**DESIGN AND IMPLEMENTATION OF COMPUTER BASED INVOICE GENERATING SYSTEM**

By

**Chukwu Nnadozie Orie**

Registration no:

**U14/NAS/CSC/051**

Being a BSc report submitted in partial fulfillment of the requirements for the award of a Bachelor’s degree in Computer Science of the Godfrey Okoye University.

**Supervisor: Miss Rosemary Ogugua**

**Department of Mathematics/Computer Science Godfrey Okoye University, Enugu**

**July, 2018**

**CERTIFICATION**

I hereby declare that the work presented herein was done by me and not by a third party. Should I be convicted of having cheated in this work, I shall accept the verdict of the university.

**ChukwuNnadozieOrie U14/NAS/CSC/051**

**APPROVAL PAGE**

In partial fulfillment of the requirement for the award of Bachelors in Computer in the Department of Computer Science this project was presented by CHUKWU NNADOZIE ORIE with Registration Number U14/NAS/CSC/051 has been approved by

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Miss**. **Ogugua Rosemary** Date/sign

Project Supervisor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 **Mrs. Monica Agu**  Date/sign

 Head of Department

**DEDICATION**

This work is dedicated to the God Almighty, for his love, kindness, mercies, guidance and protection during, within and after this work. And to my good parents MR. & MRS. ORIE G.C. for their supports through prayers, finance and otherwise to make this work a success. Also dedicate this work to my friends, family and colleagues and well-wishers who courage through the course of this work.

**ACKNOWLEDGEMENTS**

My gratitude goes to God Almighty who made it possible for me to come this far in my Academic pursuit. Without the guidance of my ever listening supervisor, Ms. OGUGUA ROSEMARY This work would not have been this successful. I appreciate your support, encouragement towards me. Also wish to appreciate my Head of Department for the help rendered and all her advises to the students. A big thanks my lecturer Dr. NDIDIAMAKA OZOFOR for his ready support, guidance and for being there for till the end of my Academic pursuit. You are wonderful.

My regard also goes to MRS. NJIDEKA ONOS, MR.BENSON, and MR. EJIKE EZINWANE, MRS. ANN NWOBODO for their support and advises. I wish to thank my parents MR & MRS ORIE G.C for their immense support. To all my relatives and siblings LINUS, OBICHERE, CHIMEZIE for being there for.

My gratitude also goes to all my friends David, Sampson, Augustine, Emmanuel, Ifeanyi, Johnbosco, Hector, Peter, Maxwell, goodness, Blessing and all others who assisted me in one way or another to make my project successful and also made my stay in Godfrey Okoye University peaceful.

## ABSTRACT

Design and implementation of computer based invoice generating system is one that generates a commercial recording document for both buyer and seller. It also identifies trading parties and list, describes and quantifies the items sold. Different record keeping systems has been practiced by different companies in this country Nigeria. This has resulted in not having a comprehensive data base for invoice recording. This has led to customers wasting a lot of time in an attempt to collect the manual invoice. Also there is poor accountability which has led to the embezzlement of fund. Design and implementation of computer based invoice generating system is aimed at building a web database for storing and recording of goods sold with invoice id number, giving account of invoice generated within a specified period of time and helps to manage and record payment. The methodology used is UNIFIED MODELING LANGUAGE (UML). UML is a standardized modeling language enabling developers to specify, visualize, construct and document artifacts of the system. The researcher used this methodology in the design of the software and it was implemented using notepad++ on a Microsoft windows system using HTML codes, PHP, CSS, and MySQL language for the relational database. The result gotten from this project was a computer based invoice generating system whereby the users of the system have to create account before having access to use the system and admin is the only person that has the privilege to edit, delete an invoice that is discarded by a customer.

.

**TABLE OF CONTENTS**

Tittle Page I

Certification II

Approval Page III

Dedication Page IV

Acknowledge Page V

Abstract VI

Table of Contents VII-VIII

List of Figures IX

List of Tables X

**CHAPTER 1**

1. Background of study 1

1.1 Statement of the problem 2

1.2 Objective of the study 2

1.3 Significance of the study 2

**CHAPTER 2**

1. Introduction 4

2.1 Theoretical background 4

2.2 Review of related literature 7

2.3 Summary 11

**CHAPTER 3**

3.0 Introduction 12

3.1 Analysis of existing system 16

3.2 Use case diagram 18

3.3 Analysis of proposed system 21

3.4 Design of the proposed system 21

3.4.1 Design of proposed system 21

3.4.2 System architecture 25

3.4.3 Input design 25

3.4.4 Output design 27

**CHAPTER 4**

4.0 Introduction 28

4.1 Choice of development environment 28

4.2 Implementation Architecture 31

4.3 Software Testing 32

4.3.1 Input Specification 33

4.3.2 Output Specification 34

4.4 Documentation 35

4.4.1 User Manual 36

**CHAPTER 5**

5.0 Summary 38

5.1 Conclusion 38

5.2 Recommendation 39

**References** 40

**Appendix** 42

## LIST OF FIGURES

Fig 3.1 UML Class Diagram 15

Fig 3.2 UML Activity Diagram 16

Fig 3.3 Sample of invoice Used during manual registration 17

Fig 3.4 Use case Diagram 19

Fig 3.5 Flow chart 20

Fig 3.6 System Architecture 25

Fig 3.7 Registration form 26

Fig 3.8Screenshot of the invoice page 27

Fig 4.1 Screenshot of Admin login page 29

Fig 4.2 Screenshot of Registration page 30

Fig 4.3 Implementation architecture 31

Fig 4.4 Screenshot of the Input form 33

Fig 4.5 Screenshot of the output form 34

Fig 4.6 Screenshot of output specification of invoice generated 35

## LIST OF TABLES

Table 3.1 Aigs Reg Table 22

Table 3.1 Aigs Invoice Table 22

**CHAPTER ONE**

**INTRODUCTION**

1. **BACKGROUND OF STUDY**

Record Keeping is the process of collecting, inputting, storing, classifying summarizing and interpreting data. Record keeping has two facets – financial record keeping and managerial record keeping such as date, address, number of goods purchased, unit price of goods, description of the goods, total amount of specified goods and ground total of all the goods (items) listed in the invoice. The basic purpose of financial record is to assist decision makers in evaluating the financial strength, profitability and future prospects of a business.

