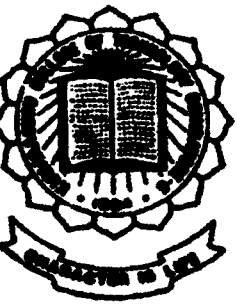


B2B PORTAL

P-511

AND

VEHICLE MANAGEMENT SYSTEM



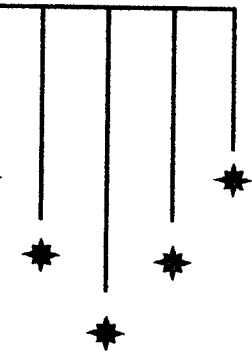
Project Report Submitted by,

S. Krishna Prabha

B. Jayamurugan

R. Jaikannan

S. Arun



Guided by,

Ms. A. Lavanya

Lecturer, Dept of CSE

2001

Submitted in partial fulfillment of the requirements for the award of the degree
of BACHELOR OF ENGINEERING in COMPUTER SCIENCE & ENGINEERING of
Bharathiyar University, Coimbatore

Department of Computer Science and Engineering

Kumaraguru College of Technology

Certificate

This is to certify that the project entitled

B2B PORTAL and VEHICLE MANAGEMENT SYSTEM

Submitted to the

Department of Computer Science and Engineering
Kumaraguru College of Technology
(Affiliated to Bharathiar University, Coimbatore)

in partial fulfillment of the requirement for the

Degree of Bachelor of Engineering in Computer Science & Engineering
is a bonafide record of work carried out by,

Ms. S. Krishna Prabha Reg no. – 9727K0151

Mr. R. Jaikannan Reg no. – 9727K0143

Mr. B. Jayamurugan Reg no. – 9727K0144

Mr. S. Arun Reg no. – 9727K0128

during their period of study in the department of Computer Science and Engineering,
Kumaraguru College of Technology, under my supervision and guidance.

.....
Ms. A. Lavanya
Staff In charge & Guide)

.....
Dr. S. Thangaswamy
(Prof & Head of Department)

Submitted for University Examination held on

.....
Internal Examiner

.....
External Examiner

Certificate

To Whomsoever It May Concern

Certified that this thesis "B2B Portal and Vehicle Management System" is the bonafide work of Ms.S.Krishna Prabha, Mr.R.Jaikannan, Mr.S.Arun & Mr.B.Jayamurugan who have carried out the project under our supervision, certified further that to the best of our knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate. The Documentation submitted by the group was well compiled and the project completely satisfied our expectations.

Director

A. Ganesh Kumar.

Project Guide

Kumaresh . K

ACKNOWLEDGEMENT

The Satisfaction that accompanies the successful completion of any task would be but incomplete without the mention of the people whose constant guidance and encouragement crowns all effort with success.

We express our heart felt thanks to our beloved principal **Dr. K.K. Padmanaban** for providing the amenities and opportunity for completing this project.

Our sincere thanks to **Dr. S. Thangaswamy**, Head of the Department of Computer Science and Engineering, Kumaraguru College of Technology for providing us this opportunity to take up the project work. He has been a source of great encouragement and inspiration throughout the curriculum. Our sincere thanks to him for his immense help and guidance during the course of the project.

We would like to express our gratitude to our internal guide, **Ms. A. Lavanya**, Lecturer, Department of Computer Science & Engineering, Kumaraguru college of Technology, who guided us throughout the project and encouraged us to successfully complete this project.

We thank **Mr. Ganesh Kumar** of Bharath Advanced Technologies for granting us this project work. We also thank **Mr. Kumaresh** of Bharath Advanced Technologies our external guide for his excellent support and timely help in completing our project.

We also extend our heartiest thanks to all other members of staff and

SYNOPSIS

The project titled “**B2B Portal**” is a comprehensive venture to develop an e-commerce website for BAT Earth Movers. The project also includes the development of B2B application software ie “**Vehicle management system**” for the same company. The details of the various vehicles used by the different branches of the company and their schedule is stored in a central database and immediate access of information can be obtained from the database from any part of the world.

This project consists of intricate operations that help the user to know the status of the vehicle at any given time such as

- Maintaining the details of the vehicle
- Keeping track of the daily reports of the vehicle
- Computing the consolidated monthly report from the daily report
- Portraying the oil consumption of vehicles for each month of every vehicle using Graphs and Charts.
- Generating various reports such as Breakdown analysis report, Spare part reminders etc.

The website features the B2B e-commerce where the exchange of information takes place between businesses rather than between businesses and consumers. It paves way for transactions through multiple channels including e-marketplaces and electronic procurement systems. Immediate updates are made possible so that the administrator can have full control over any vehicle at any time.

The website was developed using HTML and ASP. The B2B application software was developed using Visual Basic 6.0 with its native backend Microsoft Access 2000.

CONTENTS

1. *Introduction*

1.1	About the Organization.....	1
1.2	Existing System.....	1
1.3	Why B2B?.....	2
1.4	Need for B2B.....	3

2. *Requirements Analysis*

2.1	Hardware Requirements.....	5
2.2	Software Requirements.....	5
2.3	A glimpse of the technologies and software used.....	6

3. *Design*

3.1	Database design.....	30
3.2	Dataflow Diagrams.....	34



4. *Implementation*

4.1	Vehicle Management System.....	37
4.2	B2B Portal.....	38

5. *Conclusion*.....39

APPENDIX

A	- Visual Basic User Interfaces.....	40
B	- Visual Basic Sample Code.....	45
C	- HTML and ASP User Interface.....	59
D	- HTML and ASP sample code.....	62

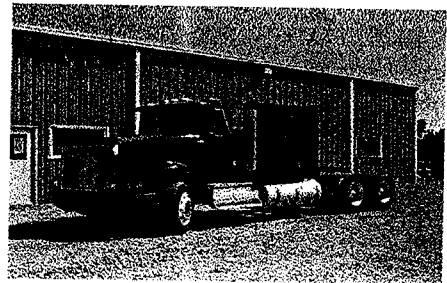
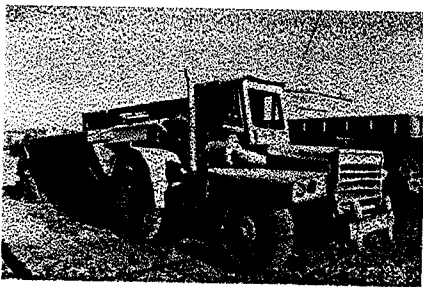


Introduction

1. INTRODUCTION

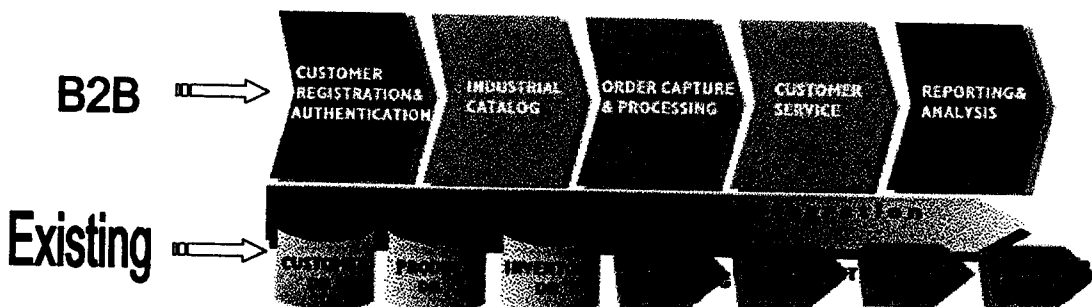
1.1 About the organization

Bharat Earthmovers is a division of BAT India, Coimbatore. The company is mainly involved in the earth moving operations. Apart from offering their services the company also rents out their equipments for use in various parts of India. The company is well established and in the process of modernizing all their facilities and equipments. The company owns about fifty tippers, forty excavators and twenty loaders.



1.2 Existing System

The existing system in BAT earthmovers is only manual entry in computers, notebooks and notes for the drivers and service personnel. The management of the system becomes tedious and inaccurate. Also the data isn't centralized so any error in the entry elsewhere in the country could lead to disasters in terms of billing and replacement of spare parts at the right time. The figure below shows how the existing system is going to be integrated with B2B technologies to make the system efficient.



1.3 Why B2B?

On the Internet, B2B (business-to-business), also known as e-biz, is the exchange of products, services, or information between businesses rather than between businesses and consumers. Although early interest centered on the growth of retailing on the Internet. Forecasts are that B2B revenue will far exceed business-to-consumers (B2C) revenue in the near future. According to studies published in early 2000, the money volume of B2B exceeds that of e-tailing by 10 to 1. Over the next five years, B2B is expected to have a compound annual growth of 41%. The Gartner Group estimates B2B revenue worldwide to be \$7.29 trillion dollars by 2004.

B2B Web sites can be sorted into:

- ✧ **Company Web sites** since the target audience for many company Web sites is other companies and their employees. Company sites can be thought of as round-the-clock mini-trade exhibits. Sometimes a company Web site serves as the entrance to an exclusive extranet available only to customers or registered site users. Some company Web sites sell directly from the site, effectively e-tailing to other businesses.

- ✧ **Product supply and procurement exchanges** where a company purchasing agent can shop for supplies from vendors, request proposals, and, in some cases, bid to make a purchase at a desired price. Sometimes referred to as e-procurement sites, some serve a range of industries and others focus on a niche market.

- ✧ **Specialized or vertical industry portals** which provide a "subWeb" of information, product listings, discussion groups, and other features. These

vertical portal sites have a broader purpose than the procurement sites (although they may also support buying and selling).

- ✧ **Brokering sites** that act as an intermediary between someone wanting a product or service and potential providers. Equipment leasing is an example.

- ✧ **Information sites** which provide information about a particular industry for its companies and their employees. These include specialized search sites and trade and industry standards organization sites.

This project deals with the development of a Company Web Site for Bharth Earth Movers and also a customized application software for use in the company.

1.4 **Need for B2B**

⌘ **Reduced Purchase costs**

One of the easiest ways that a company can cut costs is by remodeling the way it purchases spare parts and other raw goods. The National Association of Purchasing Managers says that the average manual purchase order costs a company \$79. This is because locating goods needed and then filling out the necessary paper work is a labor-intensive process. Searching for spare parts online requires much less time and electronically processing an order streamlines the ordering procedure.

⌘ **Increased Market efficiency**

Using the Internet, companies can quickly and easily get price quotes from numerous suppliers. By increasing the number of sellers, buyers are more likely to get a better price, and vice versa. Just as eBay has created


an efficient market for everything from Barbie dolls to old Atari games, B2B hosts make connections between buyers and sellers that may not have otherwise happened. The vehicle can be delivered to the customer in the shortest time by checking the nearest vehicle available to the customers site. Thus suppressing other potential companies.

↻ **Greater Market intelligence**

Related to finding good prices, B2B hosts give producers a better insight into the demand levels in any given market. Spot price levels can quickly be determined in everything from oils to wheels. This allows companies to make better decisions regarding when to buy and when not to.

↻ **Decreased Inventory levels**

Using B2B technologies, companies can better utilize their inventory and raw materials. The Internet allows even more time to be shaved off for companies that use "just in time" loading and off loading techniques. In essence, it allows firms to use less working vehicles to do the same amount of work, freeing some of these vehicles to be used elsewhere.



Requirements Analysis

*Dedicated to our
beloved Parents*

2. REQUIREMENTS ANALYSIS

2.1 Hardware Requirements

1. Pentium II 350 Mhz or equivalent
2. Color Monitor
3. Graphics Card
4. 50mb hard disk space
5. Internet Connection
6. Modem

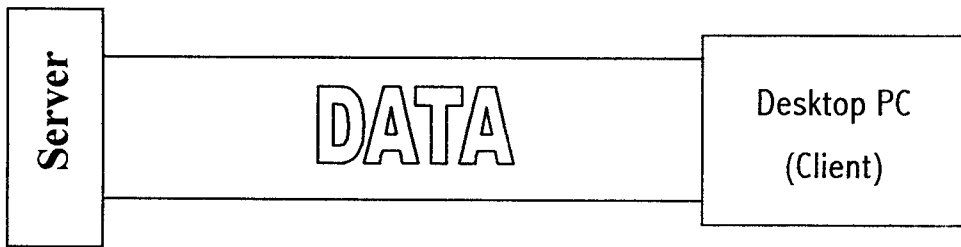
2.2 Software Requirements

1. Windows platform
2. Visual Studio 6 -Enterprise Edition
3. MS Access
4. Personal Web Server
5. MS Paint
6. MS PowerPoint
7. MS Front page

2.3 A glimpse at the technologies and software used

2.3.1 What is Client-Server Technology?

Client-Server computing simply means that two (or more) process run independently in a cooperative manner. The simplest and most common example is the classic two-tierd database application shown.



Here we see a client program communicating with a database engine running on a remote server and the two are connected via a network .The client is responsible for providing an interface to the user. Typically, the client will create an SQL request for data and send that request to the database. The database then evaluates the request, fulfills it and sends the data back to the client.

In case of multi-tiered architectures, the load on the database server will be high and to overcome this a secondary server namely, an application server is used.

The two-tiered client/server model is also known as thin client. In this the PC becomes totally network independent. If it can run Internet Explorer, it is powerful enough to run any of the client/server applications.

2.3.3.2 Oracle

Oracle 8 runs on a wide variety of platforms (Windows NT, Unix and OS/2). A personal edition of Oracle runs on Windows 95/98. Oracle has long enjoyed a reputation as the most powerful database on the market with advanced features optimizing large volume data processing, distributed data management, web connectivity, and robust OLTP (On Line Transaction Processing).

