ANY TIME TAXI

**TABLE OF CONTENT**

|  |  |
| --- | --- |
| Contents | Page No. |
| 1. Title of the project. | 4 |
| 2. Introduction and objectives of the Project. | 5 |
| 3. Project Category | 6 |
| 4. Tools/Platform, Hardware and Software requirement specifications. | 7 |
| 5. Analysis | 9 |
| 6. Software Development Life cycle | 10 |
| 7. Data Flow Diagram and ER-Diagram | 14 |
| 8. ER-Diagram | 18 |
| 9. Module Description | 23 |
| 10.Report Generation | 24 |
| 11. System Analysis | 25 |
| 12. Security And Validation Checks | 29 |
| 13. Scope of future application | 31 |
| 14. Bibliography | 32 |

1. **TITLE OF THE PROJECT**

ANY TIME TAXI



***ECB***

**2. INTRODUCTION**

**Overview:**

Now one can easily plan the journey comfortably as the process is efficient and fast with being easy to access. Bookings can be made through the Easy Cab site or by the phone call. This being a big step in terms of improvement in the Taxi system it is widely accepted across the country.

* A route-based booking system that facilitates the issue of journey-cum-booking Taxi, which can be issued from any station to any station.
* Passenger journey to multiple laps of booking can be handled from a single terminal window.
* The booking facility is offered round-the-clock (24 hours uninterrupted).
* Changes in Taxi profiles (Taxi addition, replacement, de-allocation), route structures, etc., can be made effective immediately with the appropriate contingency handling.
* Dynamic definition of the advance booking period is possible. This feature facilitates defining different advance booking periods for different Taxis.
* Any Taxi running schedule can be accommodated.
* Provides on-line aggregation of EIS figures such as revenue, Taxi utilization, etc, and presentation of the summarized data in the form of visual analytics from the operational system's information store. The data aggregation is done incrementally, to inflict minimal impact.
* Provides automatic database recovery against all kinds of hardware and software failures.
* Complete audit trails for transactions and data access.
* The application software is parametric, and standard Taxi business rules are incorporated in the form of data instead of being part of the logic.

**Project Objective**

ECB is online web based system since it is connected through the internet all time one can easily access the ECB from anywhere. This is the flexibility that stands ECB in the top order from others.

**Objective:**

There are several main objectives for this project:

1. Easy Cab system is very helpful for user; user can search the taxi according to his/her needs and book the taxi.
2. But keeping track of all the activities and their records on paper is very cumbersome and error prone. It also is very inefficient and a time-consuming process Observing the continuous increase in population and number of people booking taxi. Recording and maintaining all these records is highly unreliable, inefficient and error-prone. It is also not economically & technically feasible to maintain these records on paper
3. Thus keeping the working of the manual system as the basis of our project. We have developed an automated version of the manual system, named as “Easy Cab”.
4. The main aim of our project is to provide a paper-less Easy Cab up to 90%. It also aims at providing low-cost reliable automation of the existing systems. The system also provides excellent security of data at every level of user-system interaction and also provides robust & reliable storage and backup facilities.
5. **PROJECT CATEGORY**

This Project is coupled with material on how to use the various tool, sub sets available in PHP AND My SQL.

The need of today’s software development is competence in a GUI based front-end tool, which can connect to Relational Database engines. This gives the programmer the opportunity to develop client server based commercial applications.

These applications give users the power and ease of a GUI with the multi user capabilities of Novell, UNIX or WinNT based RDBMS engines such as My SQL.

All the important coding techniques used by programmers, in OOPS based coding is brought out in full and in great detail.

1. **TOOLS/PLATFORMS HARDWARE & SOFTWARE REQUIREMENTS**

**HARDWARE & SOFTWARE**

**HARDWARE:**

Processor : Pentium 2.4 GHz or above

Memory : 256 MB RAM or above

Cache Memory : 128 KB or above

Printer : Laser Printer

Pen Drive : 5 GB

**SOFTWARE:**

Operating System : Windows XP (Professional), WAMP Server.