The purpose of financial record keeping is to produce financial statement about a business entity. Therefore, this study has been narrowed down to the record keeping system that has been practiced by different companies in Nigeria.

Automatic invoice generation system uses a sequential order of list of items that are entered in an invoice platform, to generate an invoice with items and its price per unit from the relational data base. The system automatically summarizes the listed items due for transactions in the ordered goods document and stores them in the invoice table.

But when invoice information is created manually, the system does not adhere to any sequence/summarization rules you have defined. Rather you

manually control the sequence/summarization logic in the way you manually create the pay items.

* 1. **STATEMENT OF THE PROBLEM**
1. Lack of comprehensive database for recording the invoicing,
2. Time wasted when many customers are waiting to collect manual invoice,
3. Manually written invoice are subjected to physical damage,
4. Embezzlement of fund due to poor accountability.
	1. **AIM AND OBJECTIVES OF STUDY**

The objectives include to develop a system that can:

1. Store and record all the goods sold with an invoice id number.
2. Manage and record payments.
3. Provide a unique identification number.
4. Give account of income generated within a specified period of time,
	1. **SIGNIFICANCE OF STUDY**

With growth of information technology, the study offers numerous values in sales invoice recording system.

1. Customers get good services, which include assurance of goods purchased: with this system the company knows that the customer can easy track them down if any of the goods sold to him/ her substandard goods or refurbished and with this the company will make it as a law to abide with the selling of original goods.
2. It will provide efficient, effective and timely services: there is no much time wasted when someone is using this system to run his/her business.
3. Corruption will be reduced: the apprentice hired to assist to attend to customers will not have access to the admin dashboard so that there won’t temper with the price of the goods sold.
4. There will be no conflict between the company and the customers since terms and conditions are clearly stated.
5. To make report and company audition easy and accountable: annual report writing will be easy since the system have a way of searching all the invoice produced with the customers’ business name and identification number of the invoice which make the invoice more authenticity in the labour market

**CHAPTER TWO**

**LITERATURE REVIEW**

**2.0 INTRODUCTION**

An invoice is a list of products or services given to a client that includes the cost of those services. In other words, an invoice is a bill. You send an invoice to someone because they owe your business money.

The purpose of invoice generation system is to minimize embezzlement and increase annual revenue yield to the company. The Web based invoice generation system, tends to change the traditional way of manual writing of invoice to typing and clicking of a button to printout the complete invoice,

The disadvantages of Web based invoice generation system, is centered on the area of illiteracy and availability of materials needed for the implementation by small companies, shop owners etc. This problem can be encountered mainly in rural areas where internet services are still limited.

**2.1 THEORITICAL BACKGROUND**

The project was built with web technologies which are HTML, CSS3, PHP5, and wamp which is the database that stores all the information the script will accept as input (MySQL and php MyAdmin), having Apache as the engine controller. And it was made with a user friendly capability, with the help of an application program known as a “web browser”, example Mozilla Firefox, Opera mini, Internet Explorer, Google Chrome, and Maxton Cloud etc.

Every html tag in the web document has a function it does on the web browser, example if I write

<html>

<head>

<title>Web Based automatic Invoice Generation System</title>

</head>

<body bgcolor=”red”>

<h4 align="left" >Web Based automatic Invoice Generation System<br />by <br />Chidozie</h4>

</body>

</html>

This will display a title web based automatic invoice generation system at the top of the web browser and the main page; Web Based automatic Invoice Generation System by Chidozie, all will be displayed in left alignment. A form was designed to collect data from user which involves: business name, contact address, phone number, description of goods, etc. which when entered and submitted with the submission button that sends the data to a browser, and into the relational database to hold the data.

HTML is the major tag, which forms the visual website that is used to create forms and specify a location onto which particular portion is mapped inside a web page e.g. <div id=”form”>form location

<form action=”” method=”post”>

Full name<input type=”text” name=”name\_1”/>

Contact Address<input type=”text” name=”name\_2”/>

Description of goods<input type=”text” name=”name\_3”/>

<input type=”submit” name=”submit” value=”Submit”/>

</form>

</div>

The HTML tag above when previewed on the web browser appears on the screen as follows:

Full name

Contact Address

Description of goods

**Submit**

The division (div) positions and held every site layout.Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a documentwritten in a mark-up language.

PHP is a server-side scripting language created in 1995 and designed for web development, but also used as a general-purpose programming language. PHP code where used to send the collected data from the user to MySQL relational database.

**2.2 REVIEW OF RELATED LITERATURE**

Welsh, Wanberg, Brown, & Simmering, “E-business (also referred to as Web-based management systems) is defined as a new context for business where a large amount of information and services describe the ecommerce practices in different business organizations,” 2003.

Sanam Kadge, Uzair Khan, Arsalan Thange, Shamail Mulla, and Harshika Gupta, stated thatSales and Invoice management system with analysis of customer behaviour is a system that allows the organisations to manage a company‟s sales and invoices using technology to organize, automate and synchronize sales and invoicing activities. This is a customer-oriented feature which includes tracking of customers‟ lifecycle right from the point they show interest in a particular product to the time they make the purchase. The system also performs customer behaviour analysis based on the same, 2004.[1]

Tony Brown said using a web browser, you are able to review and verify invoice details and create payment instructions online. db-eBills is a multi-user system with flexible access rights that can be adapted to your existing invoice approval and payment process. This solution also provides the capability for routing invoice information to the responsible departments or individuals for non-financial approval. Furthermore, you can print bills or integrate the billing information into other accounts payable and Enterprise Resource Planning (ERP) systems, 2004.[5]

Thomson ReutersSince electronic billing became a legal industry trend in the 1990s, corporate law departments have been touting its advantages to their law firms. Typically, they claim that processing electronic billing is almost no additional work for law firms, that the invoice approval process is faster, and that the firms will be able to shorten bill-top payment times. But for many law firms, the reality has been very different. Firms are finding they are doing much more work and getting paid no faster—and sometimes not at all. Law firms have not gained the same process and informational advantages from electronic billing as their in-house counterparts, so the idea that law firms will recoup their extra effort and expense when their invoices are paid faster has not come to pass. Some law firms, however, have implemented a product called eBillingHub, which is specifically designed to simplify electronic billing for law firms and help them gain many of the expected process and informational advantages. By comparing—in terms of dollars—the experiences of four major law firms, we can properly quantify the real costs of electronic billing, 2016.