2.3.3.3 Microsoft SQL Server

Microsoft SQL server runs on windows NT only. Although that obviously might be limiting, it also offers some advantages. Historically, the RDBMS engine has had to be written with a separate layer to interact with the operating system, perform its own thread management, and so on. Because SQL server is written for only one operating system, it can be more closely integrated with the operating system.

2.3.4 Reasons for using MS Access as Project Backend

- ☞ Easy to modify database.
- ☞ Easy coding.
- ☞ Ability to transport database easily.
- ☞ Lesser validations to check for.
- ☞ MS Access is available on almost every system.

2.3.5 The Various data access methods

2.3.5.1 Data access Objects (DAO)

The proprietary COM interface to Microsoft's JET database engine version 3.5 and earlier. The Data Access Objects provide support for JET databases, other ISAM (Indexed Sequential Access Method) databases through installable ISAM drivers, and ODBC databases.

With the data control, DAO defaults to JET. To use ODBC direct with the data control, set the default property to 1(use ODBC).

When using DAO objects in code, after creating the workspace object, create a database object for JET data models or create a connection object to ODBC Direct. The DAO object that we interact the most with is RecordSet object. The RecordSet object itself is the program's near exclusive interface to the database. It represents either all of the records in a base table or the rows generated from a query.

2.3.5.1.1 What is JET ?

JET stands for Joint Engine Technology; it is Microsoft's single tier data engine. JET runs locally on each user workstation. Data can be stored locally or on a network server. JET ships as the native database engine in Microsoft Access, and it also ships with Microsoft Visual Basic.

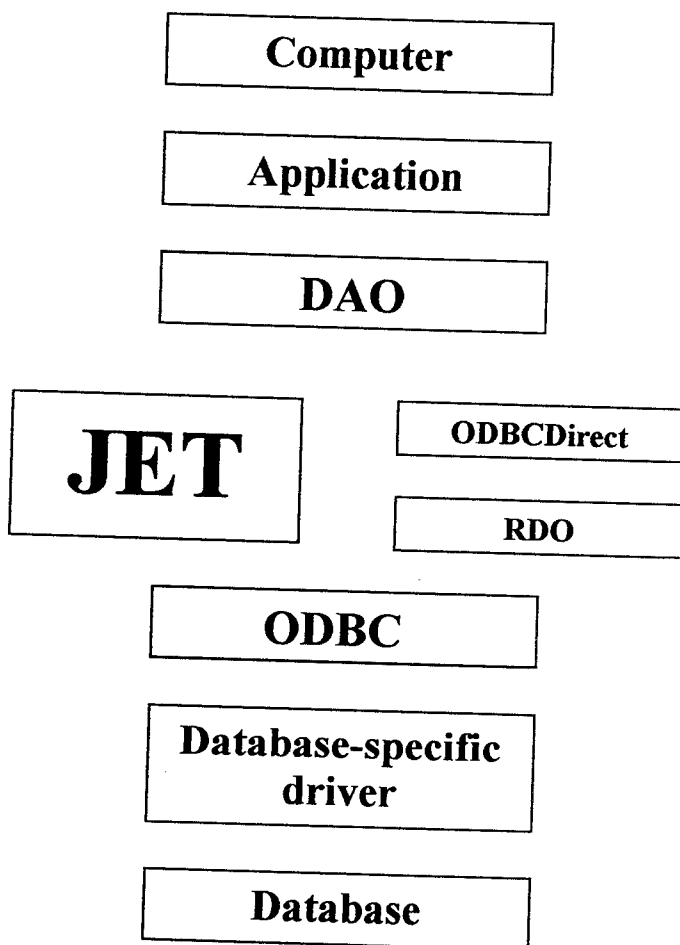
2.3.5.1.2 Open Database Connectivity (ODBC)

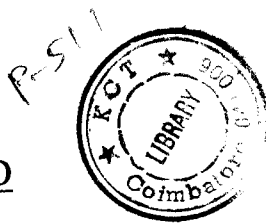
A component of the windows open services Architecture that defines a generic standard for accessing databases. ODBC defines a call-level API set and a DQL grammar conformance requirement.

2.3.5.2 Remote Data Objects (RDO)

RDO version 1 was introduced with Visual Basic 4 and was tremendously improved with RDO version 2, released with Visual Basic 5. RDO offers a compelling alternative to DAO for those who do not use the personal database in the system.

The main objects of RDO are rdoEnvironments, which is similar to the DAO workspaces. The rdoEnvironment object contains the rdoConnection collection. Each rdoConnection collection object consists of the rdoQueries, rdoResultsets etc.





2.3.5.3 Differences between RDO and DAO

RDO is table and row oriented, whereas DAO is file and record oriented; RDO places more emphasis on procedures and result sets, whereas DAO's primary emphasis is on the retrieval itself. RDO leaves the details of data retrieval to the ODBC driver.

2.3.5.4 ActiveX Data Objects

ADO allows your application to access any data that is exposed via an OLE DB interface. Further our applications can freely relate any OLE DB data source to any other OLE DB data source. ADO is to OLE DB what RDO is to ODBC. It is essentially a low-overhead wrapper around the OLE DB API. It adds key support for building client/server application residing on traditional networks or on web-based networks, including the internet.

2.3.5.4.1 OLE DB

OLE DB can be considered analogous to ODBC. It provides an interface to applications that make disparate data sources look as though they were the same data source. It logically organizes structured or unstructured data into rows and columns so that it can be accessed in a common way. An OLE DB data consumer is an application that uses OLE DB provided data. A data provider is a software component that exposes its data via an OLE DB interface.

2.3.5.4.2 ADO Objects

ADO supports an event driven access model. Therefore declaration is done using WithEvents clause. The connection object represents an open connection to a data source. The RecordSet is similar to DAO RecordSet .

2.3.6 Introduction to Visual Basic 6.0

Visual Basic is an ideal programming language for developing sophisticated professional applications for Microsoft windows. It makes use Graphical User Interface for creating robust and powerful applications. The Graphical User Interface as the name suggests, uses illustrations for text, which enables users to interact with an application. This feature makes it easier to comprehend things in a quicker and easier way.

2.3.6.1 History

Visual Basic was developed from the BASIC programming language. In the 1970s, Microsoft started developing ROM-based interpreted BASIC for the early microprocessor based computers. In 1982, Microsoft Quick Basic revolutionized Basic and was legitimized as a serious development language for the MS-DOS environment. Later on Microsoft Corporation created the enhanced version of BASIC called Visual Basic for windows.

2.3.6.2 Requirements

In order to run Visual Basic 6.0, a computer must have atleast a 486 processor and a minimum of 16 MB of RAM. A complete installation of the most powerful edition i.e. Visual Basic 6.0 Enterprise Edition requires 250 MB of hard disk space and Internet Explorer V 4.0 or higher.

2.3.6.3 Event driven programming

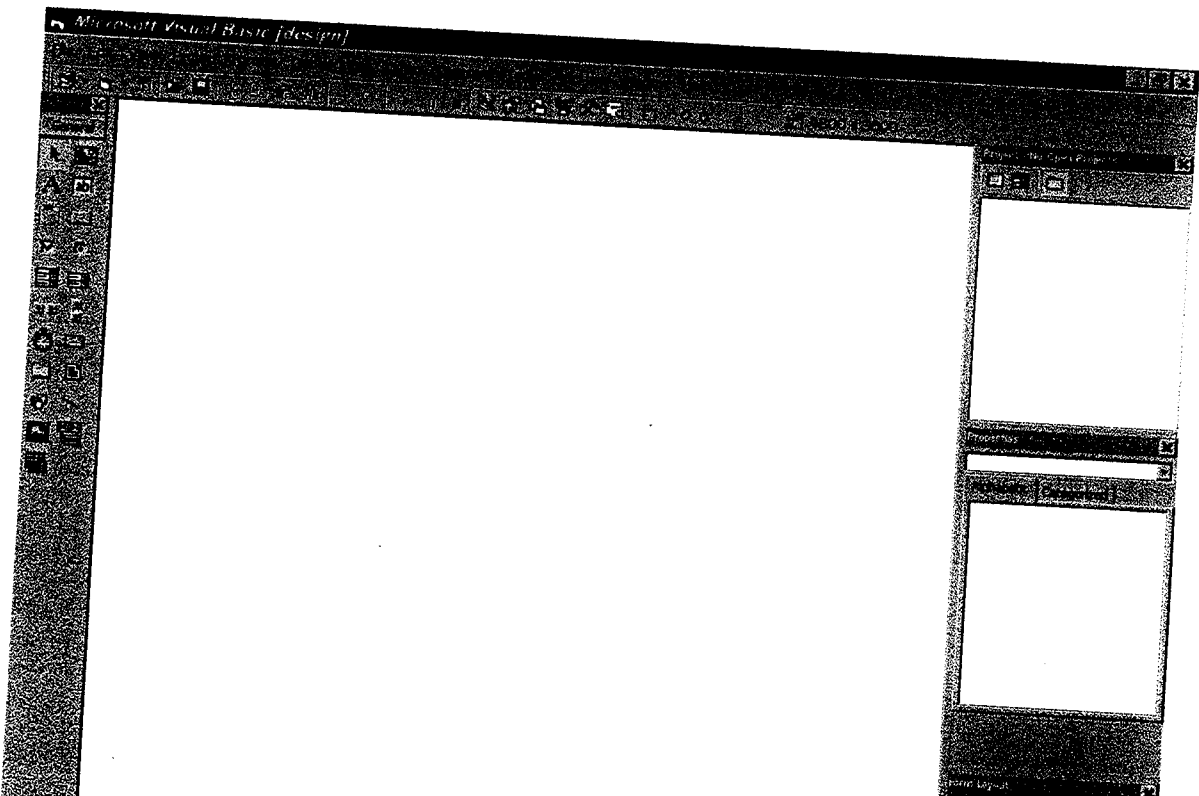
Visual Basic programs are built around events. Events are various things that can happen in a program. The program statements are executed only when a particular event calls a specific part of the code that is assigned to the event.

2.3.6.4 IDE environment

IDE refers to Integrated Development Environment because we can access virtually all of the development tools that we need from one screen called the interface .It is the design environment .The Visual Basic IDE is made up of the following components,

- ❖ **Menu bar** - Displays commands that are required to build an application.
- ❖ **Toolbox** - Provides a set of controls that are used to place on the form .
- ❖ **Project explorer** - Quick reference to various elements of the project.
- ❖ **Properties window** - Exposes characteristics of selected objects.
- ❖ **Form layout window** - Displays where and how a form is at runtime.
- ❖ **Toolbar** - Easy access to commonly used commands.
- ❖ **Form designer** - Helps in designing of the form.
- ❖ **Object Browser** - Allows us to browse through various objects.

The Visual Basic IDE



2.3.7 New features of Visual Basic 6.0

- ❖ VB 6.0 integrated Visual Database Tools and new Data Environment Designer can visually design oracle and Microsoft SQL server databases and create reusable data access queries – all without leaving the visual basic environment.
- ❖ Highly interactive web pages can be programmed as easily as a visual basic form with the new Dynamic HTML page designer.
- ❖ VB 6.0 introduces ADO (Activex Data Object) as a powerful new standard for data access.
- ❖ Data Report Designer can quickly drag-and-drop custom data-bound controls to create forms or reports.
- ❖ Custom data-aware Com controls can be created for client or middle-tier and custom OLE DB providers.
- ❖ VB 6.0 supports mobile computing support.
- ❖ More wizards, tools, components...

2.3.8 A few things about html

This specification defines the HyperText Markup Language (HTML), version 4.0, the publishing language of the World Wide Web. In addition to the text, multimedia, and hyperlink features of the previous versions of HTML, HTML 4.0 supports more multimedia options, scripting languages, style sheets, better printing facilities, and documents that are more accessible to users with disabilities. HTML 4.0 also takes great strides towards the internationalization of documents, with the goal of making the Web truly World Wide.

HTML 4.0 is an SGML application conforming to International Standard ISO 8879 -- *Standard Generalized Markup Language* [ISO8879]). As an SGML application, the syntax of conforming HTML 4.0 documents is defined by the combination of the SGML declaration and the document type definition (DTD). This specification defines the intended interpretation of HTML 4.0 elements and adds syntax constraints that may not be expressed by the DTD alone.

2.3.8.1 What is the World Wide Web?

The World Wide Web is a network of information resources. The Web relies on three mechanisms to make these resources readily available to the widest possible audience:

1. A uniform naming scheme for locating resources on the Web (e.g., URLs).
2. Protocols, for access to named resources over the Web (e.g., HTTP).
3. Hypertext, for easy navigation among resources (e.g., HTML).

The ties between the three mechanisms are apparent throughout this specification.

2.3.8.1.1 Introduction to URLs

Every resource available on the Web -- HTML document, image, video clip, program, etc. -- has an address that may be encoded by a *Uniform Resource Locator*, or "URL".

URLs typically consist of three pieces:

1. The naming scheme of the mechanism used to access the resource.
2. The name of the machine hosting the resource.
3. The name of the resource itself, given as a path.

Consider the URL that designates the current HTML specification:

`http://www.w3.org/TR/PR-html4/cover.html`

This URL may be read as follows: There is a document available via the HTTP protocol residing on the machine `www.w3.org`, accessible via the path `/TR/PRhtml4/cover.html`. Other schemes you may see in HTML documents include `"mailto"` for email and `"ftp"` for FTP.

Here is another example of a URL. This one refers to a user's mailbox:

...this is text...

For all comments, please send email to

`Joe Cool`.

2.3.8.1.2 Fragment identifiers

Some URLs refer to a location within a resource. This kind of URL ends with `"#"` followed by an anchor identifier (called the "fragment identifier"). For instance, here is a URL pointing to an anchor named `section_2`:

`http://somesite.com/html/top.html#section_2`

2.3.8.1.3 Relative URLs

A *relative* URL doesn't contain any naming scheme information. Its path generally refers to a resource on the same machine as the current document. Relative

URLs may contain relative path components (".." means one level up in the hierarchy defined by the path), and may contain fragment identifiers.

