date	<input type="text"/>
<b>Crossing Details</b>					
Crossing To (TPT)	<input type="text"/>			Phone No.	<input type="text"/>
<b>Lorry Details</b>					
Vehicle No.	<input type="text"/>	Booking From	--Select--	Destination	--Select--
<b>GR Details</b>					
G.R. No.	<input type="text"/>				
<b>Total Pkgs</b>	39	<b>Total Weight</b>	369	<b>Total to Pay</b>	15679.99

<b>Freight Details</b>	
Our Freight	<input type="text"/>
Crossing Freight	<input type="text"/>
Crossing Payment Status	Pending
<b>Payment Details</b>	
Balance Freight	<input type="text"/>
Payment Amount	<input type="text"/>
Payment Date	<input type="text"/>
Payment Status	Pending
Payment Remarks	<input type="text"/>

SAVE BACK Clear

### Add station page

The user can select the Add Station item from the menu displaying all the items. On this the user can Add Station, Add Branch, Add Party, View Existing Stations – Edit or Delete them, View Existing Branches – Edit or Delete Them, View Existing Parties – Edit or Delete them according to the requirements.


**Ravindera Road Carriers**  
H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
E-Mail: rrtransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan Crossing Challan **Add Station** Deal Monthly Report Bhanu Pratap

<b>Add Station</b>	Station Name	<input type="text"/>	Station Code	<input type="text"/>	ADD		
<b>Add Branch</b>	Branch Name	<input type="text"/>	Branch Code	<input type="text"/>	Branch Type	<input type="text"/>	ADD
	Address	<input type="text"/>			Contact No.	<input type="text"/>	
<b>Add Party</b>	Party Name	<input type="text"/>	Party Code	<input type="text"/>	CST/TIN No.	<input type="text"/>	
	Party Address	<input type="text"/>		State	--Select--	District	<input type="text"/>
	Pin Code	<input type="text"/>	Contact No.	<input type="text"/>	Email Id	<input type="text"/>	ADD

View Stations
  View Branches
  View Parties

If the user clicks on View Station (RadioButtonList Item) then all the stations will be displayed in the Grid below where he/she can Edit or Delete them by clicking on Edit Icon or Delete Icon respectively.



### Ravindera Road Carriers

H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
E-Mail: rctransport@live.com Phone No.: 011-22126610

Docket No.
Lorry Hire Cum Challan
Crossing Challan
Add Station
Deal
Monthly Report

Bhanu Pratap

**Add Station**

Station Name  Station Code  ADD

**Add Branch**

Branch Name  Branch Code  Branch Type  ADD

Address  Contact No.

**Add Party**

Party Name  Party Code  CST/TIN No.  ADD

Party Address  State  District

Pin Code  Contact No.  Email Id


View Stations
 View Branches
 View Parties

**Stations**

	Edit	Delete	Sr. No	Station Name	Station Code
			1	Delhi	Delhi
			2	Noida	Noida
			3	Jaipur	JPR
			4	Ajmer	AJM
			5	Bhilwara	BLW
			6	Udaipur	UDP
			7	Alwar	ALW
			8	Meerut	MRT
			9	Agra	AGR
			10	Lucknow	LKN

12


If the user clicks on the Edit icon of a particular station then that record will be highlighted in the grid which shows that this particular record is being edited. If in any case the user clicks on the edit icon by mistake then he can cancel the edit operation just by clicking on the CancelEdit Icon against the highlighted record.



### Ravindera Road Carriers

H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
E-Mail: rctransport@live.com Phone No.: 011-22126610

Docket No.
Lorry Hire Cum Challan
Crossing Challan
Add Station
Deal
Monthly Report

 **Bhanu Pratap**

Add Station

Station Name

Station Code

Add Branch

Branch Name

Branch Code

Branch Type

Address

Contact No.

Add Party

Party Name

Party Code

CST/TIN No.

Party Address

State

District

Pin Code

Contact No.