 Each year companies send millions of invoices either by post or email (PDF) to the (central) government of the Netherlands. The manual handling of invoices is time consuming and costly for companies and central government alike. Electronic invoicing (e-billing) is a solution that offers benefits for both the sending and receiving parties.

Therefore, the central government in the Netherlands has, after consultation with the business community, made electronic invoicing compulsory for new contracts, starting 1 January 2017.

While suppliers with contracts earlier than 1 January 2017 are exempt from this requirement, they can voluntarily adopt e-billing in consultation with the relevant department. Suppliers who acquire their orders through the central government procurement system will continue to send their invoices in the customary manner.[6]

James Hammond, Rainmaker,which states that Law firms have the ability to streamline the electronic bill submission process by switching from their current manual, file copy process of uploading bills to clients to a centralized e-billing hub process. A centralized billing system can facilitate efficiency and improve cash flow by fully automating an otherwise tedious process and it can eliminate billing backlogs and verify submission to hundreds of clients or other e-billing intermediaries.

The key for law firms is to develop systems designed to counteract costs. Firms can leverage technology and implement good processes to ease invoice upload, avoid costly and time-consuming invoice rejections and provide simplicity into the entire electronic billing operation. Firms have learned that with automated e-billing, they can actually reduce the number of staff members or distribute more meaningful tasks to those extra law firm staff members who were once spending large amounts of time just on billing issues. E-billingis a tool that simplifies and quickens the invoice uploadprocess and provides law firms with a dashboard with ease to invoice status. It also provides template-driven approaches to preparation of e-bills, integrates with leading time and billing systems, performs an e-bill submission and accesses allpotential submission errors. An e-billing system offers central visibility into the status of all e-bills, across all clients and there is no additional hardware for firms to purchase, 2008.[8]

**2.3 Summary**

Automatic invoice generation system (billing record) needs total restructuring before effective and accurate means of sales record will be determined, by using specialized method as check to ensure that goods sold are properly record to a database for easy reporting on annual bases.

**CHAPTER THREE**

**SYSTEM ANALYSIS AND DESIGN**

**3.0 INTRODUCTION**

In this chapter, we analyzed the choice of methodology used in this research work, the analysis of the existing system and design. It includes specific methods which were used in order to achieve the aim of the research, particular requirements for implementation of the project and clear explanation of reasons why such method were used for design and implementation of the proposed system, also included is a brief description of the current system ofbusiness premises payment system.

The methodology used is Unified Modeling Language (UML). UML is a standardized modeling language enabling developers to specify, visualize, construct and document artifacts of a software system. Thus, UML makes these artifacts scalable, secure and robust in execution. UML is an important aspect involved in object-oriented software development. It uses graphic notation to create visual models of software systems.

UML is designed to enable users to develop an expressive, ready to use visual modeling language. In addition, it supports high level development concepts such as frameworks, patterns and collaborations.

UML diagrams can be divided into two categories. The first type includes six diagram types representing structural information. The second includes the remaining seven representing general types of behaviour. Structure diagrams are used in documenting the architecture of software systems and are involved in the system being modeled.

Different structure diagrams are:

* Class Diagram: represents system class, attributes and relationships among the classes.
* Component Diagram: represents how components are split in a software system and dependencies among the components.
* Deployment Diagram: describes the hardware used in system implementations.
* Composite Structure Diagram: describes internal structure of classes.
* Object Diagram: represents a complete or partial view of the structure of a modeled system.
* Package Diagram: represents splitting of a system into logical groupings and dependency among the grouping.

Behavior diagrams represent functionality of software system and emphasize on what must happen in the system being modeled.

Different behaviour diagrams are:

* Activity Diagram: represents step by step workflow of business and operational components.
* Use Case Diagram: describes functionality of a system in terms of actors, goals as use cases and dependencies among the use cases.
* UML State Machine Diagram: represents states and state transition.
* Communication Diagram: represents interaction between objects in terms of sequenced messages.
* Timing Diagrams: focuses on timing constraints.
* Interaction Overview Diagram: provides an overview and nodes representing communication diagrams.
* Sequence Diagram: represents communication between objects in terms of a sequence of messages.

UML diagrams represent static and dynamic views of a system model. The static view includes class diagrams and composite structure diagrams, which emphasize static structure of systems using objects, attributes, operations and relations. The dynamic view represents collaboration among objects and changes to internal states of objects through sequence, activity and state machine diagrams. A wide variety of UML modeling tools are available to simplify the modeling process, including IBM Rational Rose, Rational Rhapsody, MagicDraw UML, StarUML, ArgoUML, Umbrello, BOUML, PowerDesigner and Dia.

UML Class Diagram

Fig 3.1

UML Activity Diagram



**3.1 ANALYSIS OF EXISTING SYSTEM**

The existing record keeping system lacks some important functional units like instant invoicing system, storage of data in voice and easy retrieval etc.Successfulautomatic invoice generation system design, and development requires a collective and cohesive effort by the researcher. This is why it is very crucial to study and understand the existing system properly, its problems and lapses, its inefficiencies and inabilities.



Figure 3.2: Sample of invoice used during manual registration.

During the course of the analysis, the following problems were discovered:

1. Cost intensive and time consuming.
2. Too much paper work.
3. Loss of invoice.
4. Double issue of invoice.
5. Difficulty during annual revenue calculation.

Requirement Specification

**3.2 Use case Diagram**

As the most known diagram type of the behavioral UML diagrams, Use case diagrams give a graphic overview of the actors involved in a system, different functions needed by those actors and how these different functions interact.

It’s a great starting point for any project discussion because you can easily identify the main actors involved and the main processes of the system.

LOGIN

ADMIN

USER

SEARH INVOICE

INPUT DATA

EDIT INVOICE

ADD LOGO

ANALYSE GOODS

PRINT

ADD TOTAL

DELETE

**Fig. 3.3: Case Diagram**

**Flow Chat**

START

**IF LOGIN EXISTS**

NO

 YES

**DISPLAY ERROR**

**FILL THE INVOICE FOR YOUR CUSTOMERS**

**SEARCH BUSINESS NAMEE**

 YES

**RETURN TO SEARCH BOX**

**DISPLAY THE GENERATED INVOICE FOR THE BUSINESS NAME**

**END**

**Fig. 3.4: Case Diagram**

**3.3 ANALYSIS OF PROPOSED SYSTEM**

The user will insert all the details needed from him and from the customer who he will issue the invoice to. The data needed from the owner of the company are company logo, company name, company address, city, state, postal code, company phone number, company e-mail address, invoice number which will be generated automatic to ensure authenticity of the document. Also the company owner is required to key in the details of goods such as, description of goods, quantity of goods, unit price, and amount.