Font-End Tool : PHP, Java Script

Back-End : My SQL (phpmyadmin)

**PHP**

PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. **PHP**, or PHP: Hypertext Preprocessor, is a widely used, general-purpose scripting language that was originally designed for web development, to produce dynamic web pages. It can be embedded into HTML and generally runs on a web server, which needs to be configured to process PHP code and create web page content from it. It can be deployed on most web servers and on almost every operating system and platform free of charge. PHP is installed on over 20 million websites and 1 million web servers. PHP has evolved to include a command line interface capability and can also be used in standalone graphical applications.

**PHPMY ADMIN (My Sql)**

Php MyAdmin is a free software tool written in PHP intended to handle the administration of My SQL over the World Wide Web. Php MyAdmin supports a wide range of operations with MySQL. The most frequently used operations are supported by the user interface (managing databases, tables, fields, relations, indexes, users, permissions, etc), while you still have the ability to directly execute any SQL statement.

1. **ANALYSIS**

As we know that the system development starts with System Analysis known that “**A System is a collection of interrelated components that work together to achieve common objective and system analysis is the specification of what the system, I required to do**” it is conducted with the following objective in mind:

* Identification and analysis of customer needs.
* System evaluation for identification
* Performing economic and technical analysis
* Allocating functions to manpower
* Establishing cost and schedule constraints
* Creating a system definition that forms a foundation for all subsequent development activities.

According to the IEEE “The application of systematic, disciplined, quantifiable approach to the development operation, maintenance of software: that is the application of engineering to software.” According to the process model opted as Water fall life cycle in this model Design just comes after the requirement specification or analysis phase.

1. **SOFTWARE DEVELOPMENT LIFE CYCLE**

**Waterfall model: -**

It also called the linear sequential model or classic life cycle. Waterfall model suggests a

Systematic, sequential approach to software development that begins at the system level and progresses through analysis, design, coding testing and maintenance. I am implementing this software development life cycle model in my project because it is simple and easy to implement. It has following phases:-

**Software requirements analysis:-**

The requirements gathering process is intensified and focused specifically on software. To understand the nature of the programs to be built, the software engineer must understand the information domain for the software, as well as required function, behavior, performance, and interfacing. Requirements

for both the system and the software are documented and reviewed with the customer.

**Feasibility study:-**

This is determining whether the system requested is feasible or not. It includes:-

* **Technical feasibility** is it possible that the work can be done with current equipment, software technology and available personnel? And if new technology is required what is the possibility that it can be developed.
* **Economic feasibility will** there sufficient be benefits in creating the system.
* **Operational feasibility** wills the system be used if it is developed and implemented or will there be resistance from the users. As in my project information about requirements are gathered from many sources like hospitals, doctor, people medical corners etc

**Design: -**

Software design is actually a multi step process that focuses on four distinct attributes of a program: data structure, software architecture, interface representations, and procedural detail. The design process translates requirements into a representation of the software that can be assessed for quality before code generation begins. Like requirements, the design is documented and becomes part of the software configuration. As in my project design should be like that there is a login interface which will accept user id and password for login and there is another links will be use for registration of user there is a no. of button so that user can navigate from one page to another page but for accessing their email account they must use their login id and password.

**Code generation: -**

The design must be translated into machine readable form. The code generation step performs this task. If design is performed in a detailed manner, code generation can be accomplished mechanistically. As we know JSP is atoll which have no. of server controls so that we can put controls according to our design which is translated into HTML coding at back to make page dynamic we use CSSto make page interactive.

**Testing: -**

Once code has been generated, program testing begins. The testing process focuses on the logical internals of the software, assuring that all statements have been tested, and on the functional externals that is conduction tests to uncover errors and ensure that defined input will produce actual results that agree with required results. First we test the system with white box testing. The white box testing will test all the coding of the modules. It will check the entire logical path. Loops and conditions applied. After while box testing. It will undergo the black box testing in which a set of inputs N prepared to test the functional requirements of this module. In the unit list of this module interface is tested to ensure that information properly flows into and out of the program because the module is not a standalone program, drover software must be developed for this unit test the driver is nothing but a main program. To test the module is executed properly.