Relative URLs are resolved to full URLs using a base URL. As an example of relative URL resolution, assume we have the base URL "http://www.acme.com/support/intro.html". The relative URL in the following markup for a hypertext link:

```
<A href="suppliers.html">Suppliers</A>
```

would expand to the full URL "http://www.acme.com/support/suppliers.html", while the relative URL in the following markup for an image

```
<IMG src="../icons/logo.gif" alt="logo">
```

would expand to the full URL "http://www.acme.com/icons/logo.gif".

In HTML, URLs play a role in these situations:

- Linking to another document or resource.
- Linking to an external style sheet or script.
- Images, objects and applets for inclusion in a page.
- Image maps.
- Form submission.
- Frames
- Citing an external reference.
- Referring to metadata conventions describing a document.

3.8.2 What is HTML?

To publish information for global distribution, one needs a universally understood language, a kind of publishing mother tongue that all computers may

potentially understand. The publishing language used by the World Wide Web is HTML (from HyperText Markup Language).

HTML gives authors the means to:

- Publish online documents with headings, text, tables, lists, photos, etc.
- Retrieve online information via hypertext links, at the click of a button.
- Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products, etc.
- Include spread-sheets, video clips, sound clips, and other applications directly in their documents.

2.3.8.3 A brief history of HTML

HTML was originally developed by Tim Berners-Lee while at CERN, and popularized by the Mosaic browser developed at NCSA. During the course of the 1990s it has blossomed with the explosive growth of the Web. During this time, HTML has been extended in a number of ways. The Web depends on Web page authors and vendors sharing the same conventions for HTML. This has motivated joint work on specifications for HTML.

HTML 2.0 was developed under the aegis of the Internet Engineering Task Force (IETF) to codify common practice in late 1994. HTML+ (1993) and [\[HTML30\]](#) (1995) proposed much richer versions of HTML. Despite never receiving consensus in standards discussions, these drafts led to the adoption of a range new features. The efforts of the World Wide Web Consortium's HTML working group to codify common practice in 1996 resulted in HTML 3.2 Changes from HTML 3.2 are summarized here.

Most people agree that HTML documents should work well across different browsers and platforms. Achieving interoperability lowers costs to content providers since they must develop only one version of a document. If the effort is not made, there is much greater risk that the Web will devolve into a proprietary world of

incompatible formats, ultimately reducing the Web's commercial potential for all participants.

Each version of HTML has attempted to reflect greater consensus among industry players so that the investment made by content providers will not be wasted and that their documents will not become unreadable in a short period of time.

HTML has been developed with the vision that all manner of devices should be able to use information on the Web: PCs with graphics displays of varying resolution and color depths, cellular telephones, hand held devices, devices for speech for output and input, computers with high or low bandwidth, and so on.

2.3.8.4 HTML 4.0

HTML 4.0 extends HTML with mechanisms for style sheets, scripting, frames, embedding objects, improved support for right to left and mixed direction text, richer tables, and enhancements to forms, offering improved accessibility for people with disabilities.

2.3.8.4.1 Internationalization

This version of HTML has been designed with the help of experts in the field of internationalization, so that documents may be written in every language and be transported easily around the world. This has been accomplished by incorporating [RFC2070], which deals with the internationalization of HTML.

One important step has been the adoption of the ISO/IEC:10646 standard as the document character set for HTML. This is the world's most inclusive standard dealing with issues of the representation of international characters, text direction, punctuation, and other world language issues.

HTML now offers greater support for diverse human languages within a document. This allows for more effective indexing of documents.

2.3.8.4.2 Accessibility

As the Web community grows and its members diversify in their abilities and skills, it is crucial that the underlying technologies be appropriate to their specific needs. HTML has been designed to make Web pages more accessible to those with physical limitations. HTML 4.0 developments in the area of accessibility include:

- Encouraging the use of style sheets (rather than tables) to achieve layout effect.
- Making it easier to provide alternate (textual and aural) descriptions of images for non-visual browsers.
- Providing labels for form fields
- Providing labeled hierarchical groupings for form fields.
- Providing the ability to associate a longer text description with an HTML element.

Authors who design pages with accessibility issues in mind will not only receive the blessings of the accessibility community, but will benefit in other ways as well: well-designed HTML documents that distinguish structure and presentation will adapt more easily to new technologies.

2.3.8.4.3 Tables

The new table model in HTML is based on [RFC1942]. Authors now have greater control over structure and layout (e.g., column groups). The ability of designers to recommend column widths allows user agents to display table data incrementally (as it arrives) rather than waiting for the entire table before rendering.

Note. At the time of writing, some HTML authoring tools rely extensively on tables for formatting, which may easily cause accessibility problems.

2.3.8.4.4 Compound documents

HTML now offers a standard mechanism for embedding generic media objects and applications in HTML documents. The OBJECT element (together with its more specific ancestor elements IMG and APPLET) provides a mechanism for including images, video, sound, mathematics, specialized applications, and other objects in a document. It also allows authors to specify a hierarchy of alternate renderings for user agents that don't support a specific rendering.

2.3.8.4.5 Style sheets

Style sheets simplify HTML markup and largely relieve HTML of the responsibilities of presentation. They give both authors and users control over the presentation of documents -- font information, alignment, colors, etc.

Style information can be specified for specific elements or groups of elements either within an HTML document or in separate style sheets.

The mechanism for associating a style sheet with a document is independent of the style sheet language.

Before the advent of style sheets, authors had limited control over rendering. HTML 3.2 included a number of attributes and elements offering control over alignment, font size, and text color. Authors also exploited tables and images as a means for laying out pages. The relatively long time it takes for users to upgrade their browsers means that these features will continue to be used for some time. However, since style sheets offer more powerful presentation mechanisms, the World Wide Web Consortium will eventually phase out many of HTML's presentation elements and attributes. Throughout the specification elements and attributes at risk are marked as "deprecated". They are usually accompanied by examples of how to achieve the same effects with other elements or style sheets.

This specification includes three Document Type Definitions (DTDs) that may be used to validate HTML 4.0 documents. One for use with framesets, a loose DTD for transitional documents and a strict DTD that excludes presentation elements and attributes.

2.3.8.4.6 Scripting

Through scripts, authors may create "smart forms" that react as users fill them out. Scripting allows designers to create dynamic Web pages, and to use HTML as a means to build networked applications. The mechanisms provided to associate HTML with scripts are independent of particular scripting languages.

2.3.8.4.7 Printing

Sometimes, authors will want to make it easy for users to print more than just the current document. When documents form part of a larger work, the relationships between them can be described using the HTML LINK element or using W3C's Resource Description Language

2.3.8.5 Designing documents with HTML 4.0

We recommend that authors and implementors observe the following general principles when working with HTML 4.0.

2.3.8.5.1 Separate structure and presentation

HTML has its roots in SGML which has always been a language for the specification of structural markup. As HTML matures, more and more of its presentational elements and attributes are being replaced by other mechanisms, in particular style sheets. Experience has shown that separating the structure of a document from its presentational aspects reduces the cost of serving a wide range of

platforms, media, etc., and facilitates document revisions

2.3.8.5.2 Consider universal accessibility to the Web

To make the Web more accessible to everyone, notably those with disabilities, authors should consider how their documents may be rendered on a variety of platforms: speech-based browsers, braille-readers, etc. We do not recommend that designers limit their creativity, only that they consider alternate renderings in their design. HTML offers a number of mechanisms to this end (e.g., the alt attribute, the accesskey attribute, etc.)

Furthermore, authors should keep in mind that their documents may be reaching a far-off audience with different computer configurations. In order for documents to be interpreted correctly, designers should include in their documents information about the natural language and direction of the text, how the document is encoded, and other issues related to internationalization.

2.3.8.5.3 Help user agents with incremental rendering

By carefully designing their tables and making use of new table features in HTML 4.0, designers can help user agents render documents more quickly. Authors can learn how to design tables for incremental rendering in the definition of the TABLE element. Implementors should consult the notes on tables in the appendix for information on incremental algorithms.

2.3.9 Active Server Pages

2.3.9.1 Introduction to E-Commerce

The explosive growth of Internet Commerce has captured the public's imagination. Not so long ago creating web sites especially commerce enabled web sites, was a task best fit to MIT graduate students. No choice was left but to wrestle with the impenetrable syntax of a language like Perl or work with a low-level programming language like C++.

Fortunately, Microsoft has developed a technology that enables you to quickly create commercial Web sites: Active Server Pages (ASP). Using Active Server Pages, you can create web sites of the same quality as Dell.com or BarnesandNoble.com.

(Both sites were created using Active Server Pages).

2.3.9.2 What is E-Commerce?

E-Commerce refers to the process of buying or selling a product or service over an electric Network. The most popular medium in which E-Commerce is conducted is the Internet.

E-commerce encompasses three types of business transactions. First, a transaction can occur between a business and a customer. When you think of E-Commerce, this type of transaction is the first thing that springs to mind. A prime example of a business that engages in business-to-customer E-Commerce is Amazon. Amazon promotes itself as the "place to find and discover anything you want to buy online" by selling books, CDs, electronics and videos to customers.

Business-to-customer E-commerce can also include services. A subscription site that doesn't sell any tangible goods can also be engaged in E-Commerce. For example, Match.com –the online dating service- sells subscription to their web site to enable customers to browse their listings for potential romantic partners.

A second general form of description of E-Commerce involves transactions between one business and another. A business that engages in the type of E-Commerce is typically less visible to consumers and therefore, to the general public. A good example of a company that engages in business-to-business E-Commerce is Cisco Systems. Cisco Systems creates much of the physical infrastructure of the Internet that allows business to communicate.

Finally, a form of E-Commerce that has been very popular over the past couple of years involves customer-to-customer transactions. The best-known example of a company that engages in this type of E-Commerce is eBay. eBay enables its customers to auction items to other customers.(eBay collects a fee from every transaction.)

When one thinks of E-Commerce, we typically think of a customer selecting a product from a Website and paying for it online with a credit card. In other words, credit card transactions would appear to be an essential part of E-Commerce. However, E-Commerce might encompass only the activities leading up to the purchase and not the final purchase itself.

2.3.9.4 Microsoft Technologies for E-Commerce

2.3.9.4.1 Microsoft Web Server and Microsoft Database

Microsoft offers two Web servers: the Personal Web Server and Internet Information Server. It is necessary that one of these Web servers installed on the computer (Advanced Programming requires IIS).

Databases: Microsoft offers two databases which is easy and simple, yet powerful enough too use with- Microsoft Access and Microsoft SQL Server. However, With minor modifications, other databases such as Oracle can also be used.

2.3.9.4.2 Microsoft Personal Web Server

The Microsoft Personal Web Server works with Windows 95, Windows 98, or Windows NT Workstation. A very low traffic web site can be hosted using this and alternatively, a prototype of the Website can be created before transferring the contents of the site to Internet Information Server.

2.3.9.4.3 Microsoft Internet Information Server

When the website is ready to be launched over the Internet, you'll need IIS. Unlike the Personal Web Server, Internet Information Server can support hundreds or even thousands of simultaneous users. Some of the largest web sites on the Internet use Internet Information Server. Microsoft uses IIS for its own Web site www.microsoft.com. The Microsoft site is the fourth busiest site on the Internet receiving 5 million visitors a day.

The Internet Information Server isn't compatible with Windows 95 or Windows 98. You will need Windows NT Server or Windows 2000 Server. It's

2.3.9.4.4 Microsoft Access

To create a commercial Web site, a database to store product and order information is necessary. Microsoft Access is part of Microsoft Office family of products.

Microsoft Access is a desktop database and not a Client/Server database like SQL server. Because Microsoft Access is a desktop Database, you should use it only for prototyping your Website or for low traffic Web sites. In general Microsoft Access cannot support more than 30 concurrent Users.

After creating the website with Microsoft Access, the Database can be upgraded to SQL Server by tools commonly called as “Upsizing Tools”. The tools are included with Microsoft Access 2000.

2.3.9.4.5 Microsoft SQL Server

Unlike Microsoft Access, Microsoft SQL Server 7.0 can scale to support thousands of concurrent users and terabyte sized databases. For all intents and purposes, SQL Server can be used to support an online store. Some of the largest Web site on the Internet is using SQL Server including Dell, Buy.com, Barnes and Noble, and 1-800-flowers.com.

There are three versions of SQL Server 7.0:

SQL Server Desktop

SQL Server Standard Edition

SQL Server Enterprise Edition

SQL Server Desktop will work with Windows 95, Windows 98 and

Windows NT Workstation

SQL Server Standard Edition was designed to work with Windows NT Server and Windows 2000 Server.

Finally, the Enterprise Edition is an enhanced version of the standard edition that supports more memory, more processors, clustering, and Online Analytical Processing (OLAP) services.

2.3.9.4.6 Microsoft Visual InterDev

Microsoft Visual InterDev is developing environment for building Websites. At its more basic level, it's a very fancy editor that allows creating and modifying Web pages, on a remote or local server. Visual InterDev can be used for both normal HTML pages and Active Server Pages. Visual InterDev works with any ODBC or OLE DB compliant databases. It also includes server-debugging tools.

2.3.9.5 What is an ASP Page?

An ASP page is any file located on your Web server that has the extension .ASP. This special extension distinguishes an ASP page from a normal HTML file that ends with the extension .HTML or .HTM.

When a user visits a webs site and requests a normal HTML file, the web server simply retrieves the file from the computer's hard drive or memory and sends the file to the user's browser. The browser interprets the HTML content of the file and the visitor sees the Web page.

When someone requests a normal HTML page, the web server doesn't care about the contents of the file. The Web Server's role is to simply retrieve the appropriate file without processing it. The user's Web browser performs all the work of interpreting the content of the files.