The data required from the customer side are client name (company name) client address, client city, client state, client postal address etc.

The system also have the ability to calculate the amount of money the customer will to the total box after inserting the goods data. The system also have the section where stored data will be retrieved using the invoice unique identification number to (ID NO) to search and printout the invoice, incase if the demand is needed in future. The system can also be used to edit, update, and delete to ensure that the data recorded in database are error free.

**3.4 DESIGN OF THE PROPOSED SYSTEM**

The designing of the proposed system are based on the below specifications.

**3.4.1 DATABASE DESIGN**

The database management system used in this research work is MySQL. This is open source relational database management system that uses structured query language.

The required tables in the database include the following:

* Aigs\_reg\_tab
* Aigs\_invoice\_tab

Table 1: Aigs\_reg\_tab

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field  | Data type | Size | Attributes | Null | Default | Extra |
| Aigs\_reg\_tab | Varchar | 11 |  | No | None | AUTO\_INCREMENT |
| Username | Varchar | 60 |  | No | None |  |
| First\_name | Varchar | 60 |  | No | None |  |
| Last\_name | varchar | 60 |  | No |  |  |
| Phone\_number | varchar | 25 |  | No |  |  |
| e-Mail | varchar | 255 |  | No |  |  |
| Date\_created | varchar | Datetime |  | No |  |  |

Table 2: Aigs\_invoice\_tab

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field  | Data type | Size | Attributes | Null | Default | Extra |
| Aigs\_invoice\_tab\_id | Varchar | 11 |  | No | None | AUTO\_INCREMENT |
| Logo | Varchar | 255 |  | yes | None |  |
| Client\_name | Varchar | 255 |  | yes | None |  |
| Client\_addres | varchar | 255 |  | yes | Null |  |
| Client\_city | varchar | 55 |  | yes |  |  |
| Client\_state | varchar | 50 |  | yes |  |  |
| Client\_postal\_code | varchar | 25 |  |  |  |  |
| Business\_name | varchar | 255 |  |  |  |  |
| Business\_address | varchar | 255 |  |  |  |  |
| Business\_city | varchar | 50 |  |  |  |  |
| Business\_state | varchar | 50 |  |  |  |  |
| Business\_phone | varchar | 50 |  |  |  |  |
| Business\_email | varchar | 50 |  |  |  |  |
| Business\_postal\_code | varchar | 25 |  |  |  |  |
| Date\_d | varchar | 25 |  |  |  |  |
| Invoice\_number | varchar | 25 |  |  |  |  |
| Description\_1 | varchar | 255 |  |  |  |  |
| Description\_2 | varchar | 255 |  |  |  |  |
| Description\_3 | varchar | 255 |  |  |  |  |
| Description\_4 | varchar | 255 |  |  |  |  |
| Description\_5 | varchar | 255 |  |  |  |  |
| Description\_6 | varchar | 255 |  |  |  |  |
| Quantity\_1 | Varchar | 4 |  |  |  |  |
| Quantity\_2 | Varchar | 4 |  |  |  |  |
| Quantity\_3 | Varchar | 4 |  |  |  |  |
| Quantity\_4 | Varchar | 4 |  |  |  |  |
| Quantity\_5 | Varchar | 4 |  |  |  |  |
| Quantity\_6 | varchar | 4 |  |  |  |  |
| Unit\_price\_1 | Varchar | 25 |  |  |  |  |
| Unit\_price\_2 | varchar | 25 |  |  |  |  |
| Unit\_price\_3 | Varchar | 25 |  |  |  |  |
| Unit\_price\_4 | varchar | 25 |  |  |  |  |
| Unit\_price\_5 | varchar | 255 |  |  |  |  |
| Unit\_price\_6 | varchar | 25 |  |  |  |  |
| Amt\_1 | Varchar | 25 |  |  |  |  |
| Amt\_2 | Varchar | 25 |  |  |  |  |
| Amt\_3 | Varchar | 25 |  |  |  |  |
| Amt\_4 | Varchar | 25 |  |  |  |  |
| Amt\_5 | Varchar | 25 |  |  |  |  |
| Amt\_6 | Varchar | 25 |  |  |  |  |
| Total  | Varchar | 25 |  |  |  |  |
| Amt\_paid | Varchar | 25 |  |  |  |  |
| Balance\_due | Varchar | 25 |  |  |  |  |

**3.4.2 System Architecture**

ADMIN LOGIN

DATA BASE

CALCULATE AMOUNT

SEARCH INVOICE

UPDATE INVOICE

EDIT INVOICE

FILE INVOICE DATA

ADMIN LOGIN

SYSTEM DASHBOARD

**Figure 3.3: System architecture**.

**3.4.3 Input Design**

The input for this recording system is by keyboard, mouse and scanner. During this input operation there are some data that are required for the acceptance of the invoice. These data are surname, first name, last name, phone number and e-mail address.

|  |
| --- |
| ADMIN REGISTRATION FORMSUBMITEMAILPHONE NUMBERLAST NAMEFIRST NMAESURNAME |

**Figure 3.4: Registration form**

The importance of the information is to ensure security and for authentication before using the system.

**3.4.4 Output Design**

This involves the various outputs gotten from the data input form in the system. The output of the automatic invoice generation system will be displayed on the web page which will contain the essential information for the company owner and the customer with full goods description or services. The output must be printed out and send to the customer or issue it at hand immediately.

LOGO

BUSINESS ADDRESS

STATE

BUSINESS NAME

CLIENT ADDRESS

CLIENT BUSINESS

POSTAL CODE

PHONE NO

CITY

PHONE

BALANCE

AMOUNT PAID

TOTAL

AMOUNT

UNIT PRICE

QUANTITY

DESCRIPTION

DATE

INVOICE NUMBER

E-MAIL

STATE

POSTAL CODE

**CHAPTER FOUR**

**SYSTEM IMPLEMENTATION**

**4.0 Introduction**

This chapter, contains a summary of the implementation of automatic invoice generation system. How the user can make the efficient use of the newly developed system of automatic invoice generation system. The system is user friendly, easy to understand andoperate.