**Maintenance: -**

Software will undoubtedly undergo change after it is delivered to the customer. Change will occur because errors have been encountered, because the software must adapted to accommodate changes in its external environment or because the customer requires functional or performance enhancements. So that their

Will be scope to make changes in our system.

**DFD AND ER-DIAGRAM**

1. **Analysis (DFD and ER Diagram)**

**Context Level DFD**

CAB BOOKING SYSTEM

SOURCE/DESTINATION

DATE/TIME

TYPE OF CAB

FARE/KM & TOTAL FARE

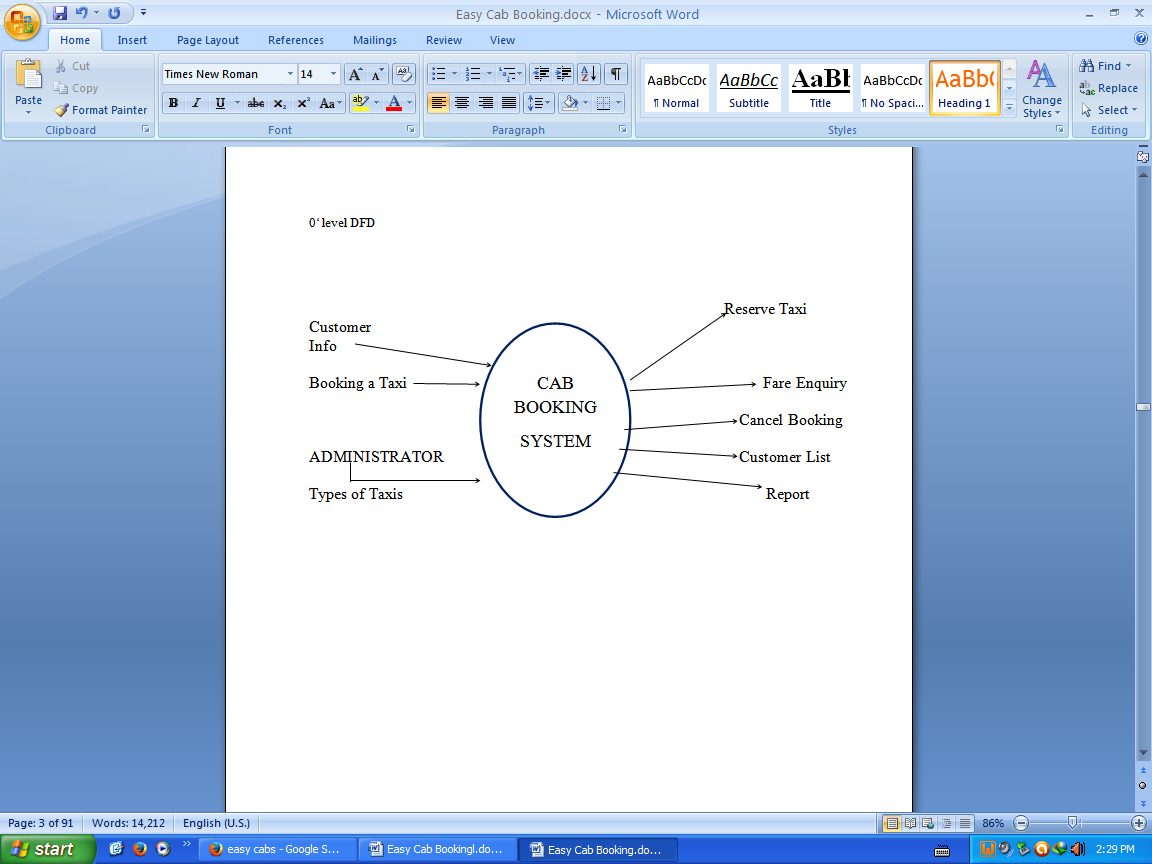