Email Id

View Stations

View Branches

View Parties

Stations					
	Edit	Delete	Sl. No	Station Name	Station Code
			1	Delhi	Delhi
			2	Noida	Noida
			3	Jaipur	JPR
			4	Ajmer	AJM
			5	Bhilwara	BLW
			6	Udaipur	UDP
			7	Alwar	ALW
			8	Meerut	MRT
			9	Agra	AGR
			10	Lucknow	LKN

IJISRT17MY30

[www.ijisrt.com](http://www.ijisrt.com)

91

If the user clicks on View Branches (RadioButtonList Item) then all the branches will be displayed in the Grid below where he/she can Edit or Delete them by clicking on Edit Icon or Delete Icon respectively.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan Crossing Challan **Add Station** Deal Monthly Report Bhanu Pratap

**Add Station**  
 Station Name  Station Code

**Add Branch**  
 Branch Name  Branch Code  Branch Type   
 Address  Contact No.

**Add Party**  
 Party Name  Party Code  CST/TIN No.   
 Party Address  State  District   
 Pin Code  Contact No.  Email Id

View Stations  View Branches  View Parties

Edit	Delete	Sr.No.	Branch Name	Branch Code
		1	Delhi	Delhi
		2	Jaipur	Jaipur
		3	Noida	Noida
		4	Gr Noida	Gr Noida

If the user clicks on the Edit icon of a particular branch then that record will be highlighted in the grid which shows that this particular record is being edited. If in any case the user clicks on the edit icon by mistake then he can cancel the edit operation just by clicking on the CancelEdit Icon against the highlighted record.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan Crossing Challan **Add Station** Deal Monthly Report Bhanu Pratap

**Add Station**  
 Station Name  Station Code

**Add Branch**  
 Branch Name  Branch Code  Branch Type   
 Address  Contact No.

**Add Party**  
 Party Name  Party Code  CST/TIN No.   
 Party Address  State  District   
 Pin Code  Contact No.  Email Id

View Stations  View Branches  View Parties

Edit	Delete	Sr.No.	Branch Name	Branch Code
		1	Delhi	Delhi
		2	Jaipur	Jaipur
		3	Noida	Noida
		4	Gr Noida	Gr Noida

If the user clicks on View Parties (RadioButtonList Item) then all the parties will be displayed in the Grid below where he/she can Edit or Delete them by clicking on Edit Icon or Delete Icon respectively.

**Ravindra Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan Crossing Challan **Add Station** Deal Monthly Report Bhanu Pratap

**Add Station**  
 Station Name:  Station Code:

**Add Branch**  
 Branch Name:  Branch Code:  Branch Type:    
 Address:  Contact No.:

**Add Party**  
 Party Name:  Party Code:  CST/TIN No.:   
 Party Address:  State:  District:    
 Pin Code:  Contact No.:  Email Id:

View Stations  View Branches  View Parties

Edit	Delete	Sr.No.	Party Name	Address	State	District	Contact Number	Party Code	Email ID	CST/TIN No.
		1	Ravindra	noida	UTTAR PRADESH	noida	7777777777	PartyRRC001	a	123
		2	Ravindrahjdsfgs/fgjhj	noida	UTTAR PRADESH	noida	7777777777	PartyRRC002	a	123
		3	MacMillan Publishers India Pvt. Ltd.	Noida	UTTAR PRADESH	Gautam Buddha Nagar	01204523689	MMPI-N	sales@macmillan.co.in	ABC123456BH

If the user clicks on the Edit icon of a particular branch then that record will be highlighted in the grid which shows that this particular record is being edited. If in any case the user clicks on the edit icon by mistake then he can cancel the edit operation just by clicking on the CancelEdit Icon against the highlighted record.

**Ravindra Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan Crossing Challan **Add Station** Deal Monthly Report Bhanu Pratap

**Add Station**  
 Station Name:  Station Code:

**Add Branch**  
 Branch Name:  Branch Code:  Branch Type:    
 Address:  Contact No.:

**Add Party**  
 Party Name:  Party Code:  CST/TIN No.:   
 Party Address:  State:  District:    
 Pin Code:  Contact No.:  Email Id:


View Stations  View Branches  View Parties

Edit	Delete	Sr.No.	Party Name	Address	State	District	Contact Number	Party Code	Email ID	CST/TIN No.
		1	Ravindra	noida	UTTAR PRADESH	noida	7777777777	PartyRRC001	a	123
		2	Ravindrahjdsfgs/fgjhj	noida	UTTAR PRADESH	noida	7777777777	PartyRRC002	a	123
		3	MacMillan Publishers India Pvt. Ltd.	Noida	UTTAR PRADESH	Gautam Buddha Nagar	01204523689	MMPI-N	sales@macmillan.co.in	ABC123456BH



## Deal page


On this page the user can Save the rates he/she has agreed to with a party, he/she can save the Rate Details, Terms & Conditions, Door Delivery & Door Collection Charges, Search the Terms & Conditions records – Edit them or Delete them, Search the Door Delivery & Door Collection Charges records – Edit them or Delete them according to the requirements.



### Ravindera Road Carriers

H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No.
Lorry Hire Cum Challan
Crossing Challan
Add Station
Deal
Monthly Report

 Bhanu Pratap

Party Code

Party Name

Rate Type per Kg.