**4.1 Choice of Development Environment**.

The system was developed as an interactive mechanism between the user at the interface, and the database using the web browser. It was designed using sublime text and wamp server. These tools enable the admin through a browser to interact with MySQL database to enter, edit, view, and retrieve data from the database, as a privilege granted. These activities were achieved using PHP, HTML forms which offer the best layout to enter data and view the database content. This form was also kept as short and simple as possible to suit the individual who will provide data.

The choice of programming used in this research work is top-down approach.

This approach was used because the project arrangement was in linear format and linear format is always from top to-down to achieve a goal.

**Admin Login Page**



**Figure 4.1: Screen shot of admin login page.**

**The user Registration Form.**



**Figure 4.2: Screen shot of registration form.**

This form can be accessible by the user (workers), through the administrator. This is because the administrator may not really have the time to start doing the work of apprentice, it is mandated to him/her to grant access the users. But he/she will always be around to monitor the work of the workers and also help them to edit any error, especially typing errors.

It is also necessary that the administrator should appoint someone who will assist in managing the database and admin page of the system.

 **4.2 Implementation Architecture**

ADMIN LOGIN

CALCULATE AMOUNT

SEARCH INVOICE

UPDATE INVOICE

EDIT INVOICE

USER ACCESS

ADMIN LOGIN

SYSTEM DASHBOARD

TOTAL OF AMOUNT

TABLE FOR RESULTFOUND

FILL INVOICE FORM

SUCCESS OR ERROR MESSAGE

CORRECT ERROR

SEARCH FOR INVOICE

PRINT THE INVOICE

DATA BASE

**Figure 4.3: Implementation architecture.**

**4.3 Software Testing**

During the development of the system, the application undergoes two phases of testing:

Firstly, testing done during development phase. This testing includes:

* Syntax error testing: this method of testing is used to check all the syntax errors during the development.
* Compatibility testing: this testing is used to test if all the languages used during the programming were compatible with each other.
* Logical testing: this is the argument involved during the programming. This will check whether the argument is accepted by the system or not.

Secondly, testing them by running the software on realistic data samples. This testing includes:

* Running the application in its fullness using a local server such as wamp server, xamp server, lamp server etc that have local host which uses Apache as engine and MYSQL as the database.
* Browser testing: this testing is used to test the appearance and shape of the software on the browser. This testing also replaces system testing in desktop application.

**4.3.1 INPUT SPECIFICATION**

These are the necessary data needed from each client or customer. This includes the information requirement from each customer to obtain an invoice. The input page requests the user to enter the following information on the fields.



**Figure 4.4: Screen shot of the input form,**

Once all the requested information are provided and click on submit the data will immediately send to database for storage and retrieval.

**4.3.2 Output Specification**

The output specification contains main information after all necessary data has been supplied by the user/apprentice.





Figure 4.5: Screenshot of output specification of invoice generated

**4.4 Documentation**

As earlier stated, this system was built with basic HTML tags, PHP, CSS, and MySQL language, they are open source program that allows modification to be done in future.

HTML: This is an acronym for Hyper Text Makeup Language; defining the structure of a webpage. Html forms are tags that allow the user to enter information (like text fields, text area fields, drop-down menus etc.) in a database.

CSS: This is the acronym for cascading style sheets, these codes are used to style html elements, when html tags are written and viewed on a web browser, it doesn’t really appear in animated format, so we use CSS to style the html tags.

PHP: Hypertext preprocessor is widely used scripting language that was originally designed for web development, to produce dynamic web pages. These codes can be embedded into html tags.

MySQL: This is my serial query language; it uses a standard form of well-known SQL data language. Different data types were used in this project. It was used aligns side with PHP to create and submit forms used in the software.

The developed system is packaged in a CD drive in a directory named *automatic invoice generation system.*

**4.4.1 User Manual**

Procedure on how to execute the program:

1. Boot the system to windows
2. Install sublime text and wamp server in the system
3. Launch Sublime text and wamp server
4. Click on start button and then select all programs.
5. Select sublime text and click on it.
6. Open the file on the CD drive and copy the file into a folder in the wamp server call “WWW”.
7. Click on start button again and select wamp server and start.
8. Go to the right hand site of the system on the task bar, open the wamp server to run the database to ensure that the modules are started.
9. Go to your browser type 127.0.0.1/aigs/user/index.php

The system has the following menu commands

1. *Search invoice*. On clicking the menu the system will display where the business name of the customer will be entered to get it printed and also ensure the business name is correct.
2. *View All Invoice*. On clicking this menu all the customers invoice that where submitted to the database will be displayed.
3. *Edit*. On clicking this menu it will open form that will give access for correction of error.
4. *Print*. On clicking this menu it will it will send the form to printer for the customer to have its hard copy on hand.
5. *Delete.* On clicking this menu it will erase the record that is not needed on the table.

**CHAPTER FIVE**

SUMMARY, RECOMMENDATIONS AND CONCLUSION

**5.0 Summary**

The automatic invoice generation system will web based business, will manage customer’s information with information of goods purchased.

This is also a method of determining the type of goods every customer normally buy and is also a means through which the business owner knows when there is market change, because you can quickly search the previous transaction made between you and your customers and analyze the present condition of market from the goods their demand and the one their buy more.

**5.1 Conclusion**

The most essential part of this project, automatic invoice generation system, is that it allows me to contribute what I have in mind, about record keeping in business parastatals.

This research gives the business owners confidence to buy more of a particular goods, since the record history shows that the goods is moving more in market demand.

**5.2 Recommendation**

After my research analysis, I observed that automatic invoice generation system will be the best practice to put in use, especially in this era of computer technology where many people embrace method record that involves the use of computer system. This system will also reduce embezzlement of money from the apprentice, since the record keeping is computerized and easy to trace back when things went wrong in the office.