On the other hand, when someone requests an ASP page, the Web server takes a more active role. Before the file is sent to the user's Web browser, the Web server first processes it. The Web server interprets and executes any script in an ASP page before sending it to the user's Web browser.

Active Server Pages include server-side scripts. An ASP page can contain scripts written in VBScript or Microsoft Jscript or PerlScript.

The advantage of using scripting languages to build Web pages is that it makes easy to modify a website even after it has been launched. When an ASP page is changed, it is automatically recompiled the next time it is requested.

2.3.9.6 Active Server Pages Contain Objects and Components

An ASP page would be severely limited if it could contain only scripts. Fortunately, an ASP page contains server-side components.

A component is something that typically has methods, properties and collections. A component's method determines the actions taking part in the project. A Component's property can be read or set to specify the state of the component. Component's collections are set of keys and value pairs related to the component.

Active Server pages includes two types of Components

- ✧ The built-in Objects
- ✧ Installed Components

Built in Active Server Page Components

- ⌘ Request Object
- ⌘ Response Object
- ⌘ Server Object
- ⌘ Session Object

In addition to the built in Objects, several installed components are bundled with ASP. Some of them are...

1. Ad Rotator Component
2. Browser Capabilities Component
3. Content Linking Component

A special set of Objects known as ActiveX Data Objects enables to access a Database from an ASP page. ActiveX components can be used to Insert, Delete, Update rows in a Database Table.



Design

3. DESIGN

3.1 Database Design

The following tables give a vivid description of the database.

Table Name – History

S.NO	FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
1.	Year_of_manufacture	Long	4	Year of manufacture
2.	Makename	Text	50	Makers Name
3.	Chasis_no	Text	50	Chasis Number
4.	Eng_sr_no	Text	50	Engine Serial Number
5.	Reg_no	Text	50	Registration Number
6.	Dop	Date/Time	8	Date of Purchase
7.	Hand_of_vehicle	Text	50	Hand of vehicle
8.	Kmsrun	Long	4	Kms run at time of purchase
9.	Lastservice	Date/Time	8	Last Service Date
10.	Eqchanged	Text	250	Equipments changed
11.	Eqnotchanged	Text	250	Equipments not changed
12.	Next_service	Date/Time	8	Next Service date

Purpose

The History table maintains the details of the vehicles such as Registration Number, Chassis number and other details along with the service details of the corresponding vehicle.

Table Name - Monthly remarks

S.No	FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
1.	Regno	Text	50	Registration Number
2.	Date	Date/Time	8	Date
3.	Remarks	Text	250	Remarks of the Month

Purpose

Table Name – Monthly Report

S.NO	FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
1.	Regno	Text	50	Registration Number
2.	Totalworking	Long	4	Total working Hours
3.	Totalkms	Long	4	Total Kilometres
4.	Totalhrs	Long	4	Total Hours
5.	Totaldiesel	Long	4	Total Diesel consumed
6.	Totaloils	Long	4	Total oils consumed
7.	Dieselconsum	Long	4	Diesel consumption per Km
8.	Greaseconsum	Long	4	Grease consumption per Km
9.	Month	Text	50	Month
10.	Year	Long	4	Year

Purpose

The consolidated values are calculated from the daily report table and stored in the monthlyreport table.

Table Name – Default Spare part reminders

S.NO	FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
1.	Regno	Text	50	Registration Number
2.	Date	Date/Time	8	Date
3.	StartingKm	Long	4	Starting Km
4.	ClosingKm	Long	4	Closing Km
5.	Sparepartname	Text	4	Spare Part Name
6.	Sparepartcode	Text	4	Spare Part Code

Purpose

The Default Spare part reminders table holds all the details of the spare parts being changed and when to change it.

Table Name – Daily Items

S.NO	FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
1.	Regno	Text	50	Registration Number
2.	Date	Date/Time	8	Date
3.	Oil	Long	4	Oil(Litres)
4.	Oilfilter	Long	4	Oil Filter
5.	Dieselfilter	Long	4	Diesel Filter

Purpose

The Daily Items table contains the details of the daily items to be changed for each vehicle.

Table Name - Passwd

S.NO	FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
1.	Username	Text	50	User name
2.	Password	Text	50	Password
3.	Entry	Long	4	Entry Level
4.	Report	Long	4	Report Level
5.	Modi	Long	4	Modification Level

Purpose

The details of the username, password and permissions of various access levels are stored in the Passwd table.

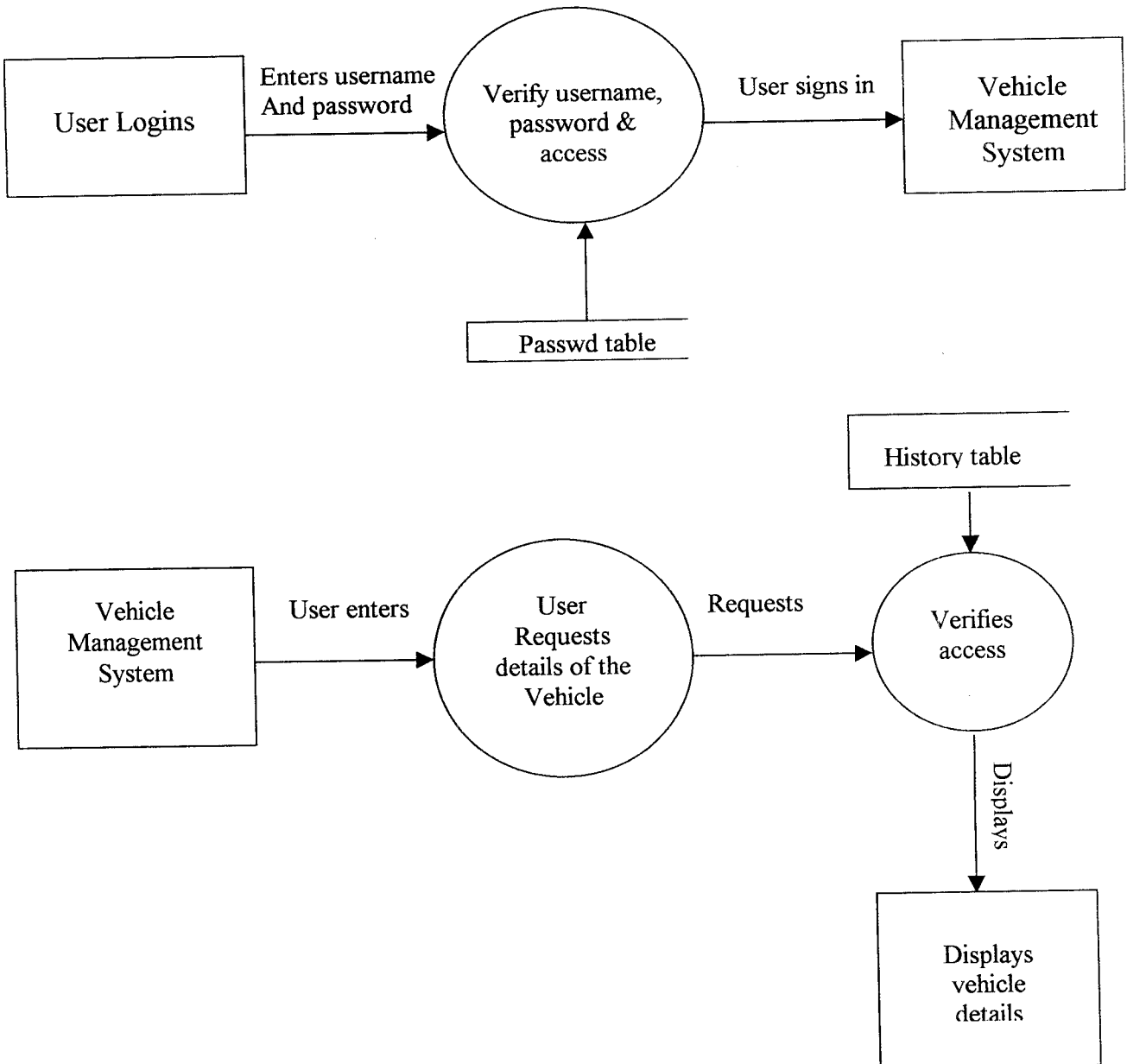
Table Name – Daily report entry

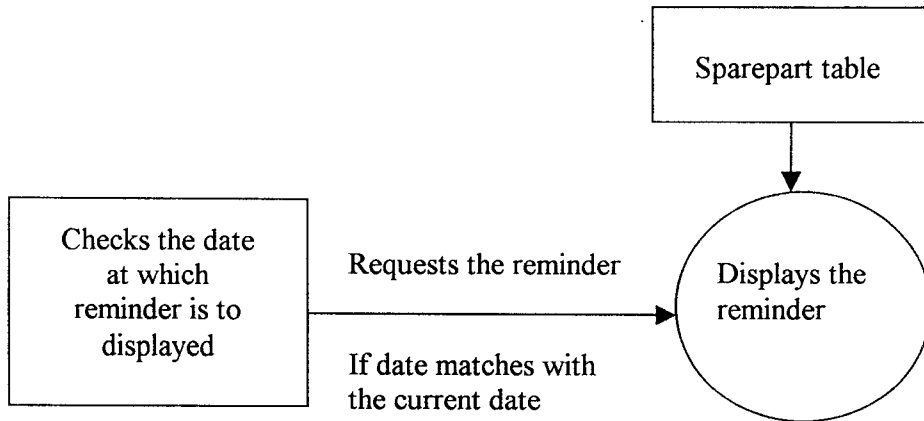
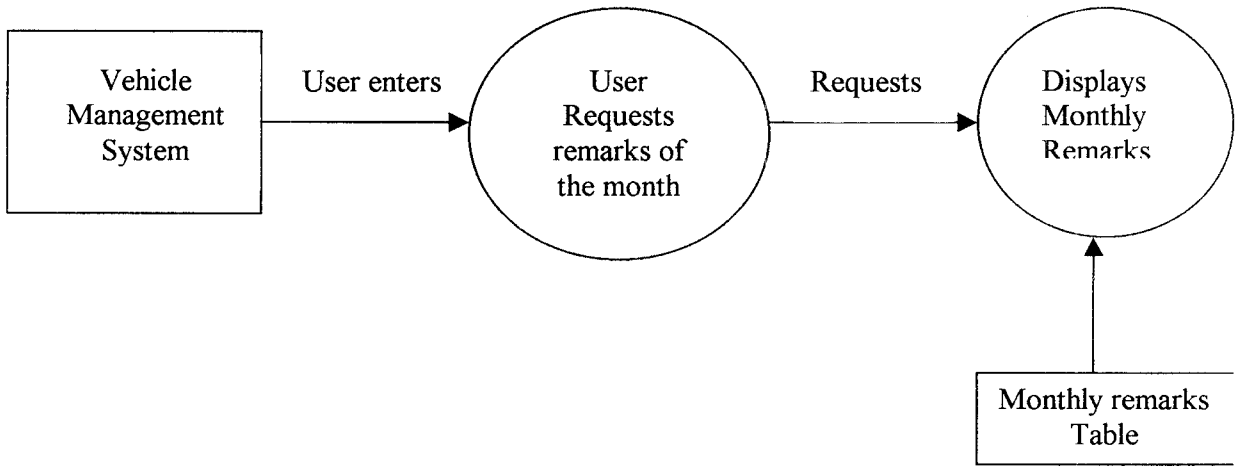
S.NO	FIELD NAME	DATATYPE	SIZE	DESCRIPTION
1.	Date	Date/Time	8	Date
2.	Regno	Text	50	Registration Number
3.	Sitename	Text	50	Site Name
4.	Starting	Long	4	Starting Km
5.	Closing	Long	4	Closing Km
6.	Working	Long	4	Working Km
7.	Totalbill	Long	4	Total Billing Hours
8.	Diesel	Long	4	Diesel(Ltrs)
9.	Engoils	Long	4	20-40 Engine Oils(Ltrs)
10.	Gradeoils	Long	4	60-98-100-140 oils(Ltrs)
11.	Grease	Long	4	Grease(Ltrs)
12.	Oldoils	Long	4	Old Oil(Ltrs)
13.	Cumhours	Long	4	Cumulative Hours
14.	Spareparts	Text	250	Spare Parts Changed
15.	Remarks	Text	250	Remarks

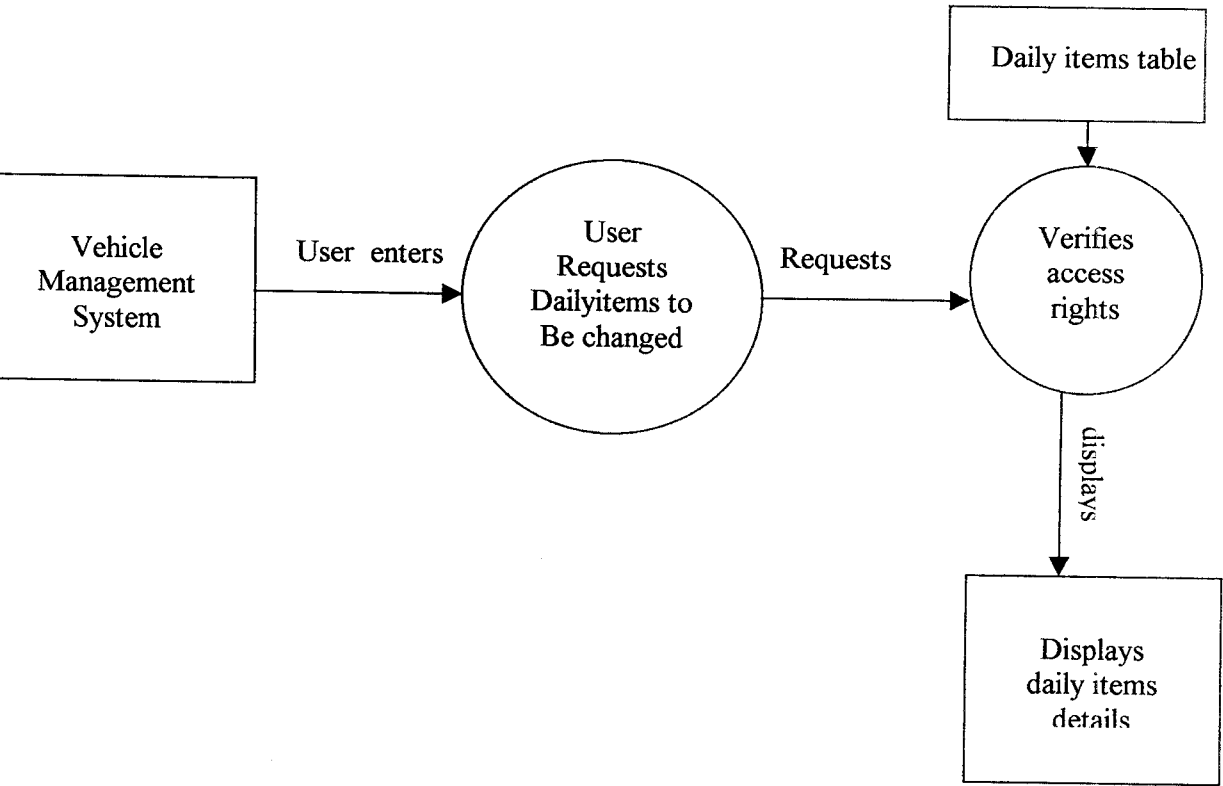
Purpose

The Daily Report table contains the details regarding the quantity of the various oils, diesel consumed and also the working hours of the vehicle. Furthermore, this table the site in which the vehicle is currently working and also its current status.