**Rate Details**

Source

Destination

Rate

Transit Time

Save Rate

**Terms And Condition**

GR Charges

Min Charge Wt

D.R. Charges

Labour Charges

Basis On per Kg.

Door Delivery Charges

Charge Basis Per Kg.

Range  To  Charge

Door Collection Charges

Same As Door Delivery Charges


Charge Basis Per Kg.

Range  To  Charge

Deal & Terms And Conditions

Door Delivery Charges & Door Collection Charges


If the user clicks the Search button against the Terms & Conditions then all the records present in the database are shown up in the Grid where the user can Edit or Delete them just by clicking on the Edit or Delete Icons respectively.



## Ravindra Road Carriers

H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
E-Mail: rctransport@live.com Phone No.: 011-22126610

Docket No.
Lorry Hire Cum Challan
Crossing Challan
Add Station
Deal
Monthly Report



Party Code 
Party Name 
Rate Type

**Rate Details**

Source 
Destination 
Rate 
Transit Time 
Save Rate

**Terms And Condition**

GR Charges 
Min Charge Wt 
D.R. Charges 
Labour Charges 
Basis On

SAVE Search

**Door Delivery Charges**

Charge Basis 
Range  To  Charge

**Door Collection Charges**

Same As Door Delivery Charges
Charge Basis

Range  To  Charge

SAVE Search

**Deal & Terms And Conditions**

Edit	Delete	Sl No	Party Code	PartyName	Rate	RateType	Source	Destination	TransitTime
		1	PartyRRC001	Ravindra Singh	100	per Kg.	Delhi	Delhi	3
		2	PartyRRC002	RRC new	100	per Bundle	Noida	Delhi	4
		3	MMPH-N	MacMillan Publishers India Pvt. Ltd.	300	per Kg.	Delhi	Jaipur	1
		4	MMPH-N	MacMillan Publishers India Pvt. Ltd.	10	per Kg.	Delhi	Delhi	1
		5	MMPH-N	MacMillan Publishers India Pvt. Ltd.	15	per Kg.	Delhi	Noida	2
		6	MMPH-N	MacMillan Publishers India Pvt. Ltd.	20	per Kg.	Delhi	Jaipur	3
		7	FBNC-N	Frank Brothers and Company.	0	per Kg.	Delhi	Noida	0
		8	FBNC-N	Frank Brothers and Company.	0	per Kg.	Delhi	Noida	0
		9	PartyRRC001	Ravindra Singh	500	per Kg.	Delhi	Noida	5

If the user clicks the Search button against the Door Delivery & Door Collection charges then all the records present in the database are shown up in the Grid where the user can Edit or Delete them just by clicking on the Edit or Delete Icons respectively.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan Crossing Challan Add Station **Deal** Monthly Report Bhanu Pratap

Party Code:  Party Name:  Rate Type:

**Rate Details**  
 Source:  Destination:  Rate:  Transit Time:

**Terms And Condition**  
 GR Charges:  Min Charge Wt:  D.R. Charges:  Labour Charges:  Basis On:

**Door Delivery Charges** **Door Collection Charges**

Charge Basis:   Same As Door Delivery Charges

Range:  To:  Charge:  Charge Basis:  Range:  To:  Charge:

**Deal & Terms And Conditions**

Door Delivery Charges & Door Collection Charges											
Edit	Delete	Sr. No	Party Code	ChargeBasis	Range	To	Charge	ChargeBasisDCC	RangeDCC	ToDCC	ChargeDCC
		1	PartyRRC001	Per Kg.	150	120	140	Per Kg.	150	120	150
		2	PartyRRC002	Per Kg.	15	120	35	Per Kg.	15	120	150
		3	MMPI-N	Per Kg.	10	100	100	Per Kg.	10	100	100
		4	PartyRRC001	Per Kg.	100	500	1000	Per Kg.	100	500	1000

**Monthly report page**

The user can select the Monthly Report from the Menu displayed on the top. On this Page the user can generate the TO Pay, Paid & T.B.B. Bills just by clicking on the Generate Excel & Generate TBB Bill buttons respectively which in turn will download the Excel Files containing the bills which can then be printed or saved for future references etc. according to the user’s requirements.

**Ravindera Road Carriers**  
 H.O.: Plot No. 9, Damodar Park, G.T. Road, Shahdara, Delhi-110095  
 E-Mail: rrctransport@live.com Phone No.: 011-22126610

Docket No. Lorry Hire Cum Challan Crossing Challan Add Station Deal **Monthly Report** Bhanu Pratap

Party Code:  Month:  Year:

## System Testing 7

---

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic.