**REFERENCES**

[1] SanamKadge, Uzair Khan and ArsalanThange*Sales and Invoice Management System with Analysis of Customer Behaviour,* 2016. [E-book] Available: https://www.ijcaonline.org/research/volume136/number10/kadge-2016-ijca-908580.pdf

[2] Chukwudi Kingsley Williams*Online Invoice Management SystemSoftware as a Service*, 9 November 2017.https://www.theseus.fi/bitstream/handle/10024/138879/williams\_chukwudi.pdf?sequence=1

[3] Department of Employee Trust Funds

*State Agency Health Insurance Administration Manual*, 2014.http://etf.wi.gov/employers/healthstate\_chap15.pdf

[4] NCHOE P. JOHNSON*A SURVE Y OF BENEFITS AN D CHALLENGES IN ELECTRONIC BILLING AND PAYMEN T IN THE KENYA POWER AND LIGHTING COMPANY LIMITED* , 2009.

http://erepository.uonbi.ac.ke/bitstream/handle/11295/23422/Fulltext.pdf?sequence=3

[5] Tony Brown  *Coordination of Benefits Agreement Electronic Billing Introductory Package,* 2004.

https://www.cms.gov/Medicare/Coordination-of-Benefits-and-Recovery/COBA-Trading-Partners/Downloads/Electronic\_Billing\_Pkg.pdf

[6] Thomson Reuters *The Real Cost of Electronic Billing, 2016.*

*http://ebillinghub.com/documents/2016/10/the-real-cost-of-electronic-billing.pdf/*

[7] Charles Smith *Electronic invoicing and the Dutch central government*, 2016*.*

file:///C:/Users/USER/Downloads/Electronic+invoicing+and+the+Dutch+central+government.pdf

[8] JAMES HAMMOND, RAINMAKER SOFTWARE *The Key to Efficiently E-bill and Increase Cash Flow,* 2008.

https://www.iltanet.org/HigherLogic/System/DownloadDocumentFile.ashx?DocumentFileKey=12ea8501-27c9-4920-94d8-3b7c0b22c099&forceDialog=0&ssopc=1

[9] UNECE *Trade facilitation Implimentation Guild,* 2012.

http://tfig.unece.org/contents/e-invoices.html

[10] J. I. Perere and J. Wagoki*EFFECT OF ELECTRONIC BILLING SERVICE ON CUSTOMER SATISFACTION AT KENYA POWER COMPANY*, 2016.

http://ijecm.co.uk/wp-content/uploads/2016/04/4425.pdf

APPENDIX 1

The codes would be broken down into section for the different page on the web application.

Code for: ***Conn.php Page***

*<?php*

*$conn = mysqli\_connect("localhost","root","")or die('could not connect to localhost server'.mysqli\_error());*

*$select\_db = mysqli\_select\_db($conn,"aigs")or die('could not select from database'.mysqli\_error());*

*?>*

Code for: ***Conn\_db.php Page***

*<?php*

*require('conn.php');*

*$bd\_namer = mysqli\_query($conn,"create database if not exists aigs")or die('could not create db'.mysqli\_error());*

*?>*

Code for: ***Coon\_tab.php Page***

*<?php*

*require('conn.php');*

*$reg\_aigs= mysqli\_query($conn, "create table if not exists reg\_aigs\_tab(*

 *reg\_aigs\_tab\_idint not null primary key auto\_increment,*

 *surnamevarchar(25),*

 *first\_namevarchar(25),*

 *last\_namevarchar(25),*

 *phone\_numbervarchar(25),*

 *emailvarchar(25),*

 *passwordvarchar(100),*

 *date\_createddatetime*

 *)*

*")or die('could not create aigs\_regtabel'.mysqli\_error($conn));*

*$aigs\_tab = mysqli\_query($conn, "create table if not exists aigs\_invoice\_tab(*

 *aigs\_invoice\_tab\_idint not null primary key auto\_increment,*

 *logovarchar(255),*

 *client\_namevarchar(255),*

 *client\_addressvarchar(255),*

 *client\_phone\_numbervarchar(255),*

 *client\_cityvarchar(55),*

 *client\_statevarchar(50),*

 *client\_postal\_codevarchar(25),*

 *business\_namevarchar(255),*

 *business\_addressvarchar(255),*

 *business\_cityvarchar(50),*

 *business\_statevarchar(50),*

 *business\_phonevarchar(50),*

 *business\_emailvarchar(50),*

 *business\_postal\_codevarchar(25),*

 *date\_dvarchar(25),*

 *invoice\_numbervarchar(25),*

 *description\_1varchar(255),*

 *description\_2varchar(255),*

 *description\_3varchar(255),*

 *description\_4varchar(255),*

 *description\_5varchar(255),*

 *description\_6varchar(255),*

 *quantity\_1varchar(4),*

 *quantity\_2varchar(4),*

 *quantity\_3varchar(4),*

 *quantity\_4varchar(4),*

 *quantity\_5varchar(4),*

 *quantity\_6varchar(4),*

 *unit\_price\_1varchar(25),*

 *unit\_price\_2varchar(25),*

 *unit\_price\_3varchar(25),*

 *unit\_price\_4varchar(25),*

 *unit\_price\_5varchar(25),*

 *unit\_price\_6varchar(25),*

 *amt\_1varchar(25),*

 *amt\_2varchar(25),*

 *amt\_3varchar(25),*

 *amt\_4varchar(25),*

 *amt\_5varchar(25),*

 *amt\_6varchar(25),*

 *totalvarchar(25),*

 *amt\_paidvarchar(25),*

 *Balance\_dueVarchar(25)*

 *)*

*")or die('could not create table for invoice'.mysqli\_error($conn));*

*?>*

Code for: ***Add.php Page***

*<html>*

*<head>*

*<script>*

*functionadd\_field()*

*{*

*vartotal\_text=document.getElementsByClassName("input\_text");*

*total\_text=total\_text.length+1;*

*document.getElementById("field\_div").innerHTML=document.getElementById("field\_div").innerHTML+*

 *"<p id='input\_text"+total\_text+"\_wrapper'><input type='text' class='input\_text' id='input\_text"+total\_text+"' placeholder='Enter Text'><input type='button' value='Remove' onclick=remove\_field('input\_text"+total\_text+"');></p>";*