3.2 Dataflow Diagrams









Implementation

4. IMPLEMENTATION

4.1 Vehicle Management System

The Vehicle Management System application was created using Visual Basic 6.0 as the front-end and MS Access as the backend.

- Data Access Method - ADO coding.
- Security for users Entry Level, Report level, Modification Level.
- All Errors are trapped in the front end.
- Easy Interface
- Standard tool bars in all the forms
- Reports using Data Reports
- Only the Vehicle Registration Number is required to generate reports.
- Help facility on all the forms

The Sample coding is given in APPENDICES A & B

4.2 B2B Portal

The B2B portal was created using HTML and ASP. ASP Pages are used for data access from the system. The B2B portal presents an Internet Format of the Vehicle Management System with all the features including security and help. This Portal will be linked to the Companies Website to provide a centralized database.

All manipulations performed in the application can also be carried out in the web site and can be carried out from any part of the world once posted on the web. Various reports and administrator facilities are implemented in the web site and hence provide excellent support for the members using it.

The sample coding and the user interfaces are given in the APPENDICES – C & D.



Conclusions

5. CONCLUSION

The Project on “B2B Portal and Vehicle Management System” was successfully completed with its focus mainly on the e-commerce buzz word B2B. It is an emerging technology and in near future all most every day to day activity will have its place in the B2B e-commerce. The main aim was to create an interactive website featuring the various operations in a huge vehicle management system and presenting the user with immediate access of information regarding the status of the vehicle at any given time. Various tools such as Graphs and Charts were used to give a visual idea and an user friendly interface.

We thoroughly enjoyed doing the project and hope this experience will be beneficial for us in future endeavors.



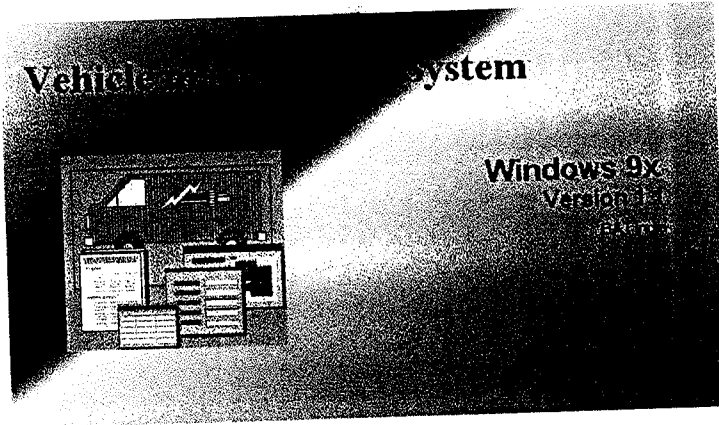
APPENDIX

APPENDIX-A

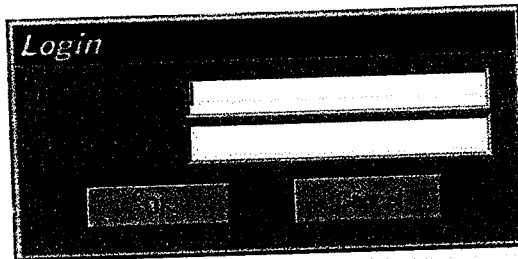
APPENDIX - A

Visual Basic User Interfaces

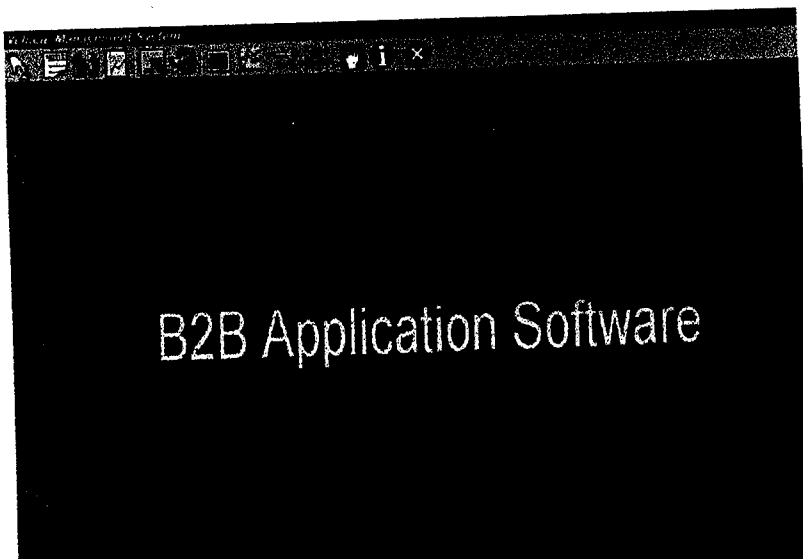
Splash Screen



Login Sc



Main Screen



History of Vehicle

Vehicle History

HIG 67 F 3343	04/07/2000
BEMC 30 Loader	Arad
2000	25
K 13345	03/04/2001
L 9753	03/07/2001

Oil, Oil filter

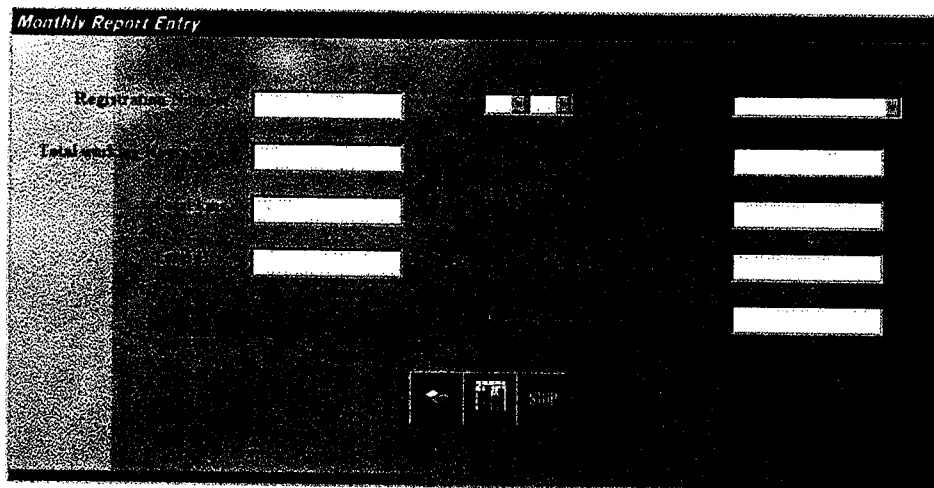
Silencer

Daily Report Entry

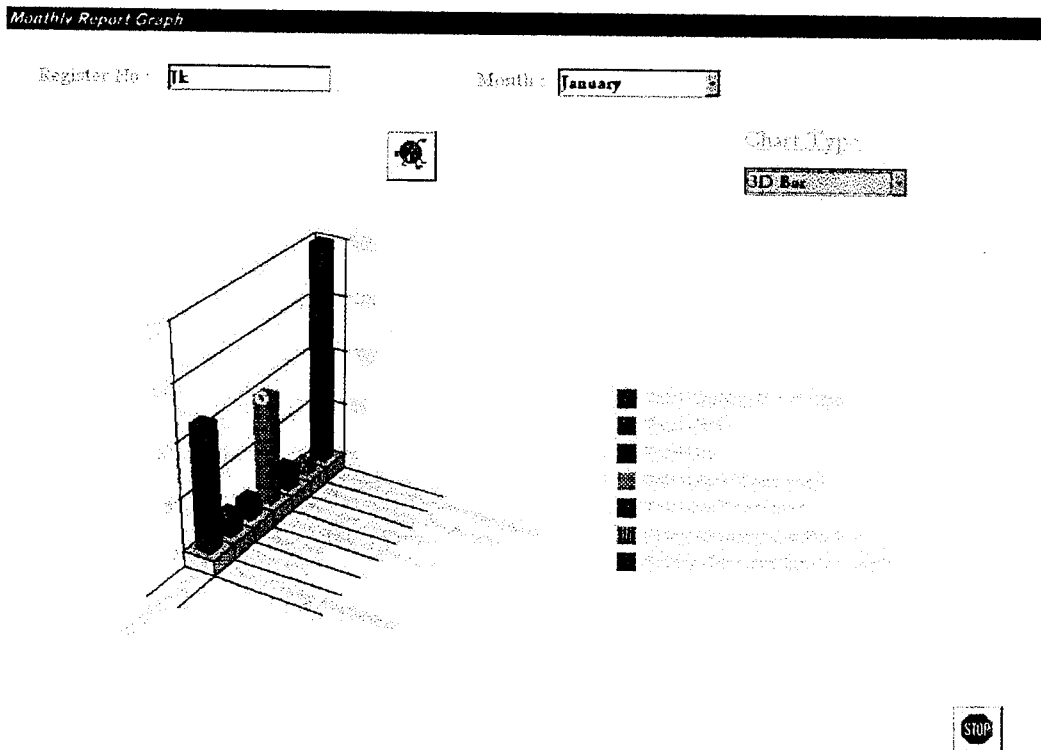
Daily Report Entry

Date :		Site Name :	
Registration Number :		20-40 Engine Oils(Litres) :	
Starting Hrs/Km :		60-95-100-140 Grade Oils :	
Closing Hrs/Km :		Grease :	
Working Hrs/Km :		Old Oil :	
Total Billing Hrs/Kms :		Idle Hours :	
Diesel Quantity (Litres) :			
Spare Parts Changed			
Remarks :			

Monthly Report Entry



Monthly Report Graph



Daily Items to be changed

A screenshot of a software window titled "Daily Items to be changed". The window contains a vertical list of five empty rectangular input fields, likely for entering daily items to be changed.

Default Spare part reminders

A screenshot of a software window titled "Default Spare part reminders". The window contains several labels and corresponding input fields:

- Register No. : [input field]
- Date : [input field]
- Starting Km : [input field]
- Closing Km : [input field]
- Spare part name : [input field]
- Spare part code : [input field]

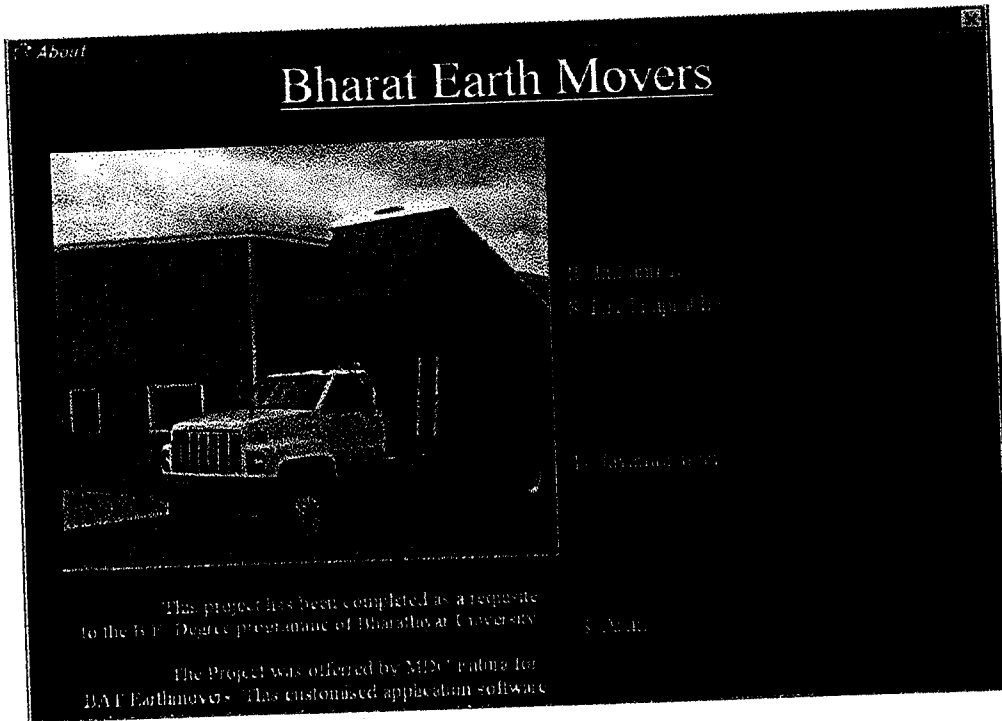
At the bottom right of the window, there is a circular button with a star icon.