As a rule, system testing takes, as its input, all of the "integrated" software components that have passed integration testing and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called assemblages) or between any of the assemblages and the hardware. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole.

### 7.1 UNIT TESTING

In computer programming, unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. Intuitively, one can view a unit as the smallest testable part of an application. In procedural programming, a unit could be an entire module, but it is more commonly an individual function or procedure. In object-oriented programming, a unit is often an entire interface, such as a class, but could be an individual method. Unit tests are short code fragments created by programmers or occasionally by white box testers during the development process. It forms the basis for component testing.

Ideally, each test case is independent from the others. Substitutes such as method stubs, mock objects, fakes, and test harnesses can be used to assist testing a module in isolation. Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended.

The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. A unit test provides a strict, written contract that the piece of code must satisfy.

## **7.2 INTEGRATION TESTING**

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

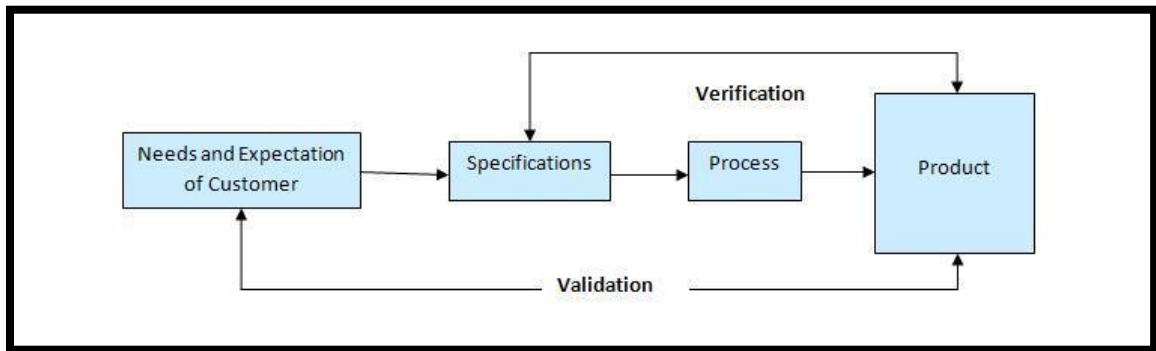
The purpose of integration testing is to verify functional, performance, and reliability requirements placed on major design items. These "design items", i.e. assemblages (or groups of units), are exercised through their interfaces using black box testing, success and error cases being simulated via appropriate parameter and data inputs. Simulated usage of shared data areas and inter-process communication is tested and individual subsystems are exercised through their input interface. Test cases are constructed to test whether all the components within assemblages interact correctly, for example across procedure calls or process activations, and this is done after testing individual modules, i.e. unit testing. The overall idea is a "building block" approach, in which verified assemblages are added to a verified base which is then used to support the integration testing of further assemblages.

Some different types of integration testing are big bang, top-down, and bottom-up. Other Integration Patterns are: Collaboration Integration, Backbone Integration, Layer Integration, Client/Server Integration, Distributed Services Integration and High-frequency Integration.

## **7.3 VALIDATION TESTING**

- Determining if the system complies with the requirements and performs functions for which it is intended and meets the organization's goals and user needs.

- Validation is done at the end of the development process and takes place after verifications are completed.
- It answers the question like: Am I building the right product?
- Am I accessing the right data (in terms of the data required to satisfy the requirement).
- It is a High level activity.
- Performed after a work product is produced against established criteria ensuring that the product integrates correctly into the environment.
- Determination of correctness of the final software product by a development project with respect to the user needs and requirements.



**Fig 7.1 Software verification and validation**

According to the Capability Maturity Model(CMMI-SW v1.1) we can also define validation as The process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements. [IEEE-STD-610].

A product can pass while verification, as it is done on the paper and no running or functional application is required. But, when same points which were verified on the paper is actually developed then the running application or product can fail while validation. This may happen because when a product or application is build as per the specification but these specifications are not up to the mark hence they fail to address the user requirements.

Advantages of Validation:

- During verification if some defects are missed then during validation process it can be caught as failures.

- If during verification some specification is misunderstood and development had happened then during validation process while executing that functionality the difference between the actual result and expected result can be understood.
- Validation is done during testing like feature testing, integration testing, system testing, load testing, compatibility testing, stress testing, etc.
- Validation helps in building the right product as per the customer's requirement and helps in satisfying their needs.