*}*

*functionremove\_field(id)*

*{*

*document.getElementById(id+"\_wrapper").innerHTML="";*

*}*

*</script>*

*<style>*

*body*

*{*

*margin:0 auto;*

*padding:0;*

*text-align:center;*

*height:2000px;*

*background-color:silver;*

*}*

*#wrapper*

*{*

*width:995px;*

*padding:0px;*

*margin:0px auto;*

*font-family:helvetica;*

*text-align:center;*

*}*

*input[type="text"]*

*{*

*width:200px;*

*height:35px;*

*margin-right:2px;*

*border-radius:3px;*

*border:1px solid green;*

*padding:5px;*

*}*

*input[type="button"]*

*{*

*background:none;*

*color:white;*

*border:none;*

*width:110px;*

*height:35px;*

*border-radius:3px;*

*background-color:green;*

*font-size:16px;*

*}*

*</style>*

*</head>*

*<body>*

*<div id="wrapper">*

*<div id="field\_div">*

*<input type="button" value="Add TextBox" onClick="add\_field();" placeholder="amount">*

*</div>*

*</div>*

*</body>*

*</html>*

Code for: ***Aigs\_search.php Page***

*<?php*

*session\_start();*

*(isset($\_SESSION['aigs']))? $aigs\_id = $\_SESSION['aigs']:header('location:index.php');*

*include('../server/conn.php');*

*$err = array();*

*if(isset($\_POST['sssd']))*

*{*

 *$serch = $\_POST['sr2'];*

 *if($serch == '')*

 *{*

 *$err[] = "Use The Right Word to Search<br>";*

 *}*

 *$ck\_serch = mysqli\_query($conn, "select \* from aigs\_invoice\_tab*

 *where*

 *client\_name = '$serch'*

 *")or die('could not select from aigs\_invoice\_tab table'.mysqli\_error());*

 *$chk\_num = mysqli\_num\_rows($ck\_serch);*

 *if($chk\_num<1)*

 *{*

 *$err[] = "Business name or phone number doesnot Exists<br>";*

 *}*

 *$select = mysqli\_query($conn, "select \* from aigs\_invoice\_tab*

 *where*

 *client\_name = '$serch'*

 *")or die('could not select from aigs\_invoice\_tab table'.mysqli\_error());*

 *$ch\_rel\_num = mysqli\_num\_rows($select);*

 *if($ch\_rel\_num>0)*

 *{*

 *$fet\_aigs = mysqli\_fetch\_assoc($select);*

 *$aigs\_id = $fet\_aigs['aigs\_invoice\_tab\_id'];*

 *$biz\_name = $fet\_aigs['client\_name'];*

 *$cl\_phone = $fet\_aigs['client\_phone\_number'];*

 *$amt = $fet\_aigs['total'];*

 *}*

 *foreach($err as $bst\_aigs)*

 *{*

 *echo $bst\_aigs;*

 *}*

*}*

*?>*

*<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">*

*<html xmlns="http://www.w3.org/1999/xhtml">*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />*

*<link rel="shortcut icon" href="../images/aigs.fw.png"/>*

*<link href="../css/css.css" rel="stylesheet" type="text/css" />*

*<script src="aigs.js"></script>*

*<title>AIGS</title>*

*</head>*

*<body>*

*<div class="aigs\_wrapper">*

 *<div class="headers">*

 *<div class="header\_part">*

 *<div class="best\_header">*

*<imgsrc="../images/aigs.fw.png" class="img\_logo" />*

*<div class="main\_title">AUTOMATIC INVOICE GENERATION</div>*

*<div class="best\_ti\_sub"> SYSTEM</div>*

*<a class="sear1" href="#"></a>*

*<a class="sear" href="index.php">Logout</a>*

*</div>*

*</div>*

*</div>*

*<div class="body\_auto">*

 *<div class="auto\_main\_search">*

 *<div class="auto\_best\_search">*

 *<form action="" method="post">*

*<input type="search" name="sr2" placeholder="Search by phone Number or Company Name" class="best\_search" required="required"/>*

*<input type="submit" name="sssd" value="Search" class="sea\_bom"/>*

*</form>*

*<table cellpadding="1" class="auto\_tab">*

*<tr>*

*<th class="s1">S/NO</th>*

*<th class="sss11">COMPANY NAME</th>*

*<th class="ss2">PHONE NUMBER</th>*

*<th class="ss2">TOTAL AMOUNT</th>*

*<th class="ss7">EDIT</th>*

*<th class="ss7">PRINT</th>*

*<th class="ss7">DELETE</th>*

*<*

*<?php*

*if(isset($\_POST['sssd']))*

*{*

*if($ch\_rel\_num>0)*

*{*

*?>*

*<tr>*

*<th class="s1"><?php echo $aigs\_id;?></th>*

*<th class="sss111"><?php echo $biz\_name;?></th>*

*<th class="ss22"><?php echo $cl\_phone;?></th>*

*<th class="ss22"><?php echo $amt;?></th>*

*<th class="ss77"><a href="view.php?voi=<?php echo $aigs\_id;?>">Edit</a></th>*

*<th class="ss77"><a target="\_blank" href="print.php?pt=<?php echo $aigs\_id;?>">Print</a></th>*

*<th class="ss77"><a href="">Delete</a></th>*

*</tr>*

*<?php*

 *}}*

 *?>*

*</table>*

*</div>*

*</div>*

*<div class="footer\_auto">*

 *<div class="auto\_f">*

 *<div class="auto\_c">Alright Reserved and Designed by Nnadozie</div>*

*</div>*

*</div>*

*</div>*

*</body>*

*</html>*

Code for: ***Index.php Page***

*<?php*

*include('../server/conn.php');*

*$err = array();*

*if(isset($\_POST['sub\_aigs']))*

*{*

 *$mail = $\_POST['e-mail'];*

 *$pass = $\_POST['password'];*

 *if($mail== '')*

 *{*

 *$err[] = "Enter E-mail<br>";*

 *}*

 *if($pass == '')*

 *{*

 *$err[] = "Enter Password<br>";*

 *}*

 *$best\_chks = mysqli\_query($conn, "select \* from reg\_aigs\_tab*

 *where*

 *email = '$mail' and password = '$pass'*

 *")or die('could not select from from'.mysqli\_error());*

 *$chks\_num = mysqli\_num\_rows($best\_chks);*

 *if($chks\_num< 1)*

 *{*

 *$err[] = "E-mail or password is not correct<br>";*

 *}*

 *$checkers = mysqli\_query($conn, "select \* from reg\_aigs\_tab*

 *where*

 *email = '$mail' and password = '$pass'*

 *")or die('could not select from reg\_aigs\_tab table'.mysqli\_error());*

 *$checker\_num = mysqli\_num\_rows($checkers);*

 *if($checker\_num>0)*

 *{*

 *$feters= mysqli\_fetch\_assoc($checkers);*

 *$aigs\_id = $feters['reg\_aigs\_tab\_id'];*

 *echo $aigs\_id;*

 *session\_start();*

 *$\_SESSION['aigs'] = $aigs\_id;*

 *header("location:invoice.php");*

 *foreach($err as $chs\_be){echo $chs\_be;}*

 *}*

*}*

*?>*

*<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">*

*<html xmlns="http://www.w3.org/1999/xhtml">*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />*