User Management Screen

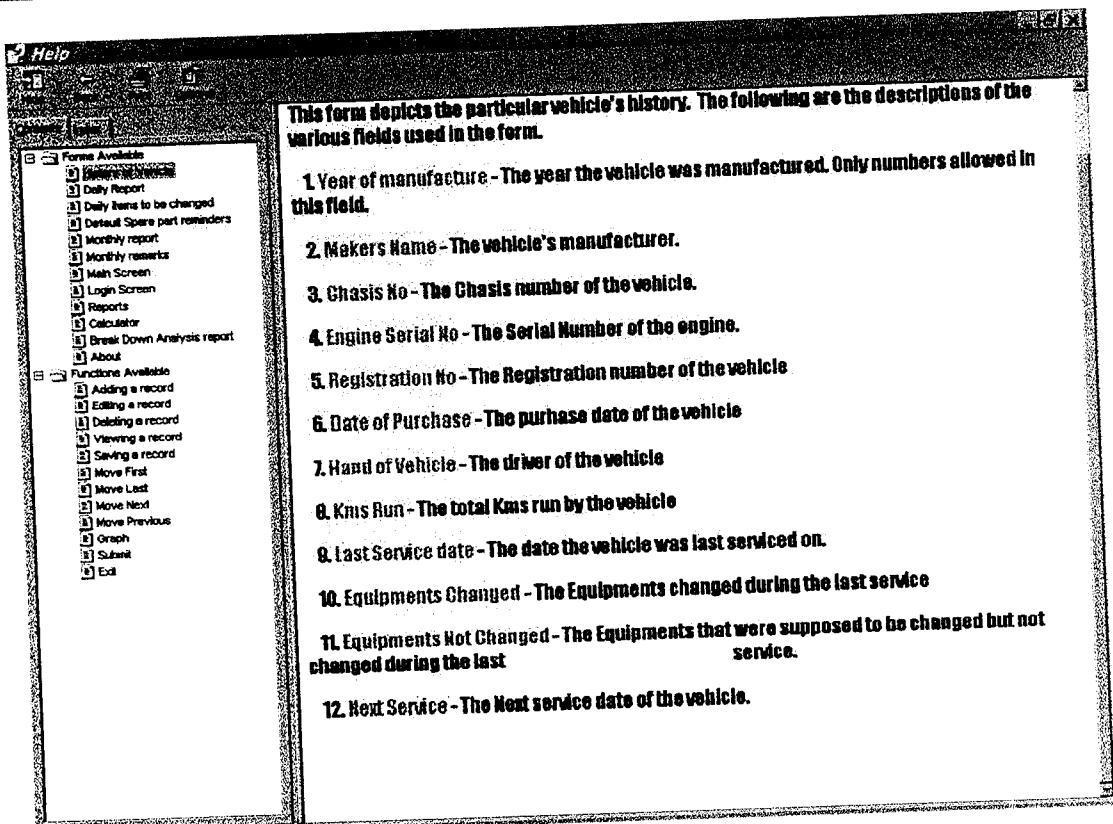
A screenshot of a software window titled "User management". The window is divided into two main sections:

- User Details:** Contains three input fields: "User ID" (with the value "jkannan"), "Password" (with asterisks), and "Confirm Password" (with asterisks). An "Exit" button is located to the right of the "User ID" field.
- User rights:** Contains three radio button options: "Entry level", "Report level", and "Modification level".

Logout Screen



Help Interface



APPENDIX-B

APPENDIX – B

Visual basic 6.0 coding used in forms

I. Entry Level Interfaces

There are several forms using the same coding listed here, the only thing that changes is the number of text boxes. These forms form the source of data entry.

Option Explicit

Dim cnn As New ADODB.Connection, rst As New ADODB.Recordset, cmd As ADODB.Command, rst2 As New ADODB.Recordset

Dim str As String, strsql As String, strsql1 As String

Dim Msg, Style, Title, Help, Ctxt, Response, MyString

Public tinsI As Variant, sasIII As Variant

Dim entry As Integer, report As Integer, modi As Integer

Private Sub Form_Load()

str = App.Path & "\vehicle.mdb"

strsql = "select * from dailyitems"

strsql1 = "select * from passwd"

cnn.Provider = "Microsoft.Jet.OLEDB.3.51"

cnn.Open str, "admin"

rst.Open strsql, cnn, adOpenKeyset, adLockOptimistic

rst2.Open strsql1, cnn, adOpenKeyset, adLockOptimistic

rst2.MoveFirst

Do Until (rst2(0) = sasIII)

rst2.MoveNext

Loop

entry = rst2(2)

report = rst2(3)

modi = rst2(4)

Text1.Enabled = False

Text2.Enabled = False

Text3.Enabled = False

Text4.Enabled = False

Text5.Enabled = False

If entry <> 0 Then

Toolbar1.Buttons(2).Enabled = False

Toolbar1.Buttons(3).Enabled = False

Toolbar1.Buttons(4).Enabled = False

Toolbar1.Buttons(7).Enabled = False

Toolbar1.Buttons(8).Enabled = False

```

mndel.Enabled = False
mncedit.Enabled = False
mncfirst.Enabled = False
mnclast.Enabled = False
mncnext.Enabled = False
mncpre.Enabled = False
mncview.Enabled = False

```

```
End If
```

```
If report <> 0 Then
```

```

Toolbar1.Buttons(2).Enabled = False
Toolbar1.Buttons(3).Enabled = False
Toolbar1.Buttons(4).Enabled = False
mndel.Enabled = False
mncedit.Enabled = False
mncview.Enabled = False

```

```
End If
```

```

Toolbar1.Buttons(5).Enabled = False
mncsave.Enabled = False
End Sub

```

```
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```

If Button = vbRightButton Then
dailyitems.PopupMenu mncclk
End If

```

```
End Sub
```

```
Private Sub mncadd_Click()
```

```

Text1.Enabled = True
Text2.Enabled = True
Text3.Enabled = True
Text4.Enabled = True
Text5.Enabled = True

```

```

Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""

```

```

Text1.SetFocus
rst.AddNew

```

```
End Sub
```

```
Private Sub mndel_Click()
```

```

Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False

```

```
nextv2:
```

```
tins1 = InputBox("Enter Vehicle register no", "Daily Items to be changed")
```

```
If Len(tins1) = 0 Then Exit Sub
```

```
rst.MoveFirst
```

```
Do While tins1 <> rst(0)
```

```

rst.MoveNext
If rst.EOF Then
rst.MoveLast
GoTo nextv2
End If
Loop
    Text1.Text = rst(0)
    Text2.Text = rst(1)
    Text3.Text = rst(2)
    Text4.Text = rst(3)
    Text5.Text = rst(4)

    Msg = "Confirm Delete..."
    Style = vbYesNo + vbQuestion + vbDefaultButton1
    Title = "Daily items"
    Response = MsgBox(Msg, Style, Title, Help, Ctxt)
    If Response = vbYes Then
rst.Delete
rst.MoveNext
If rst.EOF Then rst.MoveLast
Text1.Text = rst(0)
    Text2.Text = rst(1)
    Text3.Text = rst(2)
    Text4.Text = rst(3)
    Text5.Text = rst(4)

```

End If

End Sub

```

Private Sub mncedit_Click()
Text1.Enabled = True
Text2.Enabled = True
Text3.Enabled = True
Text4.Enabled = True
Text5.Enabled = True

```

```

tinsI = InputBox("Enter Vehicle register no", "Daily items")
rst.MoveFirst

```

```

If Len(tinsI) = 0 Then
Exit Sub
Toolbar1.Buttons(5).Enabled = False
mnsave.Enabled = False
Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False

```

End If

```

Do While tinsI <> rst(0)
rst.MoveNext
If rst.EOF Then
rst.MoveLast
Dim r As Variant
r = MsgBox("Vehicle not found, Try again", , Title)
Exit Sub

```

End If

Loop

```
Text1.Text = rst(0)
Text2.Text = rst(1)
Text3.Text = rst(2)
Text4.Text = rst(3)
Text5.Text = rst(4)
```

```
Toolbar1.Buttons(5).Enabled = True
mnsave.Enabled = True
End Sub
```

```
Private Sub mnexit_Click()
    cnn.Close
```

```
Unload Me
End Sub
```

```
Private Sub mnfirst_Click()
    rst.MoveFirst
    Text1.Text = rst(0)
    Text2.Text = rst(1)
    Text3.Text = rst(2)
    Text4.Text = rst(3)
    Text5.Text = rst(4)
```

End Sub

```
Private Sub mnlast_Click()
    rst.MoveLast
    Text1.Text = rst(0)
    Text2.Text = rst(1)
    Text3.Text = rst(2)
    Text4.Text = rst(3)
    Text5.Text = rst(4)
```

End Sub

```
Private Sub mnnext_Click()
    rst.MoveNext
    If rst.EOF Then rst.MoveLast
    Text1.Text = rst(0)
    Text2.Text = rst(1)
    Text3.Text = rst(2)
    Text4.Text = rst(3)
    Text5.Text = rst(4)
```

End Sub

```
Private Sub mnpre_Click()
    rst.MovePrevious
    If rst.BOF Then rst.MoveFirst
```

```
Text1.Text = rst(0)
Text2.Text = rst(1)
```

nd Sub

```

Private Sub mnsave_Click()
On Error Resume Next

Msg = "Update database"
Style = vbYesNo + vbQuestion + vbDefaultButton1
Title = "Daily items"
Response = MsgBox(Msg, Style, Title, Help, Ctxt)

```

```

If Response = vbYes Then
If Len(Text1.Text) <> 0 Then
rst(0) = Text1.Text

```

```

If Text2.Text <> Null Or Text2.Text <> "" Then
rst(1) = Text2.Text
Else
rst(1) = "Not Available"
End If

```

```

If Text3.Text <> Null Or Text3.Text <> "" Then
rst(2) = Text3.Text
Else
rst(2) = "Not Available"
End If

```

```

If Text4.Text <> Null Or Text4.Text <> "" Then
rst(3) = Text4.Text
Else
rst(3) = "Not Available"
End If

```

```

If Text5.Text <> Null Or Text5.Text <> "" Then
rst(4) = Text5.Text
Else
rst(4) = "Not Available"
End If

```

```

Else
MsgBox ("Reg No required")
Text1.SetFocus
GoTo ending11
End If

```

```

rst.Update
Toolbar1.Buttons(5).Enabled = False
mnsave.Enabled = False

```

```

Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False

```

```

Else
Text1.SetFocus

```


ending11:

End Sub

Private Sub mnview_Click()

Text1.Enabled = False

Text2.Enabled = False

Text3.Enabled = False

Text4.Enabled = False

Text5.Enabled = False

nextv1:

tinsl = InputBox("Enter Vehicle register no", "Daily items")

If Len(tinsl) = 0 Then Exit Sub

rst.MoveFirst

Do While tinsl <> rst(0)

rst.MoveNext

If rst.EOF Then

rst.MoveLast

GoTo nextv1

End If

Loop

Text1.Text = rst(0)

Text2.Text = rst(1)

Text3.Text = rst(2)

Text4.Text = rst(3)

Text5.Text = rst(4)

End Sub

Private Sub Text5_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = 13 Then

Msg = "Update database"

Style = vbYesNo + vbQuestion + vbDefaultButton1

Title = "Daily items"

Response = MsgBox(Msg, Style, Title, Help, Ctxt)

If Response = vbYes Then

If Len(Text1.Text) <> 0 Then

rst(0) = Text1.Text

If Text2.Text <> Null Or Text2.Text <> "" Then

rst(1) = Text2.Text

Else

rst(1) = "Not Available"

End If

If Text3.Text <> Null Or Text3.Text <> "" Then

rst(2) = Text3.Text

Else

rst(2) = "Not Available"

End If

If Text4.Text <> Null Or Text4.Text <> "" Then

```

If Text5.Text <> Null Or Text5.Text <> "" Then
    rst(4) = Text5.Text
Else
    rst(4) = "Not Available"
End If

```

```

Else
    MsgBox ("Reg no required")
    Text1.SetFocus
    GoTo ending2
End If

```

```

rst.Update
Msg = "More..."
Style = vbYesNo + vbQuestion + vbDefaultButton1
Title = "Daily items"
Response = MsgBox(Msg, Style, Title, Help, Ctxt)

```

```

If Response = vbYes Then
    Text1.Text = ""
    Text2.Text = ""
    Text3.Text = ""
    Text4.Text = ""
    Text5.Text = ""

```

```

    Text1.SetFocus
    rst.AddNew
ElseIf Response = vbNo Then
    cnn.Close
    Unload Me
End If

```

```

Else
    Text1.SetFocus
End If

```

```

ending2:
End If
End Sub

```

```

Private Sub Text2_KeyPress(KeyAscii As Integer)
If Not (KeyAscii >= 47 And KeyAscii <= 57) And Not (KeyAscii = 8) Then
KeyAscii = 0
End If
End Sub

```

```

Private Sub Toolbar1_ButtonClick(ByVal Button As MSComctlLib.Button)

```

```

Select Case Button.Index
Case 1

```

```

Text1.Enabled = True
Text2.Enabled = True
Text3.Enabled = True
Text4.Enabled = True

```

```
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
```

```
Text1.SetFocus
rst.AddNew
```