Validation is basically done by the testers during the testing. While validating the product if some deviation is found in the actual result from the expected result then a bug is reported or an incident is raised. Not all incidents are bugs. But all bugs are incidents. Incidents can also be of type 'Question' where the functionality is not clear to the tester. Hence, validation helps in unfolding the exact functionality of the features and helps the testers to understand the product in much better way. It helps in making the product more user friendly.

## 7.4 TEST CASES

Tc_Id	Component	Summary	Input data	Expected Result	Actual Result	Steps to Reproduce	Pass/Fail
Tc_001	Login	To verify login works properly or not					
	A. Email Address	1:To verify with valid data	<a href="mailto:a@a.com">a@a.com</a>	Should allow login and be redirected to the Docket No. Page	Redirects to the Docket No. Page	Enter Email Address in the Email Address textbox	Pass
	B. Password		1			Enter Password in the Password textbox > Press LOGIN button	Pass
		2:To verify with invalid	<a href="mailto:b@b.com">b@b.com</a>	Should not allow login	Login not allowed and the Login form	Enter Email Address in the Email Address	Pass

		data			pops up with blank fields	textbox	
			2			Enter Password in the Password textbox > Press LOGIN button	Pass
		3:To verify with blank fields		Must give message for mandatory fields	Login not allowed and the Login form pops up with blank fields	Press LOGIN button	Pass
Tc_002	Sign Up	To verify Signup form works properly or not					
	A. Full name	1. Verify with valid data	abc	Should Sign Up successfully and redirect to Login Form	Redirects to the Login Form	Enter Full Name > Email > Contact No. > Password > Confirm Password > Sign up	Pass
	B. Email		<a href="mailto:an@an.com">an@an.com</a>				
	C. Contact No.		9876543210				
	D. Password		123				
	E. Confirm Password		123				
		2. Verify with invalid data	1234	Should not allow successful Sign Up	Sign Up successful and redirects to the Login Form	Enter Full Name > Email > Contact No. > Password > Confirm Password > Sign up	Fail
			1234				
			Abcd				
			1				



			1				
		3. Verify with blank fields		Should not allow successful Sign Up	Sign Up not successful and Login form pops up	Press Sign up button	Pass
Tc_003	Docket No.	To verify docket no. tab					
	A. Search using single field values						
	1.Branch	Verify with valid data	Select value from Dropdown	On_click of search button must show the correct result where Branch = selected value	Shows correct results where Branch = selected value	Click on Docket No. tab > select branch > click search button	Pass
	2.Booking From	Verify with valid data	Select value from Dropdown	On_click of search button must show the correct result where Booking From = selected value	Shows correct results where Booking From = selected value	Click on Docket No. tab > select booking from > click search button	Pass
	3. Consigner Name	Verify with valid data	Enter Consigner Name into the textbox	On_click of search button must show the correct result where Consigner Name = entered	Shows correct results where Consigner Name = entered value	Click on Docket No. tab > enter consigner name > click search button	Pass

				value			
	4. Docket No.	Verify with valid data	Enter Docket No. into the textbox	On_click of search button must show the correct result where Docket No. = entered value	Shows correct results where Docket No. = entered value	Click on Docket No. tab > enter Docket No. > click search button	Pass
	5. Destination	Verify with valid data	Select value from Dropdown	On_click of search button must show the correct result where Destination = selected value	Shows correct results where Destination = selected value	Click on Docket No. tab > select Destination > click search button	Pass
	6. Consignee Name	Verify with valid data	Enter Consignee Name into the textbox	On_click of search button must show the correct result where Consignee Name = entered value	Shows correct results where Consignee Name = entered value	Click on Docket No. tab > enter Consignee Name > click search button	Pass
	7. Booking Date	Verify with valid data	Pick Date from the DatePicker	On_click of search button must show the correct result where Booking Date = selected value	Shows correct results where Booking Date = selected value	Click on Docket No. tab > select Booking Date > click search button	Pass
	8. Payment Type	Verify with valid data	Check Payment Type from the CheckBoxLi	On click of search button must show the correct	Shows correct results where Payment	Click on Docket No. tab > select Payment Type > click	Pass

			st	result where Payment Type = Checked value	Type = Checked value	search button	
	B. Search using more than one or all fields	Verify with valid data	Enter various filed values	On click of search button must show the correct result based on the entered filed values	Shows correct results based on the entered field values	Click on Docket No. tab > enter various filed values > click search button	Pass
	C. Search without entering any field values	Verify with blank fields		On_click of search button must show the correct results	Shows correct results with entire database records	Click on Docket No. tab > click search button	Pass
	D.Add New Records	Verify Add New works properly or not	Enter various field values	On click of Add New button must successfully save the data into the database	Data saved successfully in the database	Click Docket No. tab > click Add New Button > Enter all the field values or atleast mandatory field values > click SAVE Button	Pass
	E. Add New Records without entering any field values	Verify Add New works properly or not		On click of Add New Button must show required field validations	Displays message box showing required fields	Click Docket No. tab > click Add New Button > click SAVE Button	Pass
	F. Clear all the Filled values	To Verify Clear works properly or not	Enter various field values	On click of clear button must clear all the fields	Clears all the field values	Click on Docket No. tab > enter various field values > click CLEAR Button	Pass