*<link rel="shortcut icon" href="../images/aigs.fw.png"/>*

*<link href="../css/css.css" rel="stylesheet" type="text/css" />*

*<script src="aigs.js"></script>*

*<title>AIGS</title>*

*</head>*

*<body>*

*<div class="aigs\_wrapper">*

 *<div class="headers">*

 *<div class="header\_part">*

 *<div class="best\_header">*

*<imgsrc="../images/aigs.fw.png" class="img\_logo" />*

*<div class="main\_title">AUTOMATIC INVOICE GENERATION</div>*

*<div class="best\_ti\_sub"> SYSTEM</div>*

*</div>*

*</div>*

*</div>*

*<div class="body\_auto">*

 *<div class="auto\_main\_reg">*

 *<div class="auto\_best\_reg">*

 *<label class="meme">LOGIN</label>*

*<form action="" method="post" enctype="multipart/form-data" class="aigs\_from">*

 *<span class="error"><?phpforeach($err as $chs\_be){echo $chs\_be;}?></span>*

*<input type="email" name="e-mail" placeholder="E-mail" required="required"class="aigs\_reg" />*

*<input type="password" name="password" placeholder="Password" required="required" class="aigs\_reg" />*

*<input type="submit" name="sub\_aigs" value="SUBMIT" class="sun\_aigs" /><br />*

*<a class="arange" href="reg.php">REGISTER</a*

*</form>*

*</div>*

*</div>*

*<div class="footer\_auto">*

 *<div class="auto\_f">*

 *<div class="auto\_c">copyright @ aigs 2018 and Designed by Nnadozie</div>*

*</div>*

*</div>*

*</div>*

*</body>*

*</html>*

Code for: ***Login.php Page***

*<?php*

*include('../server/conn.php');*

*$err = array();*

*if(isset($\_POST['']))*

*{*

 *$mail = $\_POST['e-mail'];*

 *$pass = $\_POST['password'];*

 *if($mail== '')*

 *{*

 *$err[] = "Enter E-mail<br>";*

 *}*

 *if($pass == '')*

 *{*

 *$err[] = "Enter Password<br>";*

 *}*

 *$checkers = mysqli\_query($conn,"select \* from reg\_aigs\_tab*

 *where*

 *email = '$mail' and password = '$pass'*

 *")or die('could not select from reg\_aigs\_tab table'.mysqli\_error());*

 *$checker\_num = mysqli\_num\_rows($checkers);*

 *if($checker\_num>0)*

 *{*

 *$feters= mysqli\_fetch\_assoc($checkers);*

 *$aigs\_id = $feters['reg\_aigs\_tab\_id'];*

 *session\_start();*

 *$\_SESSION['aigs'] = $aigs\_id;*

 *header("location:invoice.php");*

 *}*

*}*

*?>*

*<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">*

*<html xmlns="http://www.w3.org/1999/xhtml">*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />*

*<link rel="shortcut icon" href="../images/aigs.fw.png"/>*

*<link href="../css/css.css" rel="stylesheet" type="text/css" />*

*<script src="aigs.js"></script>*

*<title>AIGS</title>*

*</head>*

*<body>*

*<div class="aigs\_wrapper">*

 *<div class="headers">*

 *<div class="header\_part">*

 *<div class="best\_header">*

*<imgsrc="../images/aigs.fw.png" class="img\_logo" />*

*<div class="main\_title">AUTOMATIC INVOICE GENERATION</div>*

*<div class="best\_ti\_sub"> SYSTEM</div>*

*</div>*

*</div>*

*</div>*

*<div class="body\_auto">*

 *<div class="auto\_main\_reg">*

 *<div class="auto\_best\_reg">*

 *<label class="meme">LOGIN</label>*

*<form action="" method="post" enctype="multipart/form-data" class="aigs\_from">*

*<input type="email" name="e-mail" placeholder="E-mail" class="aigs\_reg" />*

*<input type="password" name="password" placeholder="Password" class="aigs\_reg" />*

*<input type="submit" name="sub\_aigs" value="SUBMIT" class="sun\_aigs" />*

*</form>*

*</div>*

*</div>*

*<div class="footer\_auto">*

 *<div class="auto\_f">*

 *<div class="auto\_c">Alright Reserved and Designed by Nnadozie</div>*

*</div>*

*</div>*

*</div>*

*</body>*

*</html>*

Code for: ***Logout.php Page***

*<?php*

*include('../server/conn.php');*

*session\_start();*

*(isset($\_SESSION['aigs']))? $aigs\_id = $\_SESSION['aigs']:header('location:index.php');*

*session\_destroy();*

*?>*

Code for: ***Ozo.php Page***

*<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">*

*<html xmlns="http://www.w3.org/1999/xhtml">*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />*

*<title></title>*

*<link href="../css/css.css" rel="stylesheet" type="text/css" />*

*</head>*

*<body>*

*<div class="1">*

 *<div class="2">*

 *<div class="23"></div>*

*</div>*

 *<div class="2">*

 *<div class="23"></div>*

*</div>*

*</div>*

*</body>*

*</html>*

Code for: ***Up.php Page***

*<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">*

*<html xmlns="http://www.w3.org/1999/xhtml">*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />*

*<title></title>*

*<link href="../css/css.css" rel="stylesheet" type="text/css" />*

*</head>*

*<body>*

*<div class="1">*

 *<div class="2">*

 *<div class="23"></div>*

*</div>*

 *<div class="2">*

 *<div class="23"></div>*

*</div>*

*</div>*

*</body>*

*</html>*