Case 2

```
Text1.Enabled = True
Text2.Enabled = True
Text3.Enabled = True
Text4.Enabled = True
Text5.Enabled = True
```

```
tinsl = InputBox("Enter Vehicle register no", "Daily items")
rst.MoveFirst
```

```
If Len(tinsl) = 0 Then
Exit Sub
Toolbar1.Buttons(5).Enabled = False
mnsave.Enabled = False
```

```
Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False
```

```
End If
```

```
Do While tinsl <> rst(0)
rst.MoveNext
If rst.EOF Then
rst.MoveLast
Dim r As Variant
r = MsgBox("Vehicle not found, Try again", , Title)
Exit Sub
```

```
End If
```

```
Loop
```

```
Text1.Text = rst(4)
Text2.Text = rst(1)
Text3.Text = rst(2)
Text4.Text = rst(0)
Text5.Text = rst(3)
```

```
Toolbar1.Buttons(5).Enabled = True
```

Case 3

```

nextv1:
tinsI = InputBox("Enter Vehicle register no", "Daily items")
If Len(tinsI) = 0 Then Exit Sub
rst.MoveFirst
Do While tinsI <> rst(0)
rst.MoveNext
If rst.EOF Then
rst.MoveLast
GoTo nextv1
End If
Loop

```

```

Text1.Text = rst(4)
Text2.Text = rst(1)
Text3.Text = rst(2)
Text4.Text = rst(0)
Text5.Text = rst(3)

```

```

Case 4
Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False

```

```

nextv2:
tinsI = InputBox("Enter Vehicle register no", "Daily items")
If Len(tinsI) = 0 Then Exit Sub
rst.MoveFirst
Do While tinsI <> rst(0)
rst.MoveNext
If rst.EOF Then
rst.MoveLast
GoTo nextv2
End If
Loop

```

```

Text1.Text = rst(4)
Text2.Text = rst(1)
Text3.Text = rst(2)
Text4.Text = rst(0)
Text5.Text = rst(3)

```

```

Msg = "Confirm Delete..."
Style = vbYesNo + vbQuestion + vbDefaultButton1
Title = "Daily Items to be changed"
Response = MsgBox(Msg, Style, Title, Help, Ctxt)
If Response = vbYes Then
rst.Delete
rst.MoveNext
If rst.EOF Then rst.MoveLast
Text1.Text = rst(0)
Text2.Text = rst(1)
Text3.Text = rst(2)
Text4.Text = rst(3)
Text5.Text = rst(4)

```

```

End If

```

On Error Resume Next

```
Msg = "Update database"
Style = vbYesNo + vbQuestion + vbDefaultButton1
Title = "Daily items"
Response = MsgBox(Msg, Style, Title, Help, Ctxt)
```

```
If Response = vbYes Then
If Len(Text1.Text) <> 0 Then
```

```
rst(0) = Text1.Text
```

```
If Text2.Text <> Null Or Text2.Text <> "" Then
```

```
rst(1) = Text2.Text
```

```
Else
```

```
rst(1) = "Not Available"
```

```
End If
```

```
If Text3.Text <> Null Or Text3.Text <> "" Then
```

```
rst(2) = Text3.Text
```

```
Else
```

```
rst(2) = "Not Available"
```

```
End If
```

```
If Text4.Text <> Null Or Text4.Text <> "" Then
```

```
rst(3) = Text4.Text
```

```
Else
```

```
rst(3) = "Not Available"
```

```
End If
```

```
If Text5.Text <> Null Or Text5.Text <> "" Then
```

```
rst(4) = Text5.Text
```

```
Else
```

```
rst(4) = "Not Available"
```

```
End If
```

```
Else
```

```
Text1.SetFocus
```

```
GoTo ending1
```

```
End If
```

```
rst.Update
```

```
Toolbar1.Buttons(5).Enabled = False
```

```
mnsave.Enabled = False
```

```
Text1.Enabled = False
```

```
Text2.Enabled = False
```

```
Text3.Enabled = False
```

```
Text4.Enabled = False
```

```
Text5.Enabled = False
```

```
Else
```

```
Text1.SetFocus
```

```
End If
```

```
ending1:
```

```
Text2.Text = rst(1)
Text3.Text = rst(2)
Text4.Text = rst(3)
Text5.Text = rst(4)
```

Case 8

```
rst.MoveLast
Text1.Text = rst(0)
Text2.Text = rst(1)
Text3.Text = rst(2)
Text4.Text = rst(3)
Text5.Text = rst(4)
```

Case 9

```
rst.MovePrevious
If rst.BOF Then rst.MoveFirst
Text1.Text = rst(0)
Text2.Text = rst(1)
Text3.Text = rst(2)
Text4.Text = rst(3)
Text5.Text = rst(4)
```

Case 10

```
rst.MoveNext
If rst.EOF Then rst.MoveLast
Text1.Text = rst(0)
Text2.Text = rst(1)
Text3.Text = rst(2)
Text4.Text = rst(3)
Text5.Text = rst(4)
```

Case 12

```
cnn.Close
Unload Me
End Select
End Sub
```

2. User Management coding

The addition of users is similar to the addition of vehicles. The only caution is that the password should be confirmed and the user can assign only one level of administration.

3. Graph Generator coding

```
Option Explicit
Dim cnn As New ADODB.Connection, rst As New ADODB.Recordset, cmd As ADODB.Command, rst2 As
New ADODB.Recordset
Dim str As String, strsql As String, strsql1 As String
```

```

Private Sub Combo2_Click()
If Combo2.Text = "Bar" Then
MSChart1.chartType = VtChChartType2dBar
ElseIf Combo2.Text = "3D Bar" Then
MSChart1.chartType = VtChChartType3dBar
Else
MSChart1.chartType = VtChChartType2dPie
End If
End Sub

```

```

Private Sub Command1_Click()
rst.MoveFirst
Do While Text1.Text <> rst(0)
rst.MoveNext
If rst.EOF Then
rst.MoveLast
MsgBox ("Vehicle not found, Enter Again")
Text1.Text = ""
Text1.SetFocus
Exit Sub
End If
Loop

```

```

rst2.Open "select * from monthlyreport where regno= " & "" & Text1.Text & " and month=" & "" &
Combo1.Text & "", cnn
If rst2.EOF = True Then
MsgBox "Required month not available"
Combo1.SetFocus
rst2.Close
Exit Sub
End If

```

```

For i = 1 To 7
MSChart1.Column = i
MSChart1.Data = rst2(i)
Next i
rst2.Close
End Sub

```

```

Private Sub Command2_Click()
cnn.Close
Unload Me
End Sub

```

```

Private Sub Form_Load()
Command1.Picture = ImageList1.ListImages(1).Picture
Command2.Picture = ImageList1.ListImages(3).Picture
str = App.Path & "\vehicle.mdb"
strsql = "select * from monthlyreport"

```

```

cnn.Provider = "Microsoft.Jet.OLEDB.3.51"
cnn.Open str, "admin"
rst.Open strsql, cnn, adOpenKeyset, adLockOptimistic

```

```

End Sub

```

4. Sample codes for Report Generators

```

Option Explicit
Dim cnn As New ADODB.Connection, rst As New ADODB.Recordset, cmd As ADODB.Command, rst2 As
New ADODB.Recordset
Dim str As String, strsql As String, strsql1 As String
Dim Msg, Style, Title, Help, Ctxt, Response, MyString
Public tins1 As Variant, sas11 As Variant
Dim entry As Integer, report As Integer, modi As Integer

Private Sub Command2_Click()
DataEnvironment1.rsCommand2.Open "select * from dailyreport where regno=" & "" & Text1.Text & "", cnn,
adOpenDynamic, adLockOptimistic
rptdaily.Show

End Sub

Private Sub Command3_Click()
DataEnvironment1.rsCommand3.Open "select * from monthlyreport where regno=" & "" & Text1.Text & "",
cnn, adOpenDynamic, adLockOptimistic
rptmonthly.Show

End Sub

Private Sub Command5_Click()
DataEnvironment1.rsCommand4.Open "select * from sparepart where regno=" & "" & Text1.Text & "", cnn,
adOpenDynamic, adLockOptimistic
rptsparepart.Show

End Sub

Private Sub Command6_Click()
DataEnvironment1.rsCommand5.Open "select * from history where reg_no= " & "" & Text1.Text & "", cnn,
adOpenDynamic, adLockOptimistic
sparepartschanged.Show

End Sub

Private Sub Command7_Click()
DataEnvironment1.rsCommand5.Open "select * from dailyreport where regno=" & "" & Text1.Text & "", cnn,
adOpenDynamic, adLockOptimistic
Sparepartschangeddaily.Show

End Sub

Private Sub Command9_Click()
cnn.Close
Unload Me
End Sub

Private Sub Form_Load()
str = App.Path & "\vehicle.mdb"

```



```
cnn.Open (str)
Command1.Picture = ImageList2.ListImages(2).Picture
Command2.Picture = ImageList2.ListImages(9).Picture
Command3.Picture = ImageList2.ListImages(10).Picture
Command4.Picture = ImageList2.ListImages(12).Picture
Command5.Picture = ImageList2.ListImages(15).Picture
Command6.Picture = ImageList2.ListImages(5).Picture
Command7.Picture = ImageList2.ListImages(13).Picture
Command8.Picture = ImageList2.ListImages(14).Picture
Command9.Picture = ImageList2.ListImages(1).Picture
End Sub
```

```
Private Sub Command1_Click()
DataEnvironment1.rsCommand1.Open "select * from History where reg_no=" & "" & Text1.Text & "", cnn,
adOpenDynamic, adLockOptimistic
rpthistory.Show
```

```
End Sub
```

APPENDIX - C

APPENDIX - C

HTML and ASP user interfaces

Administrator Screen

- ✱ [History](#)
- ✱ [Daily Report Entry](#)
- ✱ [Monthly Report Entry](#)
- ✱ [Remarks of the Month](#)
- ✱ [Default spare part reminders](#)
- ✱ [Daily items to be changed](#)
- ✱ [Other Reports](#)
- ✱ [Break down analysis report](#)
- ✱ [Administrator](#)

Administrator Area

Restricted Entry

[Members](#)

[Vehicles](#)

Default Spare part Reminders

- ✱ [History](#)
- ✱ [Daily Report Entry](#)
- ✱ [Monthly Report Entry](#)
- ✱ [Remarks of the Month](#)
- ✱ [Default spare part reminders](#)
- ✱ [Daily items to be changed](#)
- ✱ [Other Reports](#)
- ✱ [Break down analysis report](#)

DEFAULT SPARE PART REMINDERS

=====

Vehicle ID -

Starting km-

Changing km-

Spare part name-

Adding Vehicles to the Database

<ul style="list-style-type: none"> * History * Daily Report Entry * Monthly Report Entry * Remarks of the Month * Default spare part remainders * Daily items to be changed * Other Reports * Break down analysis report * Administrator 	<h2>Vehicle Details-Add</h2> <p style="text-align: center;">Add New Vehicle</p> <hr style="border-top: 1px dashed black;"/> <p style="text-align: center;">Vehicle ID :- <input style="width: 150px;" type="text"/></p> <hr style="border-top: 1px dashed black;"/> <p>Year Of Manufacture :- <input style="width: 100px;" type="text"/></p> <p>Makers Name :- <input style="width: 100px;" type="text"/></p> <p>Chasis No :- <input style="width: 100px;" type="text"/></p> <p>Registration Number :- <input style="width: 100px;" type="text"/></p> <p>Eng.Sr.No :- <input style="width: 100px;" type="text"/></p> <p>Date Of Purchase :- <input style="width: 100px;" type="text"/></p>
---	---

Daily Report Entry

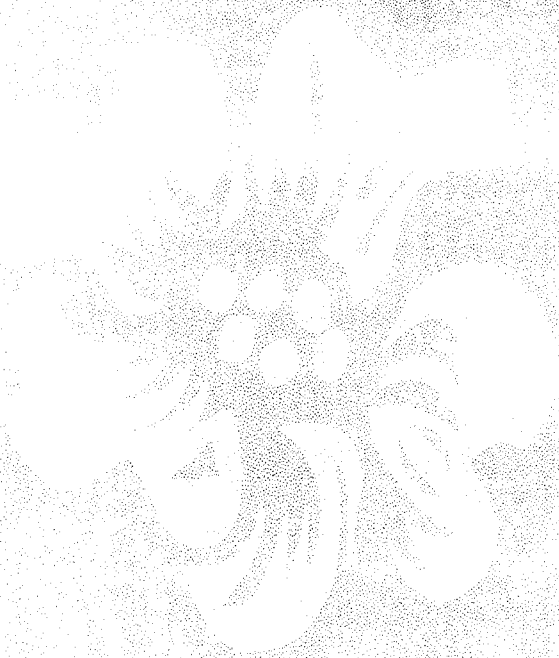
<ul style="list-style-type: none"> * History * Daily Report Entry * Monthly Report Entry * Remarks of the Month * Default spare part remainders * Daily items to be changed * Other Reports * Break down analysis report * Administrator 	<h2>Daily Report Entry</h2> <hr style="border-top: 1px dashed black;"/> <p><input style="width: 100px;" type="text"/> --Vehicle ID</p> <p><input style="width: 100px;" type="text"/> --Date</p> <p><input style="width: 100px;" type="text"/> --Site Name</p> <p><input style="width: 100px;" type="text"/> --Starting hmr/km</p> <p><input style="width: 100px;" type="text"/> --Closing hmr/km</p> <p><input style="width: 100px;" type="text"/> --Working hmr/km</p> <p><input style="width: 100px;" type="text"/> --Total billing hrs/kms</p> <p><input style="width: 100px;" type="text"/> --Diesel Quantity(ltrs)</p> <p><input style="width: 100px;" type="text"/> --20-40 Engine Oils</p> <p><input style="width: 100px;" type="text"/> --60-98-100-140 Grace Oils</p>
---	--