	G. Edit the existing records	To verify Edit works properly or not	Enter the value to be edited	On click of edit must allow edit operation	Redirected to the edit page where u can edit the fields the fields you want to change, the records get updated successfully	Click on Docket No. tab > click search button click on the edit icon > edit the fields u want to change > click UPDATE	Pass
Tc_004	Lorry Hire Cum Challan	To verify Lorry Hire Cum Challan tab					
	A. Search using single field values						
	1.Branch	Verify with valid data	Select value from Dropdown	On_click of search button must show the correct result where Branch = selected value	Shows correct results where Branch = selected value	Click on Lorry Hire Cum Challan tab > select branch > click search button	Pass
	2.Booking From	Verify with valid data	Select value from Dropdown	On_click of search button must show the correct result where Booking From = selected value	Shows correct results where Booking From = selected value	Click on Lorry Hire Cum Challan tab > select booking from > click search button	Pass
	3. Challan No.	Verify with valid data	Enter Challan No. into the textbox	On_click of search button must show	Shows correct results where	Click on Lorry Hire Cum Challan tab > enter	Pass

				the correct result where Challan No. = entered value	Challan No. = entered value	Challan No.> click search button	
	4. Challan Date	Verify with valid data	Select Challan Date from the calendar control	On_click of search button must show the correct result where Challan Date. = entered value	Shows correct results where Challan Date = entered value	Click on Lorry Hire Cum Challan tab > enter Challan Date > click search button	Pass
	5. Destination	Verify with valid data	Select value from Dropdown	On_click of search button must show the correct result where Destination = selected value	Shows correct results where Destination = selected value	Click on Lorry Hire Cum Challan tab > select Destination > click search button	Pass
	B. Search using more than one or all fields	Verify with valid data	Enter various field values	On click of search button must show the correct result based on the entered filed values	Shows correct results based on the entered field values	Click on Lorry Hire Cum Challan tab > enter various filed values > click search button	Pass
	C. Search without entering any field values	Verify with blank fields		On_click of search button must show the correct results	Shows correct results with entire database records	Click on Lorry Hire Cum Challan tab > click search button	Pass
	D.Add New Records	Verify Add New works properly or not	Enter various field values	On click of Add New button must successfull	Data saved successfull y in the database	Click Lorry Hire Cum Challan tab > click Add New Button	Pass

				y save the data into the database		> Enter all the field values or atleast mandatory field values > click SAVE Button	
	E. Add New Records without entering any field values	Verify Add New works properly or not		On click of Add New Button must show required field validations	Displays message box showing required fields	Click Lorry Hire Cum Challan tab > click Add New Button > click SAVE Button	Pass
	F. Clear all the Filled values	To Verify Clear works properly or not	Enter various field values	On click of clear button must clear all the fields	Clears all the field values	Click on Lorry Hire Cum Challan tab > enter various field values > click CLEAR Button	Pass
	G. Edit the existing records	To verify Edit works properly or not	Enter the value to be edited	On click of edit must allow edit operation	Redirected to the edit page where u can edit the fields the fields you want to change, the records get updated successfull y	Click on Lorry Hire Cum Challan tab > click search button click on the edit icon > edit the fields u want to change > click UPDATE	Pass
Tc_005	Crossing Challan	To verify Crossing Challan tab					
	A. Search using single field values						
	1. Challan No.	Verify with	Enter Challan No. into the	On_click of search button	Shows correct results	Click on Lorry Hire Cum Challan	Pass