PAYMENT OPTION

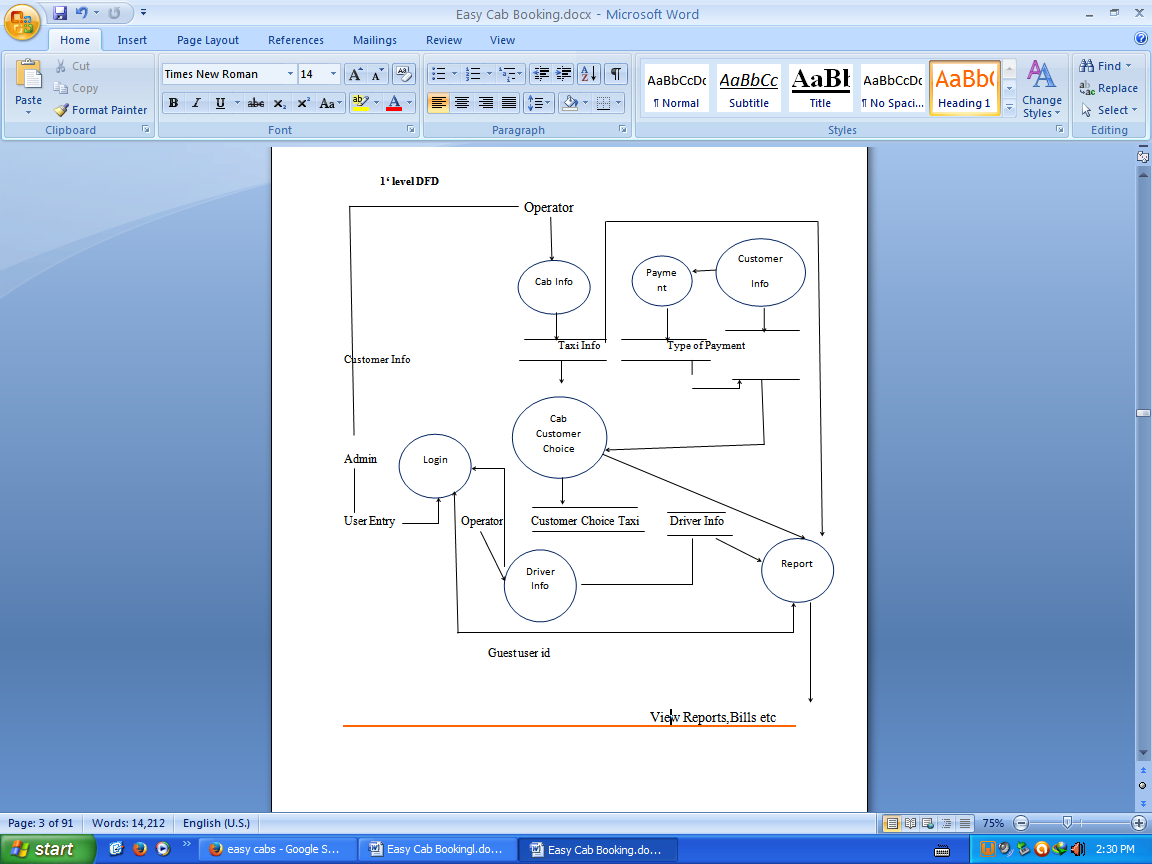
USER

ADMINISTRATOR

CUSTOMER NAME,EMAIL-ID NEME,ADDRESS AND MODE OF PAYMENT

REPORT

****



1. ER-Diagram

**CLIENT**

**Login**

Apply of project

Has Provided project

Has Feedback

**User**

**Project**

Give feedback

Are Distributed

Maintains the feedback

**Admin**

**Feedback**

**DATABASE DESIGN**

**Database Design**

The ECB database consists of 8 tables. Each and every table is setup with the proper Integrity constraints to work properly. The structure of the databasetables is given below:

**1. sign up table**

|  |  |
| --- | --- |
| Field name | Data type |
| First name | Nvarchar |
| Last name | Nvarchar |
| Address | nvarchar |
| City | Nvarchar |
| State | Nvarchar |
| Pin | Numeric |
| Contact | Numeric |
| Email | Nvarchar |
| Password | Nvarchar |
|  |  |

Table 4.1

**2. login table**

|  |  |
| --- | --- |
| **Field name** | **Data type** |
| **UserId** | **Nvarchar** |
| **Password** | **Nvarchar** |
|  |  |

1. **Module Description**

**Complete Structure of the Project**

Modules and their description ECB comprises of five modules. Brief description of each and every module is given below:

* **Module 1: Password Module**

In this module, registered user enters a password and the software checks its validity. If the password is valid then he is allowed to enter, otherwise “Invalid

#### Module 2: User Registration Module

#### In this module user can submit his details. After submitting his details, he/she can

#### his details. After submitting his details, he/she can easily ready for the interview.

#### Module 3: Booking Report Module

#### In this module, registered user can easily see his/her details by login into their account and book the cab.

#### Module 4:AckonwlgementModule

#### In this module,the acknowledgement of the registration is to be seen by the customer.

* **Module 5: Login Module**

In this module user login with registered user id or password.

1. **REPORT GENERATION**

**The reports generated by the administrator are:**

* **Users Details:** To get users details who are registered
* **Client Report:** To get details of the clients who have involved in project and the task completion details
* **Booking Report:** The progress of the project is provided to the client cab booking to keep them updated
* **Fare:** Client can view fare details without login.

**SYSTEM ANALYSIS**

1. **SYSTEM ANALYSIS**

P**roject Planning and Scheduling**

The ECB has gone through the following stages of development in its Software Development Life Cycle.

****

1. **SECURITY AND VALIDATION CHECKS**

Security is the major concern for any organization so that any unauthorized user can not access the organization data and any other secure things, same thing is applicable in our project there should be a proper authentication for each user so that nobody can access user account accept the authorized user.

In my project I have implemented the security checks. By providing each user with unique user id and password through user id and password user can access their email account and can read mails and can send mails.

**Further Enhancement**

1. **FUTURE SCOPE AND FURTHER ENHANCEMENT OF THE PROJECT**

Following modules can be added to enhance the features of the ECB.

• **Payment module** – This module can be added to ECB for maintaining the payment from the clients and payment to the employees.

**• Leave module** – This module can be added to the ECB for maintaining the leave and attendance of the employees.

**• Chat module** – This module can be added to the ECB for making clients chat with the programmers to modify their projects and tell them about their difficulties.

**• Reminder module** – This module can be added to the ECB for giving reminders of the deadline of the projects to the employee or giving deadline to the client for making payment of the proposed project.

• **SMS Module** - This module will be added to the ECB for giving reminder using SMS and every type of reminder using sms.

1. **BIBLIOGRAPHY**
2. MCA Study Material of MCS-014 - Course Material of the IGNOU MCA course relating to Systems Analysis and Design.
3. IGNOU MCA Study Material of MCS-023 - Course Material of the IGNOU MCA course relating to introduction to Database Management Systems.
4. IGNOU MCA Study Material of MCS-034 Course Material of the IGNOU MCA course relating to Software Engineering.
5. IGNOU MCA Study Material of MCS-043 Course Material of the IGNOU MCA course relating to Advance Database Management System.
6. IGNOU MCA Study Material of MCS-051 Course Material of the IGNOU MCA course relating to Advance Internet Technologies Beginning PHP6,Apache, MySql Wrox Publication http://www.php.net PHP official website.
7. IEEE Std 830-1998: IEEE Recommended practice for Software Requirements Specifications Software Engineering Standards Committee of the IEEE Society.
8. IEEE Std 1016-1998: IEEE Recommended practice for Software Design Descriptions Software Engineering Standards Committee of the IEEE Society.
9. IEEE Std 829-1998: IEEE Recommended effort Software Test Documentation Software Engineering Standards Committee of the IEEE Society.