Daily Items to be Changed

DAILY ITEMS TO BE CHANGED	
<ul style="list-style-type: none">* History* Daily Report Entry* Monthly Report Entry* Remarks of the Month* Default spars part remainders* Daily items to be changed* Other Reports* Break down analysis report* Administrator	<p>Vehicle ID- <input type="text"/></p> <p>Oil- <input type="text"/></p> <p>Oil Filter- <input type="text"/></p> <p>Diesel Filter- <input type="text"/></p>

Adding new Members

Members Details-Add	
<ul style="list-style-type: none">* History* Daily Report Entry* Monthly Report Entry* Remarks of the Month* Default spars part remainders* Daily items to be changed* Other Reports* Break down analysis report* Administrator	<p>Details Regarding Website Accessing :-</p> <p>Member ID :- <input type="text"/></p> <p>Member's UserID :- <input type="text"/></p> <p>Member's Password :- <input type="text"/></p> <p>Members Priority :- <input checked="" type="radio"/> Entry Level Access <input type="radio"/> Report Level-Partial Access</p> <p>Personal Details :-</p> <p>Member Name :- <input type="text"/></p> <p>Age :- <input type="text"/></p> <p>Sex :- <input checked="" type="radio"/> Male <input type="radio"/> Female</p> <p>Marital Status :- <input type="text"/></p>



APPENDIX-D

APPENDIX - D

HTML and ASP code

Vehicles - Add

```

<html>

<head>

<title>History of the Vehicle</title>

</head>
<body background="_themes/expeditn/exptextb.jpg" bgcolor="#FFFFFF" text="#000000"
link="#993300" vlink="#666600" alink="#CC3300"><!--mstheme--><font face="Book Antiqua,
Times New Roman, Times">

<p align="center"><b><blink><font face="Transit511 BT" size="5" color="#00CCFF">History
Of the Vehicle</font></blink></b></p>

<p><b><font face="Times New Roman" size="5" color="#800000"><blink>
<marquee behavior="slide">Add New Vehicle</marquee>
</blink></font></b></p>

<form method="post" action="history_add.asp">

<p><input type="text" name="T1" size="20">--<font face="Times New Roman"><b>Vehicle
ID</b></font></p>
<p align="center"></p>
<p><font face="Times New Roman" size="3"><b><input type="text" name="T2" size="20">--
Year Of Manufacture</b></font></p>
<p><font face="Times New Roman" size="3"><b><input type="text" name="T3" size="20">--
Makers
Name</b></font></p>
<p><font face="Times New Roman" size="3"><b><input type="text" name="T4" size="20">--
Chassis
Number</b></font></p>
<p><font face="Times New Roman" size="3"><b><input type="text" name="T5" size="20">--
Engine
Serial Number</b></font></p>
<p><font face="Times New Roman" size="3"><b><input type="text" name="T6" size="20">--
Registration
Number</b></font></p>
<p><font face="Times New Roman" size="3"><b><input type="text" name="T7" size="20">--
Date

```



```

T7=TRIM(request("T7"))
T8=TRIM(request("T8"))
T9=TRIM(request("T9"))
T10=TRIM(request("T10"))
T11=TRIM(request("T11"))
T12=TRIM(request("T12"))
T13=TRIM(request("T13"))

%>
<html>
<head><title>Add Products</title></head>
<body bgcolor="gray">

<%

set con=server.createobject("ADODB.Connection")
con.open "accessDSN1"
sqlstring="insert into history values( " &_
T1 & ", " &_
T2 & ", " &_
" "" & T3 & "" , " &_
T4 & ", " &_
" "" & T5 & "" , " &_
" "" & T6 & "" , " &_
" "" & T7 & "" , " &_
T8 & ", " &_
T9 & ", " &_
" "" & T10 & "" , " &_
" "" & T11 & "" , " &_
" "" & T12 & "" , " &_
" "" & T13 & "" )"
con.execute sqlstring
con.close
%>
<center>
<table width="600" cellpadding="4" cellspacing="0" bgcolor="lightyellow">
<tr>
<td>
<%=T1%>was added to the database
</td>
</tr>
</table>
</center>
<p>
<a href="123.asp">Back</a>
</body>
</html>

```

Administrator – Modify

```

<html>
<head>
<title>Vehicle Details</title>
</head>
<body background="_themes/expeditn/exptextb.jpg" bgcolor="#FFFFFF" text="#000000"
link="#993300" vlink="#666600" alink="#CC3300"><!--mstheme--><font face="Book Antiqua,
Times New Roman, Times">
<h1 align="center">Vehicle Details</h1>
<h3 align="center"><font face="Times New Roman" size="4"
color="#008080"><b>Modification</b></font></h3>
<hr align="center">
<form method="POST" action="adm_vehicles_modify_1.asp">
  <p align="center">&nbsp;</p>
  <p align="center">Vehicle ID :-<input type="text" name="T1" size="20"><input
type="submit" value="Get Details" name="B3"></p>
  <hr align="center">
</form>
</body>
</html>

```

adm_vehicles_modify_1.asp

```

<%
t1=request("T1")
set Con=server.createObject("ADODB.Connection")
Con.Open "accessDSN1"

set rs=server.CreateObject("ADODB.Connection")

'rs.ActiveConnection = Con

'rs.CursorType=3

sqlstring= "select * from history where vehicle_id=" &t1

```



```

t7=TRIM(request("t7"))
t8=TRIM(request("t8"))
t9=TRIM(request("t9"))
t10=TRIM(request("t10"))
t11=TRIM(request("t11"))
t12=TRIM(request("t12"))
t13=TRIM(request("t13"))

```

```

sqlstring="update history set " & _
"vehicle_id=" & t1 & "," & _
"year_of_manufacture=" & t2 & "," & _
"makers_name=" & t3 & "," & _
"chasis_no=" & t4 & "," & _
"eng_sr_no=" & t5 & "," & _
"reg_no=" & t6 & "," & _
"date_of_purchase=" & t7 & "," & _
"hand_of_vehicle=" & t8 & "," & _
"kms_run_during_purchase=" & t9 & "," & _
"last_service_details=" & t10 & "," & _
"equipment_changed=" & t11 & "," & _
"equipment_not_changed=" & t12 & "," & _
"next_service_date=" & t13 & "" & _
"where vehicle_id=" & t1

```

```

con.execute sqlstring
con.close
%>

```

```

<html>
<head><title>Add Products</title></head>
<body bgcolor="gray">

<center>
<table width="600" cellpadding="4" cellspacing="0" bgcolor="lightyellow">
<tr>
<td>
<%=T1%> was updated in the database
</td>
</tr>
</table>
</center>
<p>
<a href="modify_vehicles.asp">Back</a>
</body>
</html>

```

Administrator - Delete

```
<html>
```

```

<head>
<title>Vehicle Details</title>
</head>
<body background="_themes/expeditn/exptextb.jpg" bgcolor="#FFFFFF" text="#000000"
link="#993300" vlink="#666600" alink="#CC3300"><!--mstheme--><font face="Book Antiqua,
Times New Roman, Times">
<h1 align="center"><!--mstheme--><font color="#660033">Vehicle Details<!--mstheme--
></font></h1>
<h3 align="center"><!--mstheme--><font color="#660033"><font face="Times New Roman"
size="4" color="#008080"><b>Deletion</b></font><!--mstheme--></font></h3>
<!--msthemeseparator--><p align="center"></p>
<form method="POST" action="adm_vehicle_delete.asp">
  <p align="center">&nbsp;</p>
  <p align="center">Vehicle ID :-<input type="text" name="T1" size="20"></p>
  <!--msthemeseparator--><p align="center"></p>
  <p>&nbsp;</p>
  <p align="center"><input type="submit" value="Delete" name="B4"><input type="reset"
value="Reset" name="B5"></p>
</form>
<!--mstheme--></font></body>
</html>

```

adm_vehicle_delete.asp

```
<%
```

```
t1=request("T1")
```

```
set con=server.createobject("ADODB.Connection")
con.open "accessDSN1"
```

```
sqlstring="delete * from history where vehicle_id=" & t1
```

```
con.execute sqlstring
%>
```

```

<html>
<head><title>Delete Vehicle !!!</title></head>
<body bgcolor="gray">
<center>
<table width="600" cellpadding="4" cellspacing="0" bgcolor="lightyellow">

```

```

<td>
<%=T1%> vehicle was removed to the database
</td>
</tr>
</table>
</center>
<p>
<a href="delete_vehicles.asp">Back</a>
</body>
</html>

```

History reports

```

<html>
<head>
<title>HISTORY REPORT</title>
</head>
<body background="_themes/indust/indtextb.jpg" bgcolor="#FFFFFF" text="#000000"
link="#3366CC" vlink="#666666" alink="#996600"><!--mstheme--><font face="Trebuchet MS,
Arial, Helvetica">
<!--msthemeseparator--><p align="center"></p>
<h2 align="center"><!--mstheme--><font color="#333399">REPORTS<!--mstheme--
></font></h2>
<!--msthemeseparator--><p align="center"></p>
<p>&nbsp;</p>
<h3 align="center"><!--mstheme--><font color="#666666">History:<!--mstheme--
></font></h3>
<p>&nbsp;</p>
<form method="POST" action="history_report_1.asp">
  <!--msthemeseparator--><p align="center"></p>
  <p align="center">&nbsp;</p>
  <p align="center"><input type="submit" value="Generate" name="B1"><input type="reset"
value="Reset" name="B2"></p>
</form>
<p align="center">&nbsp;</p>
<p align="center">&nbsp;</p>
<p>&nbsp;</p>
<!--mstheme--></font></body>
</html>

```



```

<p align="center">&nbsp;</p>
<!--mstheme--></font><!--msthemelist--><table border="0" cellpadding="0" cellspacing="0"
width="100%">
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="HISTOR~1.HTM">History</a><!--mstheme--></font><!--
msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="DAILY~1.HTM">Daily Report Entry</a><!--mstheme--
></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="MONTHL~1.HTM">Monthly Report Entry</a><!--mstheme--
></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="REMARK~1.HTM">Remarks of the Month</a><!--mstheme--
></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="DEFAULT~1.HTM">Default spare
part remainders</a><!--mstheme--></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="DAILY~2.HTM">Daily items to be
changed</a><!--mstheme--></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="problem_1.asp">Problems in the
vehicle</a><!--mstheme--></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">

```


<!--mstheme-->

<hr>

<p align="center"> </p>

<!--mstheme--><!--msthemelist--><table border="0" cellpadding="0" cellspacing="0" width="100%">

<!--msthemelist--><tr><td valign="baseline" width="42"></td><td valign="top" width="100%"><!--mstheme-->

<p align="center">History<!--mstheme--><!--msthemelist--></td></tr>

<!--msthemelist--><tr><td valign="baseline" width="42"></td><td valign="top" width="100%"><!--mstheme-->

<p align="center">Daily Report Entry<!--mstheme--><!--msthemelist--></td></tr>

<!--msthemelist--><tr><td valign="baseline" width="42"></td><td valign="top" width="100%"><!--mstheme-->

<p align="center">Monthly Report Entry<!--mstheme--><!--msthemelist--></td></tr>

<!--msthemelist--><tr><td valign="baseline" width="42"></td><td valign="top" width="100%"><!--mstheme-->

<p align="center">Remarks of the Month<!--mstheme--><!--msthemelist--></td></tr>

<!--msthemelist--><tr><td valign="baseline" width="42"></td><td valign="top" width="100%"><!--mstheme-->

<p align="center">Default spare part remainders<!--mstheme--><!--msthemelist--></td></tr>

<!--msthemelist--><tr><td valign="baseline" width="42"></td><td valign="top" width="100%"><!--mstheme-->

<p align="center">Daily items to be changed<!--mstheme--><!--msthemelist--></td></tr>

<!--msthemelist--><tr><td valign="baseline" width="42"></td><td valign="top" width="100%"><!--mstheme-->

<p align="center">Other Reports<!--mstheme--><!--msthemelist--></td></tr>


```

<p align="center">&nbsp;</p>
<!--mstheme--></font><!--msthemelist--><table border="0" cellpadding="0" cellspacing="0"
width="100%">
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="HISTOR~1.HTM">History</a><!--mstheme--></font><!--
msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="DAILY_~1.HTM">Daily Report Entry</a><!--mstheme--
></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="MONTHL~1.HTM">Monthly Report Entry</a><!--mstheme--
></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="REMARK~1.HTM">Remarks of the Month</a><!--mstheme--
></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="DEFAULT~1.HTM">Default spare
part remainders</a><!--mstheme--></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="DAILY_~2.HTM">Daily items to be
changed</a><!--mstheme--></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <p align="center"><a href="problem_1.asp">Problems in the
vehicle</a><!--mstheme--></font><!--msthemelist--></td></tr>
  <!--msthemelist--><tr><td valign="baseline" width="42"></td><td
valign="top" width="100%"><!--mstheme--><font face="Book Antiqua, Times New Roman,
Times">
  <!--msthemeseparator--><p align="center"><p align="center"></p>
```

```
</html>
```

```
<%
```

```
end if  
end if  
end if
```

```
%>
```

Bibliography

1. Roger Jennings, *Database Developer's guide with Visual Basic 6.0*, Techmedia, 1999, 2nd Edition
2. Dan Appleman, *Developing COM/ActiveX Components*, Techmedia, 1999, 2nd Edition
3. Deborah S. Ray, Eric J. Ray, *Mastering HTML 4*, BPB Publications, 2000, 1st Edition
4. Stephen Walder, Jonathan Levine, *E-Com programming with ASP*, Techmedia, 2000, 1st Edition.
5. Wayne Freeze, *Visual Basic Database Programming Bible*, IDG Books, 2000, 1st Edition

Web Sites

1. www.microsoft.com
2. www.batindia.com
3. www.alltheweb.com
4. www.whatis.com
5. www.mcp.com
6. www.earthmoverssupply.com