		valid data	textbox	must show the correct result where Challan No. = entered value	where Challan No. = entered value	tab > enter Challan No.> click search button	
	2. Challan Date	Verify with valid data	Select Challan Date from the calendar control	On_click of search button must show the correct result where Challan Date. = entered value	Shows correct results where Challan Date = entered value	Click on Lorry Hire Cum Challan tab > enter Challan Date > click search button	Pass
	3. Crossing Transport	Verify with valid data	Enter Crossing Transport into the textbox	On_click of search button must show the correct result where Crossing Transport = entered value	Shows correct results where Crossing Transport = entered value	Click on Crossing Challan tab > enter Crossing Transport > click search button	Pass
	B. Search using more than one or all fields	Verify with valid data	Enter various filed values	On click of search button must show the correct result based on the entered filed values	Shows correct results based on the entered field values	Click on Crossing Challan tab > enter various filed values > click search button	Pass
	C. Search without entering any field values	Verify with blank fields		On_click of search button must show the correct results	Shows correct results with entire database records	Click on Crossing Challan tab > click search button	Pass
	D.Add New Records	Verify Add New works	Enter various field values	On click of Add New button	Data saved successfull y in the	Click Crossing Challan tab	Pass

		properly or not		must successfully save the data into the database	database	>click Add New Button > Enter all the field values or atleast mandatory field values > click SAVE Button	
	E. Add New Records without entering any field values	Verify Add New works properly or not		On click of Add New Button must show required field validations	Displays message box showing required fields	Click Crossing Challan tab > click Add New Button > click SAVE Button	Pass
	F. Clear all the Filled values	To Verify Clear works properly or not	Enter various field values	On click of clear button must clear all the fields	Clears all the field values	Click on Crossing Challan tab > enter various field values > click CLEAR Button	Pass
	G. Edit the existing records	To verify Edit works properly or not	Enter the value to be edited	On click of edit must allow edit operation	Redirected to the edit page where u can edit the fields you want to change, the records get updated successfully	Click on Crossing Challan tab > click search button click on the edit icon > edit the fields u want to change > click UPDATE	Pass
Tc-006	Add Station	To verify Add Station tab					
	A. Add a new station	To verify Add Station works properly or not	Enter Station Name And Station Code	On Click of Add Button must Add a new Station into the	Station added successfully into the database	Click on add station tab > Enter Station Name & Station Code > Click Add	Pass



				database			
	B. Add a New Branch	To verify Add Branch works properly or not	Enter Branch Name, Branch Code, Branch Type, Address, Contact No. into the respective Textboxes	On Click of Add Button must Add a new Branch into the database	Branch added successfully into the database	Click on add station tab > Branch Name, Branch Code, Branch Type, Address, Contact No. > Click Add	Pass
	C. Add a New Party	To verify Add Party works properly or not	Enter Party Name, Party Code, Party Address, State, Pincode, Contact No., CST/TIN No., Email ID, District into respective Textboxes	On Click of Add Button must Add a new Party into the database	Party added successfully into the database	Click on add station tab > Enter Party Name, Party Code, Party Address, State, Pincode, Contact No., CST/TIN No., Email ID, District > Click Add	Pass
	D. View Stations	To verify View station works properly or not	Click on the View Station Radio Button	On Selection of View Station Radio Button must show the Grid containing the station details	Grid shown properly	Click on Add Station tab > Click View Station Radio Button	Pass
	E. View Branches	To verify View Branches works properly or not	Click on the View Branches Radio Button	On Selection of View Branches Radio Button must show the Grid containing the branch details	Grid shown properly	Click on Add Station tab > Click View Branches Radio Button	Pass
	F. View	To verify	Click on the	On	Grid	Click on Add	Pass

	Parties	View Parties works properly or not	View Parties Radio Button	Selection of View Parties Radio Button must show the Grid containing the party details	shown properly	Station tab > Click View Parties Radio Button	
	G. Edit Station	To verify Edit Staion works properly or not	Enter the new value into the textbox	On click of UPDATE button must successfully update the data into the database	Data updated successfully	Click Add Station tab > click View station > click Edit icon > enter the new value in the textbox > click update	Pass
	H. Edit Branches	To verify Edit Branches works properly or not	Enter the new value into the textbox	On click of UPDATE button must successfully update the data into the database	Data updated successfully	Click Add Station tab > click View Branches > click Edit icon > enter the new value in the textbox > click update	Pass
	I. Edit Parties	To verify Edit Parties works properly or not	Enter the new value into the textbox	On click of UPDATE button must successfully update the data into the database	Data updated successfully	Click Add Station tab> click View Parties > click Edit icon > enter the new value in the textbox > click update	Pass
	J. Delete Station	To verify Delete station works properly or not		On click of Delete icon must delete the data from the grid as well as database	Data deleted successfully from the grid as well as database	Click Add Station tab> Click View Station > click Delete icon	Pass

	K. Delete Branches	To verify Delete Branches works properly or not		On click of Delete icon must delete the data from the grid as well as database	Data deleted successfully from the grid as well as database	Click Add Station tab > Click View Branches > click Delete icon	Pass
	L. Delete Parties	To verify Delete Parties works properly or not		On click of Delete icon must delete the data from the grid as well as database	Data deleted successfully from the grid as well as database	Click Add Station tab > Click View Parties > click Delete icon	Pass
Tc_007	Deal	To verify Deal tab					
	A. Save Rate	To verify Save Rate works properly or not	Enter Party Code, Source, Destination, Rate, TRansit Time in the respective textboxes	On click of Save Rate Button must save the data into the database	Data Saved successfully in the database	Click Deal tab > Enter Party Code, Source, Destination, Rate, TRansit Time > click Save Rate	Pass
	B. Save Terms & Condition	To verify Save Terms & Condition works properly or not	Enter Party Code, G.R. Charges, Min. ChWt, D.R. Charges, Labour Charges, Basis on into the respective textboxes	On click of Save Button must save the data into the database successfully	Data saved successfully into the database	Click Deal tab > Enter Party Code, G.R. Charges, Min. ChWt, D.R. Charges, Labour Charges, Basis on > click Save	Pass
	C. Save Door Delivery Charges & Door Collection Charges	To verify Save Door Delivery Charges & Door Collection Charges	Enter Party Code, Range, To, Basis in the respective textboxes	On click of Save Button must save the data into the database successfully	Data Saved successfully in the database	Click Deal tab > Enter Party Code, Range, To, Basis > click Save	Pass

		works properly or not					
	D. Search Terms & Conditions	To verify Search Terms & Condition works properly or not	Enter 0 or more fields	On click of Search Button must show the grid containing the Terms & Conditions based on the field values	Grid shown properly	Click Deal tab > Click Search button against the Terms & Condition	Pass
	1. Edit Terms & Conditions	To verify Edit works properly or not		On click of Edit must allow edit operation	Data updated successfully	Click Deal tab > Click Search button against the Terms & Condition > click Edit icon > Enter the new value > Click Update	Pass
	2. Delete Terms & Conditions	To verify delete works properly or not		On click of Delete icon must delete the data from the grid as well as database	Data deleted successfully from the grid as well as database	Click Deal tab > Click Search button against the Terms & Condition > Click Delete icon	Pass
	E. Search Door Delivery Charges & Door Collection Charges	To verify Search Terms & Condition works properly or not	Enter 0 or more fields	On click of Search Button must show the grid containing the Terms & Conditions based on the field values	Grid shown properly	Click Deal tab > Click Search button against the Door Delivery Charges & Door Collection Charges	Pass
	1. Edit Door	To verify Edit		On click of Edit must	Data updated	Click Deal > Click Search	Pass

	Delivery Charges & Door Collection Charges	works properly or not		allow edit operation	successfully	button against the Terms & Condition > click Edit icon > Enter the new value > Click Update	
	2. Delete Door Delivery Charges & Door Collection Charges	To verify delete works properly or not		On click of Delete icon must delete the data from the grid as well as database	Data deleted successfully from the grid as well as database	Click Deal tab > Click Search button against the Door Delivery Charges & Door Collection Charges > Click Delete icon	Pass
Tc_008	Monthly Report	To verify Monthly Report tab					
	A. Generate Excel	To verify Generate Excel works properly or not	Enter Party Code, Month & Year in the respective fields	On click of Generate Excel must download the Excel file containing PAID and TO PAY bills	Excel file downloaded successfully	Click Monthly Report tab > Enter Party Code, Month & Year > click Generate Excel	Pass
	B. Generate TBB Bill	To verify Generate TBB Bill works properly or not	Enter Party Code, Month & Year in the respective fields	On click of Generate TBB Bill must download the Excel file containing TBB bills	Excel file downloaded successfully	Click Monthly Report tab > Enter Party Code, Month & Year > click Generate TBB Bill	Pass

## CONCLUSION

---

The main purpose of making this project “Transport management system” is to make the Work easier and less time consuming. With the help of this project, the data can be stored and retrieved easily.

A Transportation Management System (TMS) is a system which makes the task of consigner and consignee easier. In this modern world everyone wants everything to be done quickly.

So, this website has been build to make the task of consigner and consignee easier and less time consuming.

## REFERENCES

---

### Websites Referred

- <http://en.wikipedia.org/wiki/ASP.NET>
- [www.aspsnippets.com](http://www.aspsnippets.com)
- <http://www.c-sharpcorner.com>
- <http://www.w3schools.com/aspnet/>

### Books referred

- Beginning ASP.NET 4.5.1 and Professional ASP.NET MVC 5
- Getting Started with ASP.NET 4.5 Web Forms and Visual Studio 2013
- Mastering Visual C# .